ACL-SIGLEX 2005
Workshop on
Deep Lexical Acquisition

Proceedings of the Workshop

Workshop Chairs:
Timothy Baldwin
Anna Korhonen
Aline Villavicencio

30 June 2005
University of Michigan
Ann Arbor, Michigan, USA

Production and Manufacturing by
Omnipress Inc.
Post Office Box 7214
Madison, WI 53707-7214

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
75 Paterson Street, Suite 9
New Brunswick, NJ 08901
USA
Tel: +1-732-342-9100
Fax: +1-732-342-9339
acl@aclweb.org

Introduction

This volume contains the papers accepted for presentation at the ACL-SIGLEX 2005 Workshop on Deep Lexical Acquisition, held at the University of Michigan, Ann Arbor, USA, on the 30th of June, 2005.

This workshop is supported by SIGLEX, the Special Interest Group on the Lexicon of the Association for Computational Linguistics (`http://www.clres.com/siglex.html`). Its goal is to bring together researchers interested in different facets of the automatic acquisition of deep lexical information, e.g. in the areas of computational grammars, computational lexicography, machine translation, information retrieval, question-answering, and text mining.

Deep lexical resources include lexicons for linguistically-precise grammars, template sets for information extraction systems, and ontologies for word sense disambiguation. Such resources are critical for enhancing the performance of systems and for improving their portability between domains. Most deep lexical resources in current use have been developed manually by lexicographers at considerable cost, and yet have limited coverage and require labour-intensive porting to new tasks. Automatic lexical acquisition is a more promising and cost-effective approach to take, and is increasingly viable given recent advances in NLP and machine learning technology, and corpus availability. However, a number of important challenges still need addressing before benefits can be reaped in practical language engineering, such as the (multilingual) acquisition of deep lexical information from corpora and the implementation of accurate, large-scale, portable acquisition techniques.

In the call for papers we solicited papers describing aspects of deep lexical acquisition including:

- Automatic acquisition of deep lexical information: subcategorization, diathesis alternations, selectional preferences, lexical/semantic classes, qualia structure, lexical ontologies, semantic roles, word senses, etc.
- Methods for supervised, unsupervised and weakly supervised deep lexical acquisition: machine learning, statistical, example- or rule-based, hybrid etc.
- Large-scale, cross-domain, domain-specific and portable deep lexical acquisition
- Extending and refining existing lexical resources with automatically acquired information
- Evaluation of deep lexical acquisition
- Application of deep lexical acquisition to NLP applications (e.g. machine translation, information extraction, language generation, question-answering)
- Multilingual deep lexical acquisition

Of the 22 papers submitted, the programme committee selected 11 papers for publication, representative of the state of the art in this subject today. Each full-length submission was independently reviewed by three members of the program committee, who then collectively faced the difficult task of selecting a subset of papers for publication from a very strong field. The accepted papers include proposals for automatic annotation and extension of deep lexical resources, and methods for automatically acquiring deep lexical information. Languages targeted in the papers include English, Chinese, Japanese and Catalan.

We would like to thank all the authors who submitted papers, as well as the members of the program committee for the time and effort they contributed in reviewing the papers, and Chris Brew for complementing the workshop expertly with his invited talk. Our thanks go also to the organisers of the main conference, the publication chairs (Jason Eisner and Philipp Köhn) and the conference workshop committee (Mark Dras, Mary Harper, Dan Klein, Mirella Lapata and Shuly Wintner).

Timothy Baldwin, Anna Korhonen, Aline Villavicencio

Organizers:

Timothy Baldwin, University of Melbourne, Australia
Anna Korhonen, University of Cambridge, UK
Aline Villavicencio, University of Essex, UK

Program Committee:

Collin Baker, University of California Berkeley (USA)
Roberto Basili, University of Rome Tor Vergata (Italy)
Francis Bond, NTT (Japan)
Chris Brew, Ohio State University (USA)
Ted Briscoe, University of Cambridge (UK)
John Carroll, University of Sussex (UK)
Stephen Clark, University of Oxford (UK)
Sonja Eisenbeiss, University of Essex (UK)
Christiane Fellbaum, University of Princeton (USA)
Frederick Fouvry, University of Saarland (Germany)
Sadao Kurohashi, University of Tokyo (Japan)
Diana McCarthy, University of Sussex (UK)
Rada Mihalcea, University of North Texas (USA)
Tom O'Hara, University of Maryland, Baltimore County (USA)
Martha Palmer, University of Pennsylvania (USA)
Massimo Poesio, University of Essex (UK)
Philip Resnik, University of Maryland (USA)
Patrick Saint-Dizier, IRIT-CNRS (France)
Sabine Schulte im Walde, University of Saarland (Germany)
Mark Steedman, University of Edinburgh (UK)
Mark Stevenson, University of Sheffield (UK)
Suzanne Stevenson, University of Toronto (Canada)
Dominic Widdows, MAYA Design, Inc. (USA)
Yorick Wilks, University of Sheffield (UK)
Dekai Wu, Hong Kong University of Science and Technology (Hong Kong)

Invited Speaker:

Cris Brew, Linguistics, Computer Science and Engineering and Cognitive Science Departments, Ohio State University

Table of Contents

Conference Program

08:55-09:00	Opening remarks
09:00-09:30	*Data Homogeneity and Semantic Role Tagging in Chinese* Oi Yee Kwong and Benjamin K. Tsou
09:30-10:00	*Verb Subcategorization Kernels for Automatic Semantic Labeling* Alessandro Moschitti and Roberto Basili
10:00-10:30	*Identifying Concept Attributes Using a Classifier* Massimo Poesio and Abdulrahman Almuhareb
10:30-11:00	Coffee Break
11:00-11:30	*Automatically Learning Qualia Structures from the Web* Philipp Cimiano and Johanna Wenderoth
11:30-12:00	*Automatically Distinguishing Literal and Figurative usages of Highly Polysemous Verbs* Afsaneh Fazly, Ryan North and Suzanne Stevenson
12:00-12:30	*Automatic Extraction of Idioms using Graph Analysis and Asymmetric Lexicosyntactic Patterns* Dominic Widdows and Beate Dorow
12:30-14:00	Lunch
14:00-14:30	*Frame Semantic Enhancement of Lexical-Semantic Resources* Rebecca Green and Bonnie J. Dorr
14:30-15:30	*It might be Deep Enough, but is it Broad Enough? Diversity in the Lexicon* Invited Speaker - Chris Brew, Ohio State University
15:30-16:00	Coffee Break
16:00-16:30	*Bootstrapping Deep Lexical Resources: Resources for Courses* Timothy Baldwin
16:30-17:00	*Morphology vs. Syntax in Adjective Class Acquisition* Gemma Boleda, Toni Badia and Sabine Schulte im Walde
17:00-17:30	*Automatic Acquisition of Bilingual Rules for Extraction of Bilingual Word Pairs from Parallel Corpora* Hiroshi Echizen-ya, Kenji Araki and Yoshio Momouchi
17:30-18:00	*Approximate Searching for Distributional Similarity* James Gorman and James Curran
18:00-18:05	Closing remarks

Data Homogeneity and Semantic Role Tagging in Chinese

Oi Yee Kwong and Benjamin K. Tsou
Language Information Sciences Research Centre
City University of Hong Kong
Tat Chee Avenue, Kowloon, Hong Kong
{rlolivia, rlbtsou}@cityu.edu.hk

Abstract

This paper reports on a study of semantic role tagging in Chinese in the absence of a parser. We tackle the task by identifying the relevant headwords in a sentence as a first step to partially locate the corresponding constituents to be labelled. We also explore the effect of data homogeneity by experimenting with a textbook corpus and a news corpus, representing simple data and complex data respectively. Results suggest that while the headword location method remains to be improved, the homogeneity between the training and testing data is important especially in view of the characteristic syntax-semantics interface in Chinese. We also plan to explore some class-based techniques for the task with reference to existing semantic lexicons, and to modify the method and augment the feature set with more linguistic input.

1 Introduction

As the development of language resources progresses from POS-tagged corpora to syntactically annotated treebanks, the inclusion of semantic information such as predicate-argument relations becomes indispensable. The expansion of the Penn Treebank into a Proposition Bank (Kingsbury and Palmer, 2002) is a typical move in this direction. Lexical resources also need to be enhanced with semantic information (e.g. Fellbaum *et al.*, 2001). The ability to identify semantic role relations correctly is essential to many applications such as information extraction and machine translation; and making available resources with this kind of information would in turn facilitate the development of such applications.

Large-scale production of annotated resources is often labour intensive, and thus calls for automatic labelling to streamline the process. The task is essentially done in two phases, namely *recognising* the constituents bearing some semantic relationship to the target verb in a sentence, and then *labelling* them with the corresponding semantic roles.

In their seminal proposal, Gildea and Jurafsky (2002) approached the task using various features such as headword, phrase type, and parse tree path. While such features have remained the basic and essential features in subsequent research, parsed sentences are nevertheless required, for extracting the path features during training and providing the argument boundaries during testing. The parse information is deemed important for the performance of role labelling (Gildea and Palmer, 2002; Gildea and Hockenmaier, 2003).

More precisely, parse information is rather more critical for the identification of boundaries of candidate constituents than for the extraction of training data. Its limited function in training, for instance, is reflected in the low coverage reported (e.g. You and Chen, 2004). As full parses are not always accessible, many thus resort to shallow syntactic information from simple chunking, even though results often turn out to be less satisfactory than with full parses.

This limitation is even more pertinent for the application of semantic role labelling to languages which do not have sophisticated parsing resources. In the case of Chinese, for example, there is con-

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 1–9,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

siderable variability in its syntax-semantics interface; and when one comes to more nested and complex sentences such as those from news articles, it becomes more difficult to capture the sentence structures by typical examples.

Thus in the current study, we approach the problem in Chinese in the absence of parse information, and attempt to identify the headwords in the relevant constituents in a sentence to be tagged as a first step. In addition, we will explore the effect of training on different datasets, simple or complex, to shed light on the relative importance of parse information for indicating constituent boundaries in semantic role labelling.

In Section 2, related work will be reviewed. In Section 3, the data used in the current study will be introduced. Our proposed method will be explained in Section 4, and the experiment reported in Section 5. Results and future work will be discussed in Section 6, followed by conclusions in Section 7.

2 Related Work

The definition of semantic roles falls on a continuum from abstract ones to very specific ones. Gildea and Jurafsky (2002), for instance, used a set of roles defined according to the FrameNet model (Baker *et al.*, 1998), thus corresponding to the frame elements in individual frames under a particular domain to which a given verb belongs. Lexical entries (in fact not limited to verbs, in the case of FrameNet) falling under the same frame will share the same set of roles. Gildea and Palmer (2002) defined roles with respect to individual predicates in the PropBank, without explicit naming. To date PropBank and FrameNet are the two main resources in English for training semantic role labelling systems, as in the CoNLL-2004 shared task (Carreras and Màrquez, 2004) and SENSEVAL-3 (Litkowski, 2004).

The theoretical treatment of semantic roles is also varied in Chinese. In practice, for example, the semantic roles in the Sinica Treebank mark not only verbal arguments but also modifier-head relations (You and Chen, 2004). In our present study, we go for a set of more abstract semantic roles similar to the thematic roles for English used in VerbNet (Kipper *et al.*, 2002). These roles are generalisable to most Chinese verbs and are not dependent on particular predicates. They will be further introduced in Section 3.

Approaches in automatic semantic role labelling are mostly statistical, typically making use of a number of features extracted from parsed training sentences. In Gildea and Jurafsky (2002), the features studied include phrase type (*pt*), governing category (*gov*), parse tree path (*path*), position of constituent with respect to the target predicate (*position*), voice (*voice*), and headword (*h*). The labelling of a constituent then depends on its likelihood to fill each possible role *r* given the features and the target predicate *t*, as in the following, for example:

$$P(r \mid h, pt, gov, position, voice, t)$$

Subsequent studies exploited a variety of implementation of the learning component. Transformation-based approaches were also used (e.g. see Carreras and Màrquez (2004) for an overview of systems participating in the CoNLL shared task). Swier and Stevenson (2004) innovated with an unsupervised approach to the problem, using a bootstrapping algorithm, and achieved 87% accuracy.

While the estimation of the probabilities could be relatively straightforward, the trick often lies in locating the candidate constituents to be labelled. A parser of some kind is needed. Gildea and Palmer (2002) compared the effects of full parsing and shallow chunking; and found that when constituent boundaries are known, both automatic parses and gold standard parses resulted in about 80% accuracy for subsequent automatic role tagging, but when boundaries are unknown, results with automatic parses dropped to 57% precision and 50% recall. With chunking only, performance further degraded to below 30%. Problems mostly arise from arguments which correspond to more than one chunk, and the misplacement of core arguments. Sun and Jurafsky (2004) also reported a drop in *F*-score with automatic syntactic parses compared to perfect parses for role labelling in Chinese, despite the comparatively good results of their parser (i.e. the Collins parser ported to Chinese). The necessity of parse information is also reflected from recent evaluation exercises. For instance, most systems in SENSEVAL-3 used a parser to obtain full syntactic parses for the sentences, whereas systems participating in the CoNLL task were restricted to use only shallow syntactic information. Results reported in the former tend to be higher. Although the dataset may be a factor affecting the labelling performance, it nevertheless reinforces the usefulness of full syntactic information.

According to Carreras and Màrquez (2004), for English, the state-of-the-art results reach an F_1 measure of slightly over 83 using gold standard parse trees and about 77 with real parsing results. Those based on shallow syntactic information is about 60.

In this work, we study the problem in Chinese, treating it as a headword identification and labelling task in the absence of parse information, and examine how the nature of the dataset could affect the role tagging performance.

3 The Data

3.1 Materials

In this study, we used two datasets: sentences from primary school textbooks were taken as examples for simple data, while sentences from a large corpus of newspaper texts were taken as complex examples.

Two sets of primary school Chinese textbooks popularly used in Hong Kong were taken for reference. The two publishers were Keys Press and Modern Education Research Society Ltd. Texts for Primary One to Six were digitised, segmented into words, and annotated with parts-of-speech (POS). This results in a text collection of about 165K character tokens and upon segmentation about 109K word tokens (about 15K word types). There were about 2,500 transitive verb types, with frequency ranging from 1 to 926.

The complex examples were taken from a subset of the LIVAC synchronous corpus[1] (Tsou *et al.*, 2000; Kwong and Tsou, 2003). The subcorpus consists of newspaper texts from Hong Kong, including local news, international news, financial news, sports news, and entertainment news, collected in 1997-98. The texts were segmented into words and POS-tagged, resulting in about 1.8M character tokens and upon segmentation about 1M word tokens (about 47K word types). There were about 7,400 transitive verb types, with frequency ranging from 1 to just over 6,300.

3.2 Training and Testing Data

For the current study, a set of 41 transitive verbs common to the two corpora (hereafter referred to as textbook corpus and news corpus), with frequency over 10 and over 50 respectively, was sampled.

Sentences in the corpora containing the sampled verbs were extracted. Constituents corresponding to semantic roles with respect to the target verbs were annotated by a trained human annotator and the annotation was verified by another. In this study, we worked with a set of 11 predicate-independent abstract semantic roles. According to the *Dictionary of Verbs in Contemporary Chinese* (*Xiandai Hanyu Dongci Dacidian*, 現代漢語動詞大詞典 – Lin *et al.*, 1994), our semantic roles include the necessary arguments for most verbs such as agent and patient, or goal and location in some cases; and some optional arguments realised by adjuncts, such as quantity, instrument, and source. Some examples of semantic roles with respect to a given predicate are shown in Figure 1.

Altogether 980 sentences covering 41 verb types in the textbook corpus were annotated, resulting in 1,974 marked semantic roles (constituents); and 2,122 sentences covering 41 verb types in the news corpus were annotated, resulting in 4,933 marked constituents[2].

The role labelling system was trained on 90% of the sample sentences from the textbook corpus and the news corpus separately; and tested on the remaining 10% of both corpora.

4 Automatic Role Labelling

The automatic labelling was based on the statistical approach in Gildea and Jurafsky (2002). In Section 4.1, we will briefly mention the features used in the training process. Then in Sections 4.2 and 4.3, we will explain our approach for locating headwords in candidate constituents associated with semantic roles, in the absence of parse information.

[1] http://www.livac.org

[2] These figures only refer to the samples used in the current study. In fact over 35,000 sentences in the LIVAC corpus have been semantically annotated, covering about 1,500 verb types and about 80,000 constituents were marked.

4.1 Training

In this study, our probability model was based mostly on parse-independent features extracted from the training sentences, namely:

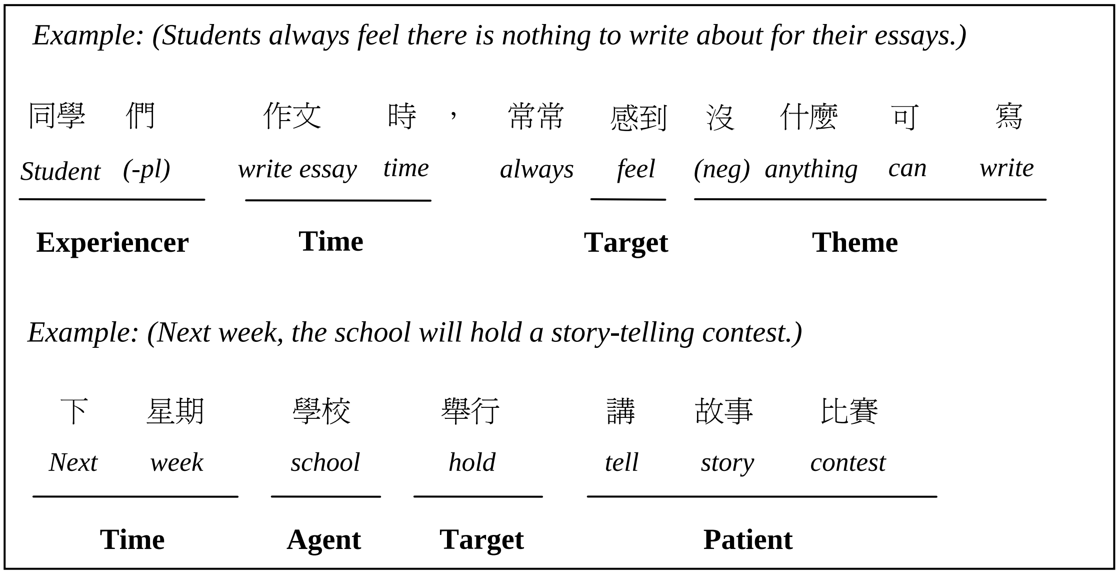
Headword (head): The headword from each constituent marked with a semantic role was identified. For example, in the second sentence in Figure 1, 學校 (school) is the headword in the constituent corresponding to the agent of the verb 舉行 (hold), and 比賽 (contest) is the headword of the noun phrase corresponding to the patient.

Position (posit): This feature shows whether the constituent being labelled appears before or after the target verb. In the first example in Figure 1, the experiencer and time appear on the left of the target, while the theme is on its right.

POS of headword (HPos): Without features provided by the parse, such as phrase type or parse tree path, the POS of the headword of the labelled constituent could provide limited syntactic information.

Preposition (prep): Certain semantic roles like time and location are often realised by prepositional phrases, so the preposition introducing the relevant constituents would be an informative feature.

Hence for automatic labelling, given the target verb *t*, the candidate constituent, and the above features, the role *r* which has the highest probability for *P(r* | *head*, *posit*, *HPos*, *prep*, *t*) will be assigned to that constituent. In this study, however, we are also testing with the unknown boundary condition where candidate constituents are not available in advance. To start with, we attempt to partially locate them by identifying their headwords first, as explained in the following sections.



Example: (Students always feel there is nothing to write about for their essays.)

同學 們	作文 時 ，	常常	感到	沒 什麼 可 寫
Student (-pl)	*write essay time*	*always*	*feel*	*(neg) anything can write*
Experiencer	**Time**		**Target**	**Theme**

Example: (Next week, the school will hold a story-telling contest.)

下 星期	學校	舉行	講 故事 比賽
Next week	*school*	*hold*	*tell story contest*
Time	**Agent**	**Target**	**Patient**

Figure 1 Examples of semantic roles with respect to a given predicate

4.2 Locating Candidate Headwords

In the absence of parse information, and with constituent boundaries unknown, we attempt to partially locate the candidate constituents by identifying their corresponding headwords first. Sentences in our test data were segmented into words and POS-tagged. We thus divide the recognition process into two steps, locating the headword of a candidate constituent first, and then expanding from the headword to determine its boundaries.

Basically, if we consider every word in the same sentence with the target verb (both to its left and to its right) a potential headword for a candidate constituent, what we need to do is to find out the most probable words in the sentence to match against individual semantic roles. We start with a feature set with more specific distributions, and back off to feature sets with less specific distributions[3]. Hence in each round we look for

$$\arg\max_{r} P(r \mid \textit{feature set})$$

for every candidate word. Ties are resolved by giving priority to the word nearest to the target verb in the sentence.

Figure 2 shows an example illustrating the procedures for locating candidate headwords. The target verb is 發現 (discover). In the first round, using features *head*, *posit*, *HPos*, and *t*, 時候 (time) and 問題 (problem) were identified as Time and Patient respectively. In the fourth subsequent round, backing off with features *posit* and *HPos*, 我們 (we) was identified as a possible Agent. In this round a few other words were identified as potential Patients. However, they would not be considered since Patient was already located in a previous round. So in the end the headwords identified for the test sentence are 我們 for Agent, 問題 for Patient and 時候 for Time.

4.3 Constituent Boundary

Upon the identification of headwords for potential constituents, the next step is to expand from these headwords for constituent boundaries. Although we are not doing this step in the current study, it can potentially be done via some finite state techniques, or better still, with shallow syntactic processing like simple chunking if available.

5 The Experiment

5.1 Testing

The system was trained on the textbook corpus and the news corpus separately, and tested on both corpora (the data is *homogeneous* if the system is trained and tested on materials from the same source). The testing was done under the "known constituent" condition and "unknown constituent" condition. The former essentially corresponds to the known-boundary condition in related studies; whereas in the unknown-constituent condition, which we will call "headword location" condition hereafter, we tested our method of locating candidate headwords as explained above in Section 4.2. In this study, every noun, verb, adjective, pronoun, classifier, and number within the test sentence containing the target verb was considered a potential headword for a candidate constituent corresponding to some semantic role. The performance was measured in terms of the precision (defined as the percentage of correct outputs among all outputs), recall (defined as the percentage of correct outputs among expected outputs), and F_1 score which is the harmonic mean of precision and recall.

5.2 Results

The results are shown in Tables 1 and 2, for training on homogeneous dataset and different dataset respectively, and testing under the known constituent condition and the headword location condition.

When trained on homogeneous data, the results were good on both datasets under the known constituent condition, with an F_1 score of about 90. This is comparable or even better to the results reported in related studies for known boundary condition. The difference is that we did not use any parse information in the training, not even phrase type. When trained on a different dataset, however, the accuracy was maintained for textbook data, but it decreased for news data, for the known constituent condition.

For the headword location condition, the performance in general was expectedly inferior to that for the known constituent condition. Moreover, this degradation seemed to be quite consistent in most cases, regardless of the nature of the training set. In fact, despite the effect of training set on news data, as mentioned above, the degradation

[3] In this experiment, we back off in the following order: *P(r|head, posit, HPos, prep t), P*(*r*|*head, posit, t*), *P*(*r* | *head, t*), *P*(*r* | *HPos, posit, t*), *P*(*r* | *HPos, t*). However, the *prep* feature becomes obsolete when constituent boundaries are unknown.

from known constituent to headword location is nevertheless the least for news data when trained on different materials.

Hence the effect of training data is only obvious in the news corpus. In other words, both sets of training data work similarly well with textbook test data, but the performance on news test data is worse when trained on textbook data. This is understandable as the textbook data contain fewer examples and the sentence structures are usually much simpler than those in newspapers. Hence the system tends to miss many secondary roles like location and time, which are not sufficiently represented in the textbook corpus. The conclusion that training on news data gives better result might be premature at this stage, given the considerable difference in the corpus size of the two datasets. Nevertheless, the deterioration of results on textbook sentences, even when trained on news data, is simply reinforcing the importance of data homogeneity, if nothing else. More on data homogeneity will be discussed in the next section.

In addition, the surprisingly low precision under the headword location condition is attributable to a technical inadequacy in the way we break ties. In this study we only make an effort to eliminate multiple tagging of the same role to the same target verb in a sentence on either side of the target verb, but not if they appear on both sides of the target verb. This should certainly be dealt with in future experiments.

Sentence:
溫習的時候，我們發現了許多平時沒有想到，或是未能解決的問題，於是就去問爸爸。
During revision, we discover a lot of problems which we have not thought of or cannot be solved, then we go and ask father.

Candidate Headwords	*Round 1*	…	*Round 4*	*Final Result*
溫習 (revision)			~~Patient~~	
時候 (time)	Time		----	***Time***
我們 (we)			Agent	***Agent***
平時 (normally)				
想到 (think)			~~Patient~~	
能 (can)				
解決 (solve)			~~Patient~~	
問題 (problem)	Patient		----	***Patient***
去 (go)			~~Patient~~	
問 (ask)			~~Patient~~	
爸爸 (father)			~~Patient~~	

Figure 2 Example illustrating the procedures for locating candidate headwords

	Textbook Data			News Data		
	Precision	Recall	F_1	Precision	Recall	F_1
Known Constituent	93.85	87.50	90.56	90.49	87.70	89.07
Headword Location	46.12	61.98	52.89	38.52	52.25	44.35

Table 1 Results for Training on Homogeneous Datasets

	Textbook Data			News Data		
	Precision	Recall	F_1	Precision	Recall	F_1
Known Constituent	91.85	88.02	89.86	80.30	66.80	72.93
Headword Location	38.87	57.29	46.32	37.89	42.01	39.84

Table 2 Results for Training on Different Datasets

6 Discussion

6.1 Role of Parse Information

According to Carreras and Màrquez (2004), the state-of-the-art results for semantic role labelling systems based on shallow syntactic information is about 15 lower than those with access to gold standard parse trees, i.e., around 60. With homogeneous training and testing data, our experimental results for the headword location condition, with no syntactic information available at all, give an F_1 score of 52.89 and 44.35 respectively for textbook data and news data. Such results are in line with and comparable to those reported for the unknown boundary condition with automatic parses in Gildea and Palmer (2002), for instance. Moreover, when they used simple chunks instead of full parses, the performance resulted in a drop to below 50% precision and 35% recall with relaxed scoring, hence their conclusion on the necessity of a parser.

The more degradation in performance observed in the news data is nevertheless within expectation, and it suggests that simple and complex data seem to have varied dependence on parse information. We will further discuss this below in relation to data homogeneity.

6.2 Data Homogeneity

The usefulness of parse information for semantic role labelling is especially interesting in the case of Chinese, given the flexibility in its syntax-semantics interface (e.g. the object after 吃 ‘eat’ could refer to the *patient* as in 吃蘋果 ‘eat apple’, *location* as in 吃食堂 ‘eat canteen’, *duration* as in 吃三年 ‘eat three years’, etc.).

As reflected from the results, the nature of training data is obviously more important for the news data than the textbook data; and the main reason might be the failure of the simple training data to capture the many complex structures of the news sentences, as we suggested earlier. The relative flexibility in the syntax-semantics interface of Chinese is particularly salient; hence when a sentence gets more complicated, there might be more intervening constituents and the parse information would be useful to help identify the relevant ones in semantic role labelling.

With respect to the data used in the experiment, we tried to explore the complexity in terms of the average sentence length and number of semantic role patterns exhibited. For the news data, the average sentence length is around 59.7 characters (syllables), and the number of semantic role patterns varies from 4 (e.g. 打算 ‘to plan’) to as many as 25 (e.g. 進行 ‘to proceed with some action’), with an average of 9.5 patterns per verb. On the other hand, the textbook data give an average sentence length of around 39.7 characters, and the number of semantic role patterns only varies from 1 (e.g. 決定 ‘to decide’) to 11 (e.g. 舉行 ‘to hold some event’), with an average of 5.1 patterns per verb. Interestingly, the verb 進行, being very polymorphous in news texts, only shows 5 different patterns in textbooks.

Thus the nature of the dataset for semantic role labelling is worth further investigation. The design of the method and the feature set should benefit from more linguistic analysis and input.

6.3 Future Work

In terms of future development, apart from improving the handling of ties in our method, as mentioned above, we plan to expand our work in several respects. The major part would be on the generalization to unseen headwords and unseen predicates. As is with other related studies, the examples available for training for each target verb are very limited; and the availability of training data is also insufficient in the sense that we cannot expect them to cover all target verb types. Hence it is very important to be able to generalize the process to unseen words and predicates. To this end we will experiment with a semantic lexicon like *Tongyici Cilin* (同義詞詞林, a Chinese thesaurus) in both training and testing, which we expect to improve the overall performance.

Another area of interest is to look at the behaviour of near-synonymous predicates in the tagging process. Many predicates may be unseen in the training data, but while the probability estimation could be generalized from near-synonyms as suggested by a semantic lexicon, whether the similarity and subtle differences between near-synonyms with respect to the argument structure and the corresponding syntactic realisation could be distinguished would also be worth studying. Related to this is the possibility of augmenting the feature set. Xue and Palmer (2004), for instance, looked into new features such as syntactic frame, lexicalized constituent type, etc., and found that enriching the feature set improved the labelling performance. In particular, given the importance of data homogeneity as observed from the experimental results, and the challenges posed by the characteristic nature of Chinese, we intend to improve our method and feature set with more linguistic consideration.

7 Conclusion

The study reported in this paper has thus tackled semantic role labelling in Chinese in the absence of parse information, by attempting to locate the corresponding headwords first. We experimented with both simple and complex data, and have explored the effect of training on different datasets. Using only parse-independent features, our results under the known boundary condition are comparable to those reported in related studies. The headword location method can be further improved. More importantly, we have observed the importance of data homogeneity, which is especially salient given the relative flexibility of Chinese in its syntax-semantics interface. As a next step, we plan to explore some class-based techniques for the task with reference to existing semantic lexicons, and to modify the method and augment the feature set with more linguistic input.

Acknowledgements

This work is supported by Competitive Earmarked Research Grants (CERG) of the Research Grants Council of Hong Kong under grant Nos. CityU1233/01H and CityU1317/03H.

References

Baker, C.F., Fillmore, C.J. and Lowe, J.B. (1998) The Berkeley FrameNet Project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL '98)*, Montreal, Quebec, Canada, pp.86-90.

Carreras, X. and Màrquez, L. (2004) Introduction to the CoNLL-2004 Shared Task: Semantic Role Labeling. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004)*, Boston, Massachusetts, pp.89-97.

Fellbaum, C., Palmer, M., Dang, H.T., Delfs, L. and Wolf, S. (2001) Manual and Automatic Semantic Annotation with WordNet. In *Proceedings of the NAACL-01 SIGLEX Workshop on WordNet and Other Lexical Resources*, Invited Talk, Pittsburg, PA.

Gildea, D. and Jurafsky, D. (2002) Automatic Labeling of Semantic Roles. *Computational Linguistics, 28(3)*: 245-288.

Gildea, D. and Palmer, M. (2002) The Necessity of Parsing for Predicate Argument Recognition. In *Proceedings of the 40th Meeting of the Association for Computational Linguistics (ACL-02)*, Philadelphia, PA.

Gildea, D. and Hockenmaier, J. (2003) Identifying Semantic Roles Using Combinatory Categorial Grammar. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, Sapporo, Japan.

Kingsbury, P. and Palmer, M. (2002) From TreeBank to PropBank. In *Proceedings of the Third Conference on Language Resources and Evaluation (LREC-02)*, Las Palmas, Canary Islands, Spain.

Kipper, K., Palmer, M. and Rambow, O. (2002) Extending PropBank with VerbNet Semantic Predicates. In *Proceedings of the AMTA-2002 Workshop on Applied Interlinguas*, Tiburon, CA.

Kwong, O.Y. and Tsou, B.K. (2003) Categorial Fluidity in Chinese and its Implications for Part-of-speech Tagging. In *Proceedings of the Research Note Session of the 10th Conference of the European Chapter* *of the Association for Computational Linguistics*, Budapest, Hungary, pages 115-118.

Lin, X., Wang, L. and Sun, D. (1994) *Dictionary of Verbs in Contemporary Chinese*. Beijing Language and Culture University Press.

Litkowski, K.C. (2004) SENSEVAL-3 Task: Automatic Labeling of Semantic Roles. In *Proceedings of the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text (SENSEVAL-3)*, Barcelona, Spain, pp.9-12.

Sun, H. and Jurafsky, D. (2004) Shallow Semantic Parsing of Chinese. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2004)*, Boston, pp.249-256.

Swier, R.S. and Stevenson, S. (2004) Unsupervised Semantic Role Labelling. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, pp.95-102.

Tsou, B.K., Tsoi, W.F., Lai, T.B.Y., Hu, J. and Chan, S.W.K. (2000) LIVAC, A Chinese Synchronous Corpus, and Some Applications. In *Proceedings of the ICCLC International Conference on Chinese Language Computing*, Chicago, pp. 233-238.

Xue, N. and Palmer, M. (2004) Calibrating Features for Semantic Role Labeling. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, pp.88-94.

You, J-M. and Chen, K-J. (2004) Automatic Semantic Role Assignment for a Tree Structure. In *Proceedings of the 3rd SigHAN Workshop on Chinese Language Processing*, ACL-04, Barcelona, pp.109-115.

啓思中國語文 *Qisi Zhongguo Yuwen*. Primary 1-6, 24 volumes, 2004. Hong Kong: Keys Press.

現代中國語文 *Xiandai Zhongguo Yuwen*. Primary 1-6, 24 volumes, 2004. Hong Kong: Modern Education Research Society Ltd.

Verb subcategorization kernels for automatic semantic labeling

Alessandro Moschitti and **Roberto Basili**
Department of Computer Science
University of Rome ”Tor Vergata”
Rome, Italy
{moschitti,basili}@info.uniroma2.it

Abstract

Recently, many researches in natural language learning have considered the representation of complex linguistic phenomena by means of structural kernels. In particular, tree kernels have been used to represent verbal subcategorization frame (SCF) information for predicate argument classification. As the SCF is a relevant clue to learn the relation between syntax and semantic, the classification algorithm accuracy was remarkable enhanced. In this article, we extend such work by studying the impact of the SCF tree kernel on both PropBank and FrameNet semantic roles. The experiments with Support Vector Machines (SVMs) confirm a strong link between the SCF and the semantics of the verbal predicates as well as the benefit of using kernels in diverse and complex test conditions, e.g. classification of unseen verbs.

1 Introduction

Some theories of verb meaning are based on syntactic properties, e.g. the alternations of verb arguments (Levin, 1993). In turn, Verb Subcategorization Frame (SCF) characterizes different syntactic alternations, thus, it plays a central role in the linking theory between verb semantics and their syntactic structures.

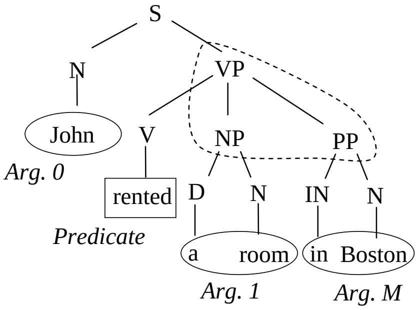


Figure 1: A predicate argument structure in a parse-tree representation.

Figure 1 shows the parse tree for the sentence `"John rented a room in Boston"` along with the semantic shallow information embodied by the verbal predicate *to rent* and its three arguments: Arg0, Arg1 and ArgM. The SCF of such verb, i.e. `NP-PP`, provides a synthesis of the predicate argument structure.

Currently, the systems which aim to derive semantic shallow information from texts recognize the SCF of a target verb and represent it as a flat feature (e.g. (Xue and Palmer, 2004; Pradhan et al., 2004)) in the learning algorithm. To achieve this goal, a lexicon which describes the SCFs for each verb, is required. Such a resource is difficult to find especially for specific domains, thus, several methods to automatically extract SCF have been proposed (Korhonen, 2003). In (Moschitti, 2004), an alternative to the SCF extraction was proposed, i.e. the SCF kernel (SK). The subcategorization frame of verbs was implicitly represented by means of the syntactic subtrees which include the predicate with its arguments. The similarity between such syntactic structures was evaluated by means of convolution kernels.

Convolution kernels are machine learning approaches which aim to describe structured data in

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 10–17,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

terms of its substructures. The similarity between two structures is carried out by kernel functions which determine the number of common substructures without evaluating the overall substructure space. Thus, if we associate two SCFs with two subtrees, we can measure their similarity with such functions applied to the two trees. This approach determines a more syntactically motivated verb partition than the traditional method based on flat SCF representations (e.g. the `NP-PP` of Figure 1). The subtrees associated with SCF group the verbs which have similar syntactic realizations, in turn, according to Levin's theories, this would suggest that they are semantically related.

A preliminary study on the benefit of such kernels was measured on the classification accuracy of semantic arguments in (Moschitti, 2004). In such work, the improvement on the PropBank arguments (Kingsbury and Palmer, 2002) classification suggests that SK adds information to the prediction of semantic structures. On the contrary, the performance decrease on the FrameNet data classification shows the limit of such approach, i.e. when the syntactic structures are shared among several semantic roles SK seems to be useless.

In this article, we use Support Vector Machines (SVMs) to deeply analyze the role of SK in the automatic predicate argument classification. The major novelty of the article relates to the extensive experimentation carried out on the PropBank (Kingsbury and Palmer, 2002) and FrameNet (Fillmore, 1982) corpora with diverse levels of task complexity, e.g. test instances of unseen predicates (typical of free-text processing). The results show that: (1) once a structural representation of a linguistic object, e.g. SCF, is available we can use convolution kernels to study its connections with another linguistic phenomenon, e.g. the semantic predicate arguments. (2) The tree kernels automatically derive the features (structures) which support also a sort of back-off estimation in case of unseen verbs. (3) The structural features are in general robust in all testing conditions.

The remainder of this article is organized as follows: Section 2 defines the Predicate Argument Extraction problem and the standard solution to solve it. In Section 3 we present our kernels whereas in Section 4 we show comparative results among SVMs using standard features and the proposed kernels. Finally, Section 5 summarizes the conclusions.

2 Parsing of Semantic Roles and Semantic Arguments

There are two main resources that relate to predicate argument structures: PropBank (PB) and FrameNet (FN). PB is a 300,000 word corpus annotated with predicative information on top of the Penn Treebank 2 Wall Street Journal texts. For any given predicate, the expected arguments are labeled sequentially from Arg 0 to Arg 9, ArgA and ArgM. The Figure 1 shows an example of the PB predicate annotation. Predicates in PB are only embodied by verbs whereas most of the times Arg 0 is the *subject*, Arg 1 is the *direct object* and ArgM may indicate *locations*, as in our example.

FrameNet also describes predicate/argument structures but for this purpose it uses richer semantic structures called frames. These latter are schematic representations of situations involving various participants, properties and roles, in which a word may be typically used. Frame elements or semantic roles are arguments of target words that can be verbs or nouns or adjectives. In FrameNet, the argument names are local to the target frames. For example, assuming that *attach* is the target word and *Attaching* is the target frame, a typical sentence annotation is the following.

[$_{Agent}$ They] attach$_{Tgt}$ [$_{Item}$ themselves] [$_{Connector}$ with their mouthparts] and then release a digestive enzyme secretion which eats into the skin.

Several machine learning approaches for argument identification and classification have been developed, e.g. (Gildea and Jurasfky, 2002; Gildea and Palmer, ; Gildea and Hockenmaier, 2003; Pradhan et al., 2004). Their common characteristic is the adoption of feature spaces that model predicate-argument structures in a flat feature representation. In the next section we present the common parse tree-based approach to this problem.

2.1 Predicate Argument Extraction

Given a sentence in natural language, all the predicates associated with the verbs have to be identified along with their arguments. This problem can be divided into two subtasks: (a) the detection of the target argument boundaries, i.e. all its compounding words, and (b) the classification of the argument type, e.g. *Arg0* or *ArgM* in PropBank or *Agent* and *Goal* in FrameNet.

The standard approach to learn both the detection and the classification of predicate arguments is summarized by the following steps:

1. Given a sentence from the *training-set*, generate a full syntactic parse-tree;
2. let $\mathcal{P}$ and $\mathcal{A}$ be the set of predicates and the set of parse-tree nodes (i.e. the potential arguments), respectively;
3. for each pair $<p, a> \in \mathcal{P} \times \mathcal{A}$:
   - extract the feature representation set, $F_{p,a}$;
   - if the subtree rooted in a covers exactly the words of one argument of p, put $F_{p,a}$ in T^+ (positive examples), otherwise put it in T^- (negative examples).

For instance, in Figure 1, for each combination of the predicate *rent* with the nodes N, S, VP, V, NP, PP, D or IN the instances $F_{rent,a}$ are generated. In case the node a exactly covers "Paul", "a room" or "in Boston", it will be a positive instance otherwise it will be a negative one, e.g. $F_{rent,IN}$.

The T^+ and T^- sets can be re-organized as positive $T^+_{arg_i}$ and negative $T^-_{arg_i}$ examples for each argument i. In this way, an individual ONE-vs-ALL classifier for each argument i can be trained. We adopted this solution as it is simple and effective (Pradhan et al., 2004). In the classification phase, given a sentence of the *test-set*, all its $F_{p,a}$ are generated and classified by each individual classifier C_i. As a final decision, we select the argument associated with the maximum value among the scores provided by the individual classifiers.

2.2 Standard feature space

The discovery of relevant features is, as usual, a complex task, nevertheless, there is a common consensus on the basic features that should be adopted. These standard features, firstly proposed in (Gildea and Jurasfky, 2002), refer to a flat information derived from parse trees, i.e. *Phrase Type*, *Predicate Word*, *Head Word*, *Governing Category*, *Position* and *Voice*. For example, the *Phrase Type* indicates the syntactic type of the phrase labeled as a predicate argument, e.g. NP for Arg_1 in Figure 1. The *Parse Tree Path* contains the path in the parse tree between the predicate and the argument phrase, expressed as a sequence of non-terminal labels linked by direction (up or down) symbols, e.g. V $\uparrow$ VP $\downarrow$ NP for Arg_1 in Figure 1. The *Predicate Word* is the surface form of the verbal predicate, e.g. *rent* for all arguments.

In the next section we describe the SVM approach and the basic kernel theory for the predicate argument classification.

2.3 Learning with Support Vector Machines

Given a vector space in $\Re^n$ and a set of positive and negative points, SVMs classify vectors according to a separating hyperplane, $H(\vec{x}) = \vec{w} \times \vec{x} + b = 0$, where $\vec{w} \in \Re^n$ and $b \in \Re$ are learned by applying the *Structural Risk Minimization principle* (Vapnik, 1995).

To apply the SVM algorithm to Predicate Argument Classification, we need a function $\phi : \mathcal{F} \rightarrow \Re^n$ to map our features space $\mathcal{F} = \{f_1, .., f_{|\mathcal{F}|}\}$ and our predicate/argument pair representation, $F_{p,a} = F_z$, into $\Re^n$, such that:

$$F_z \rightarrow \phi(F_z) = (\phi_1(F_z), .., \phi_n(F_z))$$

From the kernel theory we have that:

$$H(\vec{x}) = \Big(\sum_{i=1..l} \alpha_i \vec{x}_i \Big) \cdot \vec{x} + b =$$

$$\sum_{i=1..l} \alpha_i \vec{x}_i \cdot \vec{x} + b = \sum_{i=1..l} \alpha_i \phi(F_i) \cdot \phi(F_z) + b.$$

where, $F_i \; \forall i \in \{1, .., l\}$ are the training instances and the product $K_T(F_i, F_z) =<\phi(F_i) \cdot \phi(F_z)>$ is the kernel function associated with the mapping ϕ.

The simplest mapping that we can apply is $\phi(F_z) = \vec{z} = (z_1, ..., z_n)$ where $z_i = 1$ *if* $f_i \in F_z$ and $z_i = 0$ otherwise, i.e. the characteristic vector of the set F_z with respect to $\mathcal{F}$. If we choose the scalar product as a kernel function we obtain the linear kernel $K_L(F_x, F_z) = \vec{x} \cdot \vec{z}$.

Another function that has shown high accuracy for the predicate argument classification (Pradhan et al., 2004) is the polynomial kernel:

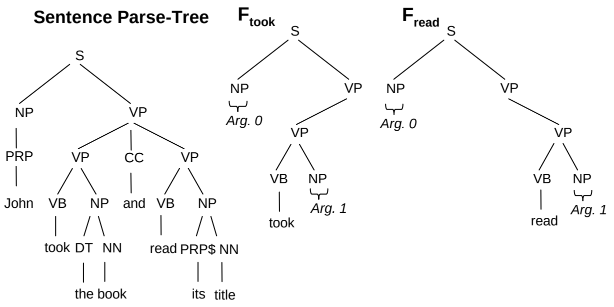


Figure 2: Subcategorization frame structure for two predicate argument structures.

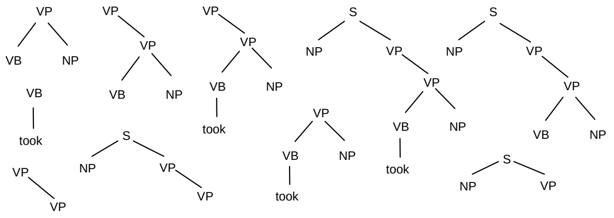


Figure 3: All 10 valid fragments of the SCFS associated with the arguments of F_{took} of Figure 2.

$K_{Poly}(F_x, F_z) = (c + \vec{x} \cdot \vec{z})^d$, where c is a constant and d is the degree of the polynom.

The interesting property is that we do not need to evaluate the ϕ function to compute the above vector; only the $K(\vec{x}, \vec{z})$ values are required. This allows us to define efficient classifiers in a huge (possible infinite) feature set, provided that the kernel is processed in an efficient way. In the next section, we introduce the convolution kernel that we used to represent subcategorization structures.

3 Subcategorization Frame Kernel (SK)

The convolution kernel that we have experimented was devised in (Moschitti, 2004) and is characterized by two aspects: the semantic space of the subcategorization structures and the kernel function that measure their similarities.

3.1 Subcategorization Frame Structure (SCFS)

We consider the predicate argument structures annotated in PropBank or FrameNet as our semantic space. As we assume that semantic structures are correlated to syntactic structures, we used a kernel that selects semantic information according to the syntactic structure of a predicate. The subparse tree which describes the subcategorization frame of the target verbal predicate defines the target Subcategorization Frame Structure (SCFS). For example, Figure 2 shows the parse tree of the sentence `"John took the book and read its title"` together with two SCFS structures, F_{took} and F_{read} associated with the two predicates *took* and *read*, respectively. Note that SCFS includes also the external argument (i.e. the subject) although some linguistic theories do not consider it being part of the SCFs.

Once the semantic representation is defined, we need to design a tree kernel function to estimate the similarity between our objects.

3.2 The tree kernel function

The main idea of tree kernels is to model a $K(T_1, T_2)$ function which computes the number of the common substructures between two trees T_1 and T_2. For example, Figure 3 shows all the fragments of the argument structure F_{took} (see Figure 2) which will be matched against the fragment of another SCFS.

Given the set of fragments $\{f_1, f_2, ..\} = \mathcal{F}$ extracted from all SCFSs of the training set, we define the indicator function $I_i(n)$ which is equal 1 if the target f_i is rooted at node n and 0 otherwise. It follows that:

$$K(T_1, T_2) = \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} \Delta(n_1, n_2) \qquad (1)$$

where N_{T_1} and N_{T_2} are the sets of the T_1's and T_2's nodes, respectively and $\Delta(n_1, n_2) = \sum_{i=1}^{|\mathcal{F}|} I_i(n_1) I_i(n_2)$. This latter is equal to the number of common fragments rooted in the n_1 and n_2 nodes. We can compute Δ as follows:

1. if the productions at n_1 and n_2 are different then $\Delta(n_1, n_2) = 0$;
2. if the productions at n_1 and n_2 are the same, and n_1 and n_2 have only leaf children (i.e. they are pre-terminals symbols) then $\Delta(n_1, n_2) = 1$;
3. if the productions at n_1 and n_2 are the same, and n_1 and n_2 are not pre-terminals then

$$\Delta(n_1, n_2) = \prod_{j=1}^{nc(n_1)} (1 + \Delta(c^j_{n_1}, c^j_{n_2})) \qquad (2)$$

where $\sigma \in \{0, 1\}$, $nc(n_1)$ is the number of the children of n_1 and c^j_n is the j-th child of the node n.

Note that, as the productions are the same $nc(n_1) = nc(n_2)$.

The above kernel has the drawback of assigning higher weights to larger structures[1]. To overcome this problem we can scale the relative importance of the tree fragments using a parameter λ in the conditions 2 and 3 as follows: $\Delta(n_x, n_z) = \lambda$ and $\Delta(n_x, n_z) = \lambda \prod_{j=1}^{nc(n_x)} (\sigma + \Delta(c_{n_1}^j, c_{n_2}^j))$.

The set of fragments that belongs to SCFs are derived by human annotators according to semantic considerations, thus they generate a semantic subcategorization frame kernel (SK). We also note that SK estimates the similarity between two SCFSs by counting the number of fragments that are in common. For example, in Figure 2, $K_T(\phi(F_{took}), \phi(F_{read}))$ is quite high (i.e. 6 out 10 substructures) as the two verbs have the same syntactic realization.

In other words the fragments encode semantic information which is measured by SK. This provides the argument classifiers with important clues about the possible set of arguments suited for a target verbal predicate. To support this hypothesis the next section presents the experiments on the predicate argument type of FrameNet and ProbBank.

4 The Experiments

A clustering algorithm which uses SK would group together verbs that show a similar syntactic structure. To study the properties of such clusters we experimented SK in combination with the traditional kernel used for the predicate argument classification. As the polynomial kernel with degree 3 was shown to be the most accurate for the argument classification (Pradhan et al., 2004; Moschitti, 2004) we use it to build two kernel combinations:

- $Poly + SK$: $\frac{K_{Poly}}{|K_{Poly}|} + \gamma \frac{K_T}{|K_T|}$, i.e. the sum between the normalized polynomial kernel (see Section 2.3) and the normalized SK[2].
- $Poly \times SK$: $\frac{K_{Poly} \times K_T}{|K_{Poly}| \times |K_T|}$, i.e. the normalized product between the polynomial kernel and SK.

For the experiments we adopted two corpora PropBank (PB) and FrameNet (FN). PB, available at www.cis.upenn.edu/∼ace, is used along with the Penn TreeBank 2 (www.cis.upenn.edu/∼treebank) (Marcus et al., 1993). This corpus contains about 53,700 sentences and a fixed split between training and testing which has been used in other researches, e.g. (Pradhan et al., 2004; Gildea and Palmer,). In this split, Sections from 02 to 21 are used for training, section 23 for testing and sections 1 and 22 as development set. We considered all 12 arguments from *Arg0* to *Arg9*, *ArgA* and *ArgM* for a total of 123,918 and 7,426 arguments in the training and test sets, respectively. It is worth noting that in the experiments we used the gold standard parsing from the Penn TreeBank, thus our kernel structures are derived with high precision.

The second corpus was obtained by extracting from FrameNet (www.icsi.berkeley.edu/∼framenet/) all 24,558 sentences from 40 frames of the Senseval 3 (http://www.senseval.org) Automatic Labeling of Semantic Role task. We considered 18 of the most frequent roles for a total of 37,948 arguments[3]. Only verbs are selected to be predicates in our evaluations. Moreover, as there is no fixed split between training and testing, we randomly selected 30% of the sentences for testing and 30% for *validation-set*, respectively. Both training and testing sentences were processed using Collins' parser (Collins, 1997) to generate parse-tree automatically. This means that our shallow semantic parser for FrameNet is fully automated.

4.1 The Classification set-up

The evaluations were carried out with the SVM-light-TK software (Moschitti, 2004) available at http://ai-nlp.info.uniroma2.it/moschitti/ which encodes the tree kernels in the SVM-light software (Joachims, 1999).

The classification performance was measured using the F_1 measure[4] for the individual arguments and the accuracy for the final multi-class classifier. This latter choice allows us to compare the results

[1] With a similar aim and to have a similarity score between 0 and 1, we also apply the normalization in the kernel space, i.e. $K'(T_1, T_2) = \frac{K(T_1,T_2)}{\sqrt{K(T_1,T_1) \times K(T_2,T_2)}}$.

[2] To normalize a kernel $K(\vec{x}, \vec{z})$ we can divide it by $\sqrt{K(\vec{x}, \vec{x}) \times K(\vec{z}, \vec{z})}$.

[3] We mapped together roles having the same name

[4] F_1 assigns equal importance to Precision P and Recall R, i.e. $F_1 = \frac{2P \times R}{P+R}$.

with previous literature works, e.g. (Gildea and Jurasfky, 2002; Pradhan et al., 2004; Gildea and Palmer,).

For the evaluation of SVMs, we used the default regularization parameter (e.g., $C = 1$ for normalized kernels) and we tried a few cost-factor values (i.e., $j \in \{1, 2, 3, 5, 7, 10, 100\}$) to adjust the rate between Precision and Recall. We chose the parameters by evaluating the SVMs using the K_{Poly} kernel (degree = 3) over the *validation-set*. Both λ (see Section 3.2) and γ parameters were evaluated in a similar way by maximizing the performance of SVM using *Poly+SK*. We found that the best values were 0.4 and 0.3, respectively.

4.2 Comparative results

To study the impact of the subcategorization frame kernel we experimented the three models $Poly$, $Poly + SK$ and $Poly \times SK$ on different training conditions.

First, we run the above models using all the verbal predicates available in the training and test sets. Tables 1 and 2 report the F_1 measure and the global accuracy for PB and FN, respectively. Column 2 shows the accuracy of $Poly$ (90.5%) which is substantially equal to the accuracy obtained in (Pradhan et al., 2004) on the same training and test sets with the same SVM model. Columns 3 and 4 show that the kernel combinations $Poly + SK$ and $Poly \times SK$ remarkably improve $Poly$ accuracy, i.e. 2.7% (93.2% vs. 90.5%) whereas on FN only $Poly + SK$ produces a small accuracy increase, i.e. 0.7% (86.2% vs. 85.5%).

This outcome is lower since the FN classification requires dealing with a higher variability of its semantic roles. For example, in ProbBank most of the time, the PB *Arg0* and *Arg1* corresponds to the *logical subject* and *logical direct object*, respectively. On the contrary, the FN *Cause* and *Agent* roles are often both associated with the *logical subject* and share similar syntactic realizations, making SCFS less effective to distinguish between them. Moreover, the training data available for FrameNet is smaller than that used for PropBank, thus, the tree kernel may not have enough examples to generalize, correctly.

Second, we carried out other experiments using a subset of the total verbs for training and another disjoint subset for testing. In these conditions, the impact of SK is amplified: on PB, $SK \times Poly$ outperforms *Poly* by 4.8% (86.9% vs. 82.1%), whereas, on FN, SK increases *Poly* of about 2%, i.e. 74.6% vs. 72.8%. These results suggest that (a) when test-set verbs are not observed during training, the classification task is harder, e.g. 82.1% vs. 90.5% on PB and (b) the syntactic structures of the verbs, i.e. the SCFSs, allow the SVMs to better generalize on unseen verbs.

Args	All Verbs			Disjoint Verbs		
	$Poly$	$+SK$	$\times SK$	$Poly$	$+SK$	$\times SK$
Arg0	90.8	94.6	94.7	86.8	90.9	91.1
Arg1	91.1	92.9	94.1	81.7	86.8	88.3
Arg2	80.0	77.4	82.0	49.9	49.5	47.6
Arg3	57.9	56.2	56.4	20.3	22.9	20.6
Arg4	70.5	69.6	71.1	0	0	0
ArgM	95.4	96.1	96.3	90.3	93.4	93.7
Acc.	90.5	92.4	93.2	82.1	86.3	86.9

Table 1: Kernel accuracies on PropBank.

Role	All Verbs			Disjoint Verbs		
	$Poly$	$+SK$	$\times SK$	$Poly$	$+SK$	$\times SK$
agent	91.7	94.4	94.0	82.5	84.8	84.7
cause	57.4	60.6	56.4	29.1	28.1	26.9
degree	77.1	77.2	60.9	40.6	44.6	22.6
depict.	85.8	86.2	85.9	73.6	74.0	71.2
instrum.	67.1	69.1	64.6	13.3	13.0	12.8
manner	80.5	79.7	77.7	74.8	74.3	72.3
Acc.	85.5	86.2	85.0	72.8	74.6	74.2

Table 2: Kernel accuracies on 18 FrameNet semantic roles.

To verify that the kernel representation is superior to the traditional representation we carried out an experiment using a flat feature representation of the SCFs, i.e. we used the syntactic frame feature described (Xue and Palmer, 2004) in place of SK. The result as well as other literature findings, e.g. (Pradhan et al., 2004) show an improvement on PB of about 0.7% only. Evidently flat features cannot derive the same information of a convolution kernel.

Finally, to study how the verb complexity impacts on the usefulness of SK, we carried out additional experiments with different verb sets. One dimension of complexity is the frequency of the verbs in the target corpus. Infrequent verbs are associated with predicate argument structures poorly represented in the training set thus they are more difficult to classify. Another dimension of the verb complexity is the number of different SCFs that they show in different contexts. Intuitively, the higher is the number

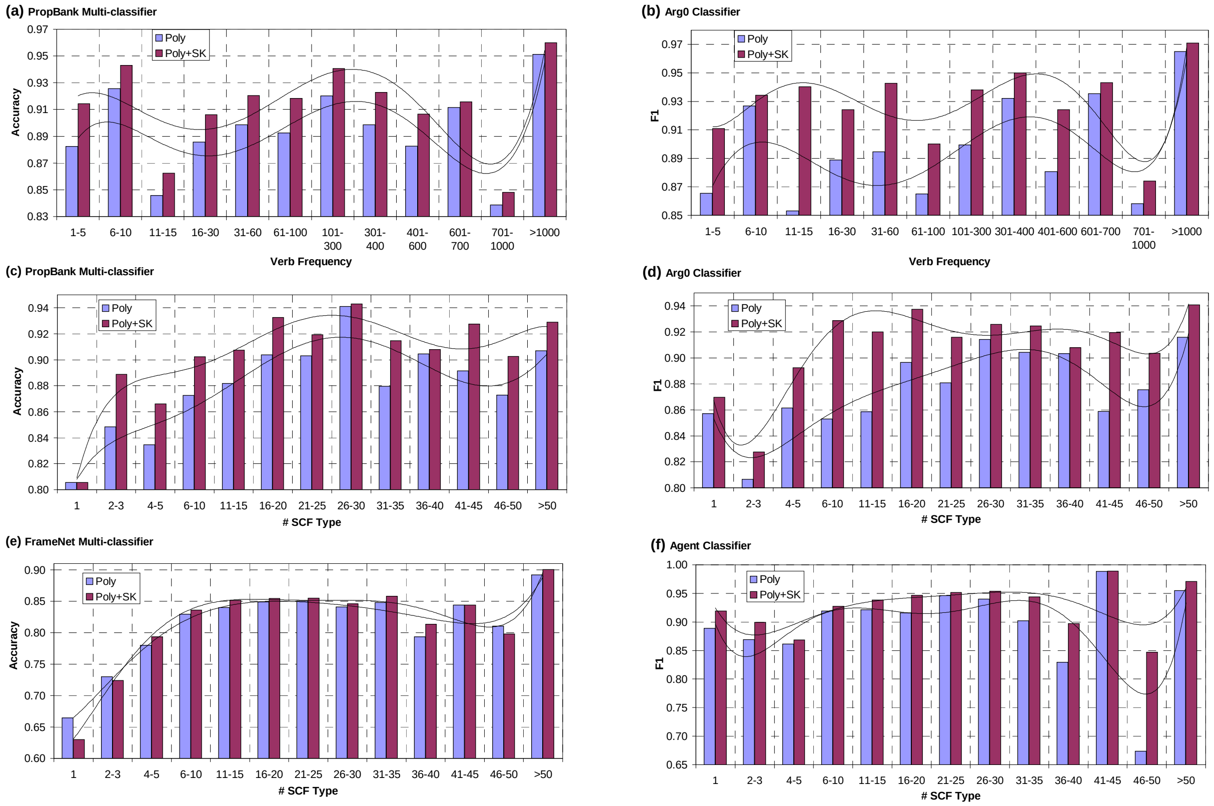


Figure 4: The impact of SCF on the classification accuracy of the semantic arguments and semantic roles according to the verb complexity.

of verb's SCF types the more difficult is the classification of its arguments.

Figure 4.a, reports the accuracy along with the trend line plot of $Poly$ and $SK + Poly$ according to subsets of different verb frequency. For example, the label 1-5 refers to the class of verbal predicates whose frequency ranges from 1 to 5. The associated accuracy is evaluated on the portions of the training and test-sets which contain only the verbs in such class. We note that SK improves $Poly$ for any verb frequency. Such improvement decreases when the frequency becomes very high, i.e. when there are many training instances that can suggest the correct classification. A similar behavior is shown in Figure 4.b where the F_1 measure for Arg0 of PB is reported.

Figures 4.c and 4.d illustrate the accuracy and the F_1 measure for all arguments and Arg0 of PB according to the number of SCF types, respectively. We observe that the Semantic Kernel does not produce any improvement on the verbs which are syntactically expressed by only one type of SCF. As the number of SCF types increases (> 1) $Poly + SK$ outperforms $Poly$ for any verb class, i.e. when the verb is *enough* complex SK always produces useful information independently of the number of the training set instances. On the one hand, a high number of verb instances reduces the complexity of the classification task. On the other hand, as the number of verb type increases the complexity of the task increases as well.

A similar behavior can be noted on the FN data (Figure 4.e) even if the not so strict correlation between syntax and semantics prevents SK to produce high improvements. Figure 4.f shows the impact of SK on the *Agent* role. We note that, the F_1 increases more than the global accuracy (Figure 4.e) as the *Agent* most of the time corresponds to Arg0. This is confirmed by the Table 2 which shows an improvement for the *Agent* of up to 2% when SK is used along with the polynomial kernel.

5 Conclusive Remarks

In this article, we used Support Vector Machines (SVMs) to deeply analyze the role of the subcategorization frame kernel (SK) in the automatic predicate argument classification of PropBank and

FrameNet corpora. To study the SK's verb classification properties we have combined it with the polynomial kernel on standard flat features.

We run the SVMs on diverse levels of task complexity. The results show that: (1) in general SK remarkably improves the classification accuracy. (2) When there are no training instances of the test-set verbs the improvement of SK is almost double. This suggests that tree kernels automatically derive features which support also a sort of back-off estimation in case of unseen verbs. (3) In all complexity conditions the structural features are in general very robust, maintaining a high improvement over the basic accuracy. (4) The semantic role classification in FrameNet is affected with more noisy data as it is based on the output of a statistical parser. As a consequence the improvement is lower. Anyway, the systematic superiority of SK suggests that it is less sensible to parsing errors than other models. This opens promising direction for a more weakly supervised application of the statistical semantic tagging supported by SK.

In summary, the extensive experimentation has shown that the SK provides information robust with respect to the complexity of the task, i.e. verbs with richer syntactic structures and sparse training data.

An important observation on the use of tree kernels has been pointed out in (Cumby and Roth, 2003). Both computational efficiency and classification accuracy can often be superior if we select the most informative tree fragments and carry out the learning in the feature space. Nevertheless, the case studied in this paper is well suited for using kernels as: (1) it is difficult to guess which fragment from an SCF should be retained and which should be discarded, (2) it may be the case that all fragments are useful as SCFs are small structures and all theirs substructures may serve as different back-off levels and (3) small structures do not heavily penalize efficiency.

Future research may be addressed to (a) the use of SK kernel to explicitly generate verb clusters and (b) the use of convolution kernels to study other linguistic phenomena: we can use tree kernels to investigate which syntactic features are suited for an unknown phenomenon.

Acknowledgement

We would like to thank the anonymous reviewers for their competence and commitment showed in the review of this paper.

References

Michael Collins. 1997. Three generative, lexicalized models for statistical parsing. In *Proceedings of the ACL'97,*Somerset, New Jersey.

Chad Cumby and Dan Roth. 2003. Kernel methods for relational learning. In *Proceedings of ICML'03*, Washington, DC, USA.

Charles J. Fillmore. 1982. Frame semantics. In *Linguistics in the Morning Calm*, pages 111–137.

Daniel Gildea and Julia Hockenmaier. 2003. Identifying semantic roles using combinatory categorial grammar. In *Proceedings of EMNLP'03*, Sapporo, Japan.

Daniel Gildea and Daniel Jurasfky. 2002. Automatic labeling of semantic roles. *Computational Linguistic*, 28(3):496–530.

Daniel Gildea and Martha Palmer. The necessity of parsing for predicate argument recognition. In *Proceedings of ACL'02*.

T. Joachims. 1999. Making large-scale SVM learning practical. In B. Sch$\ddot{o}$lkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*.

Paul Kingsbury and Martha Palmer. 2002. From Treebank to PropBank. In *Proceedings of LREC'02*.

Anna Korhonen. 2003. *Subcategorization Acquisition*. Ph.D. thesis, Techical Report UCAM-CL-TR-530. Computer Laboratory, University of Cambridge.

Beth Levin. 1993. *English Verb Classes and Alternations A Preliminary Investigation*. Chicago: University of Chicago Press.

M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. 1993. Building a large annotated corpus of english: The Penn Treebank. *Computational Linguistics*, 19:313–330.

Alessandro Moschitti. 2004. A study on convolution kernels for shallow semantic parsing. In *proceedings ACL'04*, Barcelona, Spain.

Sameer Pradhan, Kadri Hacioglu, Valeri Krugler, Wayne Ward, James H. Martin, and Daniel Jurafsky. 2005. Support vector learning for semantic argument classification. *to appear in the Machine Learning Journal*.

V. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer.

Nianwen Xue and Martha Palmer. 2004. Calibrating features for semantic role labeling. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP'04*, Barcelona, Spain, July.

Identifying Concept Attributes Using a Classifier

Massimo Poesio
University of Essex
Computer Science /
Language and Computation
poesio at essex.ac.uk

Abdulrahman Almuhareb
University of Essex
Computer Science /
Language and Computation
aalmuh at essex.ac.uk

Abstract

We developed a novel classification of concept attributes and two supervised classifiers using this classification to identify concept attributes from candidate attributes extracted from the Web. Our binary (attribute / non-attribute) classifier achieves an accuracy of 81.82% whereas our 5-way classifier achieves 80.35%.

1 Introduction

The assumption that concept **attributes** and, more in general, **features**[1] are an important aspect of conceptual representation is widespread in all disciplines involved with conceptual representations, from Artificial Intelligence / Knowledge Representation (starting with at least (Woods, 1975) and down to (Baader et al, 2003)), Linguistics (e.g., in the theories of the lexicon based on typed feature structures and/or Pustejovsky's Generative Lexicon theory: (Pustejovsky 1995)) and Psychology (Murphy 2002, Vinson et al 2003). This being the case, it is surprising how little attention has been devoted to this aspect of lexical representation in work on large-scale lexical semantics in Computational Linguistics. The most extensive resource at our disposal, WordNet (Fellbaum, 1998) contains very little information that would be considered as being about 'attributes'—only information about parts, not about qualities such as **height**, or even to the values of such attributes in the adjective network—and this information is still very sparse. On the other hand, the only work on the extraction of lexical semantic relations we are aware of has concentrated on the type of relations found in WordNet: hyponymy (Hearst, 1998; Caraballo, 1999) and meronymy (Berland and Charniak, 1999; Poesio et al, 2002).[2]

The work discussed here could be perhaps best described as an example of **empirical ontology**: using linguistics and philosophical ideas to improve the results of empirical work on lexical / ontology acquisition, and vice versa, using findings from empirical analysis to question some of the assumptions of theoretical work on ontology and the lexicon. Specifically, we discuss work on the acquisition of (nominal) concept attributes whose goal is twofold: on the one hand, to clarify the notion of 'attribute' and its role in lexical semantics, if any; on the other, to develop methods to acquire such information automatically (e.g., to supplement WordNet).

The structure of the paper is as follows. After a short review of relevant literature on extracting semantic relations and on attributes in the lexicon, we discuss our classification of attributes, followed by the features we used to classify them. We then discuss our training methods and the results we achieved.

[1] The term **attribute** is used informally here to indicate the type of relational information about concepts that is expressed using so-called **roles** in Description Logics (Baader et al, 2003)—i.e., excluding **IS-A** style information (that cars are vehicles, for instance). It is meant to be a more restrictive term than the term **feature**, often used to indicate any property of concepts, particularly in Psychology. We are carrying out a systematic analysis of the sets of features used in work such as (Vinson et al, 2003) (see Discussion).

[2] In work on the acquisition of lexical information about verbs there has been some work on the acquisition of thematic roles, (e.g., Merlo and Stevenson, 2001).

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 18–27,
Ann Arbor, June 2005.

2 Background

2.1 Using Patterns to Extract Semantic Relations

The work discussed here belongs to a line of research attempting to acquire information about lexical and other semantic relations other than similarity / synonymy by identifying **syntactic constructions** that are often (but not always!) used to express such relations. The earliest work of this type we are aware of is the work by Hearst (1998) on acquiring information about hyponymy (= **IS-A** links) by searching for instances of patterns such as

```
NP {, NP}* or other NP
```

(as in, e.g., *bruises …. broken bones and other INJURIES*). A similar approach was used by Berland and Charniak (1999) and Poesio et al (2002) to extract information about **part-of** relations using patterns such as

```
the N of the N is ….
```

(as in *the wheel of the CAR is*) and by Girju and Moldovan (2002) and Sanchez-Graillet and Poesio (2004) to extract causal relations. In previous work (Almuhareb and Poesio, 2004) we used this same approach to extract attributes, using the pattern

```
“the * of the C [is|was]”
```

(suggested by, e.g., (Woods, 1975) as a test for ‘attributehood’) to search for attributes of concept C in the Web, using the Google API. Although the information extracted this way proved a useful addition to our lexical representations from a clustering perspective, from the point of view of lexicon building this approach results in too many false positives, as very few syntactic constructions are used to express exclusively one type of semantic relation. For example, the ‘attributes’ of **deer** extracted using the text pattern above include “the *majority* of the deer,” “the *lake* of the deer,” and “the *picture* of the deer.” Girju and Moldovan (2002) addressed the problem of false positives for causal relations by developing WordNet-based *filters* to remove unlikely candidates. In this work, we developed a semantic filter for attributes based on a linguistic theory of attributes which does not rely on WordNet except as a source of morphological information (see below).

2.2 Two Theories of Attributes

The earliest attempt to classify attributes and other properties of substances we are aware of goes back to Aristotle, e.g., in *Categories*,[3] but our classification of attributes was inspired primarily by the work of Pustejovsky (1995) and Guarino (e.g., (1992)). According to Pustejovsky’s *Generative Lexicon* theory (1995), an integral part of a lexical entry is its **Qualia Structure**, which consists of four ‘roles’:[4] the **Formal Role**, specifying what type of object it is: e.g., in the case of a book, that it has a shape, a color, etc.; the **Constitutive Role**, specifying the stuff and parts that it consists of (e.g., in the case of a book, that it is made of paper, it has chapters and an index, etc.); the **Telic Role**, specifying the purpose of the object (e.g., in the case of a book, *reading*); and the **Agentive Role**, specifying how the object was created (e.g., in the case of a book, by *writing*).

Guarino (1992) argues that there are two types of attributes: **relational** and **non-relational**. Relational attributes include **qualities** such as *color* and *position*, and **relational social roles** such as *son* and *spouse*. Non-relational attributes include **parts** such as *wheel* and *engine*. Activities are not viewed as attributes in Guarino’s classification.

3 Attribute Extraction and Classification

The goal of this work is to identify genuine attributes by classifying candidate attributes collected using text patterns as discussed in (Almuhareb and Poesio, 2004) according to a scheme inspired by those proposed by Guarino and Pustejovsky.

The scheme we used to classify the training data in the experiment discussed below consists of six categories:

- **Qualities**: Analogous to Guarino’s qualities and Pustejovsky’s formal ‘role’. (E.g., “the *color* of the car”.)
- **Parts:** Related to Guarino’s non-relational attributes and Pustejovsky’s constitutive ‘roles’. (E.g., “the *hood* of the car”).
- **Related-Objects:** A new category introduced to cover the numerous physical objects which are ‘related’ to an object but are not part of it—e.g., “the *track* of the deer”.

[3] E.g., http://plato.stanford.edu/entries/substance. Thanks to one of the referees for drawing our attention to this.

[4] ‘Facets’ would be perhaps a more appropriate term to avoid confusions with the use of the term ‘role’ in Knowledge Representation.

- **Activities:** These include both the types of activities which are part of Pustejovsky's telic 'role' and those which would be included in his agentive 'role'. (E.g., "the *repairing* of the car".)
- **Related-Agents:** For the activities in which the concept in question is acted upon, the agent of the activity: e.g., "the *writer* of the book", "the *driver* of the car".
- **Non-Attributes:** This category covers the cases in which the construction "the N of the N" expresses other semantic relations, as in: "the *last* of the deer", "the *majority* of the deer," "the *lake* of the deer," and "in the *case* of the deer".

We will quickly add that (i) we do not view this classification as definitive—in fact, we already collapsed the classes 'part' and 'related objects' in the experiments discussed below—and (ii) not all of these distinctions are very easy even for human judges to do. For example, *design*, as an attribute of a *car*, can be judged to be a quality if we think of it as taking values such as *modern* and *standard*; on the other hand, *design* might also be viewed as an activity in other contexts discussing the designing process. Another type of difficulty is that a given attribute may express different things for different objects. For example, *introduction* is a part of a *book*, and an activity for a *product*. An additional difficulty results from the strong similarity between parts and related-objects. For example, "key" is a related-object to a *car* but it is not part of it. We will return to this issue and to agreement on this classification scheme when discussing the experiment.

One difference from previous work is that we use additional linguistic constructions to extract candidate attributes. The construction "the X of the Y is" used in our previous work is only one example of genitive construction. Quirk *et al* (1985) list eight types of genitives in English, four of which are useful for our purposes:

- *Possessive Genitive*: used to express qualities, parts, related-objects, and related-agents.
- *Genitive of Measure*: used to express qualities.
- *Subjective & Objective Genitives*: used to express activities.

We used all of these constructions in the work discussed here.

4 Information Used to Classify Attributes

Our attribute classifier uses four types of information: *morphological information*, an *attribute model*, a *question model*, and an *attributive-usage model*. In this section we discuss how this information is automatically computed.

4.1 Morphological Information

Our use of morphological information is based on the noun classification scheme proposed by Dixon (1991). According to Dixon, derivational morphology provides some information about attribute type. Parts are concrete objects and almost all of them are expressed using basic noun roots (i.e., not derived from adjectives or verbs). Most of qualities and properties are either basic noun roots or derived from adjectives. Finally, activities are mostly nouns derived from verbs. Although these rules only have a heuristic value, we found that morphologically based heuristics did provide useful cues when used in combination with the other types of information discussed below.

As we are not aware of any publicly available software performing automatic derivational morphology, we developed our own (and very basic) heuristic methods. The techniques we used involve using information from WordNet, suffix-checking, and a POS tagger.

WordNet was used to find nouns that are derived from verbs and to filter out words that are not in the noun database. Nouns in WordNet are linked to their derivationally related verbs, but there is no indication about which is derived from which. We use a heuristic based on length to decide this: the system checks if the noun contains more letters than the most similar related verb. If this is the case, then the noun is judged to be derived from the verb. If the same word is used both as a noun and as a verb, then we check the usage familiarity of the word, which can also be found in WordNet. If the word is used more as a verb and the verbal usage is not rare, then again the system treats the noun as derived from the verb.

To find nouns that are derived from adjectives we used simple heuristics based on suffix-checking. (This was also done by Berland and Charniak (1999).) All words that end with "*ity*" or "*ness*" are considered to be derived from adjectives. A noun not found to be derived from a verb or an adjective is assumed to be a basic noun root.

In addition to derivational morphology, we used the Brill tagger (Brill, 1995) to filter out adjectives and other types of words that can occasionally be used as nouns such as *better*, *first*, and *whole* before training. Only nouns, base form verbs, and gerund form verbs were kept in the candidate attribute list.

4.2 Clustering Attributes

Attributes are themselves concepts, at least in the sense that they have their own attributes: for example, a part of a car, such as a wheel, has its own parts (the tyre) its qualities (weight, diameter) etc. This observation suggests that it should be possible to find similar attributes in an unsupervised fashion by looking at their attributes, just as we did earlier for concepts (Almuhareb and Poesio, 2004). In order to do this, we used our text patterns for finding attributes to collect from the Web up to 500 pattern instances for each of the candidate attributes. The collected data were used to build a vectorial representation of attributes as done in (Almuhareb and Poesio, 2004). We then used CLUTO (Karypis, 2002) to cluster attributes using these vectorial representations. In a first round of experiments we found that the classes 'parts' and 'related objects' were difficult to differentiate, and therefore we merged them. The final model clusters candidate attributes into five classes: activities, parts & related-objects, qualities, related-agents, and non-attributes. This classification was used as one of the input features in our supervised classifier for attributes.

We also developed a measure to identify particularly distinctive 'attributes of attributes'—attributes which have a strong tendency to occur primarily with attributes (or any concept) of a given class—which has proven to work pretty well. This measure, which we call *Uniqueness*, actually is the product of two factors: the degree of uniqueness proper, i.e., the probability $P(class_i \mid attribute_j)$ that an attribute (or, in fact, any other noun) will belong to class i given than it has attribute j; and a measure of 'definitional power' –the probability $P(attribute_j \mid class_i)$ that a concept belonging to a given class will have a certain attribute. Using MLE to estimate these probabilities, the degree of uniqueness of $attributes_j$ of $class_i$ is computed as follows:

$$Uniqueness_{i,j} = \frac{C(class_i, attribute_j)^2}{n_i \times C(attribute_j)}$$

where $\boldsymbol{n_i}$ is the number of concepts in $class_i$. $\boldsymbol{C}$ is a count function that counts concepts that are associated with the given attribute. Uniqueness ranges from 0 to 1.

Table 1 shows the 10 most distinctive attributes for each of the five attribute classes, as determined by the Uniqueness measure just introduced, for the 1,155 candidate attributes in the training data for the experiment discussed below.

Class	Top 10 Distinctive Attributes
Related-Agent (0.39)	identity, hands, duty, consent, responsibility, part, attention, voice, death, job
Part & Related-Object (0.40)	inside, shape, top, outside, surface, bottom, center, front, size, interior
Activity (0.29)	time, result, process, results, timing, date, effect, beginning, cause, purpose
Quality (0.23)	measure, basis, determination, question, extent, issue, measurement, light, result, increase
Non-Attribute (0.18)	content, value, rest, nature, meaning, format, interpretation, essence, size, source

Table 1: Top 10 distinctive attributes of the five classes of candidate attributes. Average distinctiveness (uniqueness) for the top 10 attributes is shown between parentheses

Most of the top 10 attributes of related-agents, parts & related-objects, and activities are genuinely distinctive attributes for such classes. Thus, attributes of related-agents reflect the 'intentionality' aspect typical of members of this class: *identity*, *duty*, and *responsibility*. Attributes of parts are common attributes of physical objects (e.g., *inside*, *shape*). Most attributes of activities have to do with temporal properties and causal structure: e.g., *beginning*, *cause*. The 'distinctive' attributes of the quality class are less distinctive, but four such attributes (*measure*, *extent*, *measurement*, and *increase*) are related to values since many of the qualities can have different values (e.g., *small* and *large* for the quality *size*). There are however several attributes in common between these classes of attributes, emphasizing yet again how some of these distinctions at least are not completely clear cut: e.g., *result*, in common between activities and qualities (two classes which are sometimes difficult to distinguish). Finally, as one would expect, the attributes of the non-attribute class are not really distinctive: their average uniqueness score is the lowest. This is because 'non-attribute' is a heterogeneous class.

4.3 The Question Model

Certain types of attributes can only be used when asking certain types of questions. For example, it is possible to ask "*What is the color of the car?*" but not "∗*When is the color of the car?*".

We created a text pattern for each type of question and used these patterns to search the Web and collect counts of occurrences of particular questions. An example of such patterns would be:

- *"what is|are the A of the"*

where *A* is the candidate attribute under investigation. Patterns for *who*, *when*, *where*, and *how* are similar.

After collecting occurrence frequencies for all the candidate attributes, we transform these counts into weights using the *t*-test weighting function as done for all of our counts, using the following formula from Manning and Schuetze (1999):

$$t_{i,j} \approx \frac{\frac{C(question_i, attribute_j)}{N} - \frac{C(question_i) \times C(attribute_j)}{\mathrm{N}^2}}{\sqrt{\frac{C(question_i, attribute_j)}{N^2}}}$$

where $\boldsymbol{N}$ is the total number of relations, and $\boldsymbol{C}$ is a count function.

Table 2 shows the 10 most frequent attributes for each question type. This data was collected using a more restricted form of the question patterns and a varying number of instances for each type of questions. The restricted form includes a question mark at the end of the phrase and was used to improve the precision. For example, the *what*-pattern would be *"what is the * of the *?"*.

Question	Top 10 Attributes
what	purpose, name, nature, role, cost, function, significance, size, source, status
who	author, owner, head, leader, president, sponsor, god, lord, father, king
where	rest, location, house, fury, word, edge, center, end, ark, voice
how	quality, rest, pace, level, length, morale, performance, content, organization, cleanliness
when	end, day, time, beginning, date, onset, running, birthday, fast, opening

Table 2: Frequent attributes for each question type

Instances of the *what*-pattern are frequent in the Web: the Google count was more than 2,000,000 for a query issued in mid 2004. The *who*-pattern is next in terms of occurrence, with about 350,000 instances. The *when*-pattern is the most infrequent pattern, about 5,300 instances.

The counts broadly reflected our intuitions about the use of such questions. W*hat*-questions are mainly used with qualities, whereas *who*-questions are used with related-agents. Attributes occurring with *when*-questions have some temporal aspects; attributes occurring with *how*-questions are mostly qualities and activities, and attributes in *where*-questions are of different types but some are related to locations. Parts usually do not occur with these types of questions.

4.4 Attributive Use

Finally, we exploited the fact that certain types of attributes are used more in language as concepts rather than as attributes. For instance, it is more common to encounter the phrase *"the size of the* ∗*"* than *"the* ∗ *of the size"*. On the other hand, it is more common to encounter the phrase *"the * of the window"* than *"the window of the *"*. Generally speaking, parts, related-objects, and related-agents are more likely to have more attributes than qualities and activities. We used the two patterns "the * of the A" and "the A of the *" to collect Google counts for all of the candidate attributes. These counts were also weighted using the *t*-test as in the question model.

Table 3 illustrates the attributive and conceptual usage for each attribute class using a training data of 1,155 attributes. The usage averages confirm the initial assumption.

Attribute Class	Average *T*-Test Score	
	Conceptual	Attributive
Parts & Related-Objects	18.81	3.00
Non-Attributes	13.29	11.07
Related-Agents	12.15	2.54
Activities	3.22	5.08
Qualities	0.23	17.09

Table 3: Conceptual and attributive usage averages for each attribute class

5 The Experiment

We trained two classifiers: a 2-way classifier that simply classifies candidate attributes into attributes and non-attributes, and a 5-way classifier that classifies candidate attributes into activities, parts & related-objects, qualities, related-agents, and non-attributes. These classifiers were trained using decision trees algorithm (J48) from WEKA (Witten and Frank, 1999).

Feature	election	abdomen	acidity	creator	problem
Cluster Id	1	2	4	0	3
What	0.00	0.00	0.00	0.00	3.80
When	2.62	0.00	0.00	0.00	0.00
Where	0.78	0.94	0.00	0.00	0.00
Who	0.00	0.00	0.00	30.28	0.00
How	2.05	0.00	1.54	0.00	2.61
Conceptual	38.16	20.15	0.00	0.00	135.40
Attributive	0.00	0.00	10.22	1.60	0.00
Morph	DV	BN	DA	DV	BN
Attribute Class (Output)	Activity	Part	Quality	Related Agent	Non-Attribute

Table 4: Five examples of training instances. The values for **morph** are as follows: DV: derived from verb; BN: basic noun; DA: derived from adjective

Our training and testing material was acquired as follows. We started from the 24,178 candidate attributes collected for the concepts in the balanced concept dataset we recently developed (Almuhareb and Poesio, 2005). We threw out every candidate attribute with a Google frequency less than 20; this reduced the number of candidate attributes to 4,728. We then removed words other than nouns and gerunds as discussed above, obtaining 4,296 candidate attributes.

The four types of input features for this filtered set of candidate attributes were computed as discussed in the previous section. The best results were obtained using all of these features. A training set of 1,155 candidate attributes was selected and hand-classified (see below for agreement figures). We tried to include enough samples for each attribute class in the training set. Table 4 shows the input features for five different training examples, one for each attribute class.

6 Evaluation

For a qualitative idea of the behavior of our classifier, the best attributes for some concepts are listed in Appendix A. We concentrate here on quantitative analyses.

6.1 Classifier Evaluation 1: Cross-Validation

Our two classifiers were evaluated, first of all, using 10-fold cross-validation. The 2-way classifier correctly classified 81.82% of the candidate attributes (the baseline accuracy is 80.61%). The 5-way classifier correctly classified 80.35% of the attributes (the baseline accuracy is 23.55%). The precision / recall results are shown in Table 5.

Attribute Class	P	R	F
2-Way Classifier			
Attribute	0.854	0.934	0.892
Non-Attribute	0.551	0.335	0.417
5-Way Classifier			
Related-Agent	0.930	0.970	0.950
Part & Related-Object	0.842	0.882	0.862
Activity	0.822	0.878	0.849
Quality	0.799	0.821	0.810
Non-Attribute	0.602	0.487	0.538

Table 5: Cross-validation results for the two attribute classifiers

As it can be seen from Table 5, both classifiers achieve good *F* values for all classes except for the non-attribute class: *F*-measures range from 81% to 95%. With the 2-way classifier, the valid attribute class has an *F*-measure of 89.2%. With the 5-way classifier, **related-agent** is the most accurate class (*F* = 95%) followed by **part & related-object**, **activity**, and **quality** (86.2%, 84.9%, and 81.0%, respectively). With **non-attribute,** however, we find an *F* of 41.7% in the 2-way classification, and 53.8% in the 5-way classification. This suggests that the best strategy for lexicon building would be to use these classifiers to 'find' attributes rather than 'filter' non-attributes.

6.2 Classifier Evaluation 2: Human Judges

Next, we evaluated the accuracy of the attribute classifiers against two human judges (the authors). We randomly selected a concept from each of the 21 classes in the balanced dataset. Next, we used the classifiers to classify the 20 best candidate attributes of each concept, as determined by their *t*-test scores. Then, the judges decided if the assigned classes are correct or not. For the 5-way classifier, the judges also assigned the correct class if the automatic assigned class is incorrect.

After a preliminary examination we decided not to consider two troublesome concepts: *constructor* and *future*. The reason for eliminating c*onstructor* is that we discovered it is ambiguous: in addition to the sense of 'a person who builds things', we discovered that *constructor* is used widely in the Web as a name for a fundamental *method* in object oriented programming languages such as Java. Most of the best candidate attributes (e.g., *call*, *arguments*, *code*, and *version*) related to the latter sense, that doesn't exist in WordNet. Our system is currently not able to do word sense discrimination, but we are currently working on this issue. The reason for ignoring the concept *future* was that this word is most commonly used as a modifier in phrases such as: "*the car of the future*", and "*the office of the future*", and that all of the best candidate attributes occurred in this type of construction. This reduced the number of evaluated concepts to 19.

According to the judges, the 2-way classifier was on average able to correctly assign attribute classes for 82.57% of the candidate attributes. This is very close to its performance in evaluation 1. The results using the *F*-measure reveal similar results too. Table 6 shows the results of the two classifiers based on the precision and recall measures.

According to the judges, the 5-way classifier correctly classified 68.72% on average. This performance is good but not as good as its performance in evaluation 1 (80.35%). The decrease in the performance was also shown in the *F*-measure. The *F*-measure ranges from 0.712 to 0.839 excluding the non-attribute class.

Attribute Class	**P**	**R**	**F**
2-Way Classifier			
Attribute	0.928	0.872	0.899
Non-Attribute	0.311	0.459	0.369
5-Way Classifier			
Related-Agent	0.813	0.868	0.839
Part & Related-Object	0.814	0.753	0.781
Activity	0.870	0.602	0.712
Quality	0.821	0.658	0.730
Non-Attribute	0.308	0.632	0.414

Table 6: Evaluation against human judges results for the two classifiers

An important question when using human judges is the degree of agreement among them. The *K*-statistic was used to measure this agreement. The values of *K* are shown in Table 7. In the 2-way classification, the judges agreed on 89.84% of the cases. On the other hand, the *K*-statistic for this classification task is 0.452. This indicates that part of this strong agreement is because that the majority of the candidate attributes are valid attributes. It also shows the difficulty of identifying non-attributes even for human judges. In the 5-way classification, the two judges have a high level of agreement; Kappa statistic is 0.749. The judges and the 5-way classifier agreed on 63.71% of the cases.

Description	**2-Way**	**5-Way**
Human Judges	89.84%	80.69%
Human Judges (Kappa)	0.452	0.749
Human Judges & Classifier	78.36%	63.71%

Table 7: Level of agreement between the human judges and the classifiers

6.3 Re-Clustering the Balanced Dataset

Finally, we looked at whether using the classifiers results in a better lexical description for the purposes of clustering (Almuhareb and Poesio, 2004). In Table 8 we show the results obtained using the output of the 2-way classifier to re-cluster the 402 concepts of our balanced dataset, comparing these results with those obtained using all attributes (first column) and all attributes that remain after frequency cutoff and POS filtering (column 2). The results are based on the CLUTO evaluation measures: *Purity* (which measures the degree of cohesion of the clusters obtained) and *Entropy*. The purity and entropy formulas are shown in Table 9.

Description	All Candidate Attributes	Filtered Candidate Attributes	2-Way Attributes
Purity	0.657	0.672	0.693
Entropy	0.335	0.319	0.302
Vector Size	24,178	4,296	3,824

Table 8: Results of re-clustering concepts using different sets of attributes

Clustering the concepts using only filtered candidate attributes improved the clustering purity from 0.657 to 0.672. This improvement in purity is not significant. However, clustering using only the attributes sanctioned by the 2-way classifier improved the purity further to 0.693, and this improvement in purity from the initial purity was significant ($t = 2.646$, $df = 801$, $p < 0.05$).

	Entropy	Purity
Single Cluster	$E(S_r) = -\frac{1}{\log q}\sum_{i=1}^{q}\frac{n_r^i}{n_r}\log\frac{n_r^i}{n_r}$	$P(S_r) = \frac{1}{n_r}\max_i(n_r^i)$
Over-all	$Entropy = \sum_{r=1}^{k}\frac{n_r}{n}E(S_r)$	$Purity = \sum_{r=1}^{k}\frac{n_r}{n}P(S_r)$

Table 9: Entropy and Purity in CLUTO.
S_r is a cluster, n_r is the size of the cluster, q is the number of classes, n^i_r is the number of concepts from the *i*th class that were assigned to the *r*th cluster, n is the number of concepts, and k is the number of clusters.

7 Discussion and Conclusions

The lexicon does not simply contain information about synonymy and hyponymy relations; it also contains information about the attributes of the concepts expressed by senses, as in Qualia structures. In previous work, we developed techniques for mining candidate attributes from the Web; in this paper we presented a method for improving the quality of attributes thus extracted, based on a classification for attributes derived from work in linguistics and philosophy, and a classifier that automatically tags candidate attributes with such classes. Both the 2-way and the 5-way classifiers achieve good precision and recall. Our work also reveals, however, that the notion of attribute is not fully understood. On the one hand, that attribute judgments are not always easy for humans even given a scheme; on the other hand, the results for certain types of attributes, especially activities and qualities, could certainly be improved. We also found that whereas attributes of physical objects are relatively easy to classify, the attributes of other types of concepts are harder –particularly with activities. (See the Appendix for examples.) Our longer term goal is thus to further clarify the notion of attribute, possibly refining our classification scheme, in collaboration with linguists, philosophers, and psycholinguists. One comparison we are particularly interested in pursuing at the moment is that with feature lists used by psychologist, for whom knowledge representation is entirely concept-based, and virtually every property of a concept counts as an attribute, including properties that would be viewed as **IS-A** links and what would be considered a value. Is it possible to make a principled, yet cognitively based distinction?

Acknowledgments

Abdulrahman Almuhareb is supported by King Abdulaziz City for Science and Technology (KACST), Riyadh, Saudi Arabia. We wish to thank the anonymous referees for many helpful suggestions.

References

Almuhareb, A. and Poesio, M. (2004). Attribute-Based and Value-Based Clustering: An Evaluation. In *Proc. of EMNLP*. Barcelona, July.

Almuhareb, A. and Poesio, M. (2005). Concept Learning and Categorization from the Web. In *Proc. of CogSci*. Italy, July.

Baader, F., Calvanese, D., McGuinness, D., Nardi, D. and Patel-Schneider, P. (Editors). (2003). *The Description Logic Handbook*. Cambridge University Press.

Berland, M. and Charniak, E. (1999). Finding parts in very large corpora. In *Proc. of the 37th ACL*, (pp. 57–64). University of Maryland.

Brill, E. (1995). Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging. *Computational Linguistics*.

Caraballo, S. A. (1999). Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proc. of the 37th ACL*.

Dixon, R. M. W. (1991). *A New Approach to English Grammar, on Semantic Principles*. Clarendon Press, Oxford.

Fellbaum, C. (Editor). (1998). *WordNet: An electronic lexical database*. The MIT Press.

Girju, R. and Moldovan, D. (2002). Mining answers for causal questions. In *Proc. AAAI*.

Guarino, N. (1992). Concepts, attributes and arbitrary relations: some linguistic and ontological criteria for structuring knowledge base. *Data and Knowledge Engineering*, 8, (pp. 249–261).

Hearst, M. A. (1998). Automated discovery of WordNet relations. In Fellbaum, C. (Editor). *WordNet: An Electronic Lexical Database*. MIT Press.

Karypis, G. (2002). *CLUTO: A clustering toolkit. Technical Report 02-017*. University of Minnesota. At http://www-users.cs.umn.edu/~karypis/cluto/.

Manning, C. D. and Schuetze H. (1999). *Foundations of Statistical NLP*. MIT Press.

Merlo, P. and Stevenson, S. (2001). Automatic Verb Classification Based on Statistical Distributions of Argument Structure. *Computational Linguistics*. 27: 3, 373-408.

Murphy, G. L. (2002). *The Big Book of Concepts*. The MIT Press.

Poesio, M., Ishikawa, T., Schulte im Walde, S. and Vieira, R. (2002). Acquiring lexical knowledge for anaphora resolution. In *Proc. Of LREC*.

Pustejovsky, J. (1995). *The generative lexicon*. MIT Press.

Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J. (1985). *A comprehensive grammar of the English language*. London: Longman.

Sanchez-Graillet, O. and Poesio, M. (2004). Building Bayesian Networks from text. In *Proc. of LREC*, Lisbon, May.

Vinson, D. P., Vigliocco, G., Cappa, S., and Siri, S. (2003). The breakdown of semantic knowledge: insights from a statistical model of meaning representation. *Brain and Language*, 86(3), 347-365(19).

Witten, I. H. and Frank, E. (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann.

Woods, W. A. (1975). What's in a link: Foundations for semantic networks. In Daniel G. Bobrow and Alan M. Collins, editors, *Representation and Understanding: Studies in Cognitive Science*, (pp. 35-82). Academic Press, New York.

Appendix A. 5-Way Automatic Classification of the Best Candidate Attributes of Some Concepts

Car

Class	Best Attributes
Activity	acceleration, performance, styling, construction, propulsion, insurance, stance, ride, movement
Part & Related-Object	front, body, mass, underside, hood, roof, nose, graphics, side, trunk, engine, boot, frame, bottom, backseat, chassis, wheelbase, silhouette, floor, battery, windshield, seat, undercarriage, tank, window, steering, drive, finish
Quality	speed, weight, handling, velocity, color, condition, width, look, colour, feel, momentum, heritage, shape, appearance, ownership, make, convenience, age, quality, reliability
Related-Agent	driver, owner, buyer, sponsor, occupant, seller
Non-Attribute	rest, price, design, balance, motion, lure, control, use, future, cost, inertia, model, wheel, style, position, setup, sale, supply, safety

Camel

Class	Best Attributes
Activity	introduction, selling, argument, exhaustion
Part & Related-Object	nose, hump, furniture, saddle, hair, flesh, neck, milk, head, reins, foot, eye, hooves, humps, ass, feet, hoof, flanks, bones, ears, bag, skin, haunches, stomach, legs, urine, meat, penis, load, breast, backside, testicles, rope, corpse, house, nostrils, foam, bell, sight, butt, fur, bodies, toe, hoofs, heads, knees, pancreas, mouth, coat, uterus, necks, chin, udders
Quality	origins, gait, domestication, usefulness, pace, fleetness, smell, existence, appeal, birth, awkwardness
Related-Agent	ghost
Non-Attribute	gift, rhythm, physiology, battle, case, example, dance, manner, description

Cancer

Class	Best Attributes
Activity	growth, development, removal, treatment, recurrence, diagnosis, pain, spreading, metastasis, detection, eradication, elimination, production, discovery, remission, advance, excision, prevention, evolution, disappearance, anxiety
Part & Related-Object	location, site, lump, nature, root, cells, margin, formation, margins, roots, world, region
Quality	extent, size, seriousness, progression, severity, aggressiveness, cause, progress, symptoms, effects, risk, incidence, staging, biology, onset, characteristics, histology, ability, status, appearance, thickness, sensitivity, causes, prevalence, responsiveness, ravages, frequency, aetiology, circumstances, rarity, outcome, behavior, genetics
Related-Agent	club, patient
Non-Attribute	stage, spread, grade, origin, course, power, return, area, response, presence, type, particulars, occurrence, prognosis, pathogenesis, source, news, cure, pathology, properties, genesis, boundaries, drama, stages, chapter

Family

Class	Best Attributes
Activity	disintegration, protection, decline, destruction, breakup, abolition, participation, reunification, reconciliation, dissolution, composition, restoration
Part & Related-Object	head, institution, support, flower, core, fabric, culture, dimension, food, lineage, cornerstone, community
Quality	breakdown, importance, honor, structure, sociology, integrity, unity, sanctity, health, privacy, survival, definition, influence, honour, involvement, continuity, stability, size, preservation, upbringing, centrality, ancestry, solidarity, hallmark, status, functioning, primacy, autonomy
Related-Agent	father, baby, member, mother, members, patriarch, breadwinner, matriarch, man, foundation, founder, heir, daughter
Non-Attribute	rest, role, income, history, concept, welfare, pedigree, genealogy, presence, context, origin, bond, tradition, taxonomy, system, wealth, lifestyle, surname, crisis, ideology, rights, economics, safety

Automatically Learning Qualia Structures from the Web

Philipp Cimiano & Johanna Wenderoth
Institute AIFB
University of Karlsruhe

Abstract

Qualia Structures have many applications within computational linguistics, but currently there are no corresponding lexical resources such as WordNet or FrameNet. This paper presents an approach to automatically learn qualia structures for nominals from the World Wide Web and thus opens the possibility to explore the impact of qualia structures for natural language processing at a larger scale. Furthermore, our approach can be also used support a lexicographer in the task of manually creating a lexicon of qualia structures. The approach is based on the idea of matching certain lexico-syntactic patterns conveying a certain semantic relation on the World Wide Web using standard search engines. We evaluate our approach qualitatively by comparing our automatically learned qualia structures with the ones from the literature, but also quantitatively by presenting results of a human evaluation.

1 Introduction

Qualia Structures have been originally introduced by (Pustejovsky, 1991) and are used for a variety of purposes in Natural Language processing such as the analysis of compounds (Johnston and Busa, 1996), co-composition and coercion (Pustejovsky, 1991) as well as for bridging reference resolution (Bos et al., 1995). Further, it has also been argued that qualia structures and lexical semantic relations in general have applications in information retrieval (Voorhees, 1994; Pustejovsky et al., 1993). One major bottleneck however is that currently Qualia Structures need to be created by hand, which is probably also the reason why there are no practical system using qualia structures, but a lot of systems using globally available resources such as WordNet (Fellbaum, 1998) or FrameNet[1] as source of lexical/world knowledge. The work described in this paper addresses this issue and presents an approach to automatically learning qualia structures for nominals from the Web. The approach is inspired in recent work on using the Web to identify instances of a relation of interest such as in (Markert et al., 2003) and (Cimiano and Staab, 2004). These approaches are in essence a combination of the usage of lexico-syntactic pattens conveying a certain relation of interest such as in (Hearst, 1992), (Charniak and Berland, 1999), (Iwanska et al., 2000) or (Poesio et al., 2002) with the idea of using the web as a big corpus (Resnik and Smith, 2003), (Grefenstette, 1999), (Keller et al., 2002).

The idea of learning Qualia Structures from the Web is not only a very practical, it is in fact a principled one. While single lexicographers creating qualia structures - or lexicon entries in general - might take very subjective decisions, the structures learned from the Web do not mirror the view of a single person, but of the whole world as represented on the World Wide Web. Thus, an approach learning qualia structures from the Web is in principle more reliable than letting lexicographers craft lexical entries on their own. Obviously, on the other hand, using an automatic web based approach yields also a lot of inappropriate results which are due to 1) errors produced by the linguistic analysis (e.g. part-of-speech tagging), 2) idiosyncrasies of ranking algorithms of search machines, 3) the fact that the Web or in particular search engines are to a great extent commercially biased, 4) the fact that people also publish erroneous information on the Web, and 5) lexical ambiguities. Because of these reasons our aim is in fact not to replace lexicographers, but to support them in the task of creating qualia structures on the basis of the automatically learned qualia structures. The paper is structured as follows: Section 2 introduces qualia structures and describes the specific qualia structures we aim to acquire. Section 3 describes our approach in detail and section 4 presents a quantitative and qualitative evaluation of our approach. Before concluding, we discuss some related work in Section 5.

[1] http://framenet.icsi.berkeley.edu/

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 28–37,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

2 Qualia Structures

According to Aristotle, there are four basic factors or causes by which the nature of an object can be described (cf. (Kronlid, 2003)):

- the *material cause*, i.e. the material an object is made of
- the *agentive cause*, i.e. the source of movement, creation or change
- the *formal cause*, i.e. its form or type
- the *final cause*, i.e. its purpose, intention or aim

In his Generative Lexicon (GL) framework (Pustejovsky, 1991) reused Aristotle's basic factors for the description of the meaning of lexical elements. In fact he introduced so called *Qualia Structures* by which the meaning of a lexical element is described in terms of four roles:

- *Constitutive*: describing physical properties of an object, i.e. its weight, material as well as parts and components
- *Agentive*: describing factors involved in the *bringing about* of an object, i.e. its creator or the causal chain leading to its creation
- *Formal*: describing that properties which distinguish an object in a larger domain, i.e. orientation, magnitude, shape and dimensionality
- *Telic*: describing the purpose or function of an object

Most of the qualia structures used in (Pustejovsky, 1991) however seem to have a more restricted interpretation. In fact, in most examples the *Constitutive* role seems to describe the parts or components of an object, while the *Agentive* role is typically described by a verb denoting an action which typically brings the object in question into existence. The *Formal* role normally consists in typing information about the object, i.e. its hypernym or superconcept. Finally, the *Telic* role describes the purpose or function of an object either by a verb or nominal phrase. The qualia structure for *knife* for example could look as follows (cf. (Johnston and Busa, 1996)):

Formal:	artifact_tool
Constitutive:	blade,handle,...
Telic:	cut_act
Agentive:	make_act

Our understanding of *Qualia Structure* is in line with this restricted interpretation of the qualia roles. Our aim is to automatically acquire Qualia Structures from the Web for nominals, looking for (i) nominals describing the type of the object, (ii) verbs defining its agentive role, (iii) nominals describing its parts or components and (iv) nouns or verbs describing its intended purpose.

3 Approach

Our approach to learning qualia structures from the Web is on the one hand based on the assumption that instances of a certain semantic relation can be learned by matching certain lexico-syntactic patterns more or less reliably conveying the relation of interest in line with the seminal work of (Hearst, 1992), who defined the following patterns conveying a hypernym relation:

(1) NP_0 such as NP_1, NP_2, ..., NP_{n-1} (and|or) NP_n[2]
(2) such NP_0 as NP_1, NP_2, ... NP_{n-1} (and|or) NP_n
(3) NP_1, NP_2, ..., NP_n (and|or) other NP_0
(4) NP_0, (including|especially) NP_1, NP_2, ..., NP_{n-1} (and|or) NP_n

According to Hearst, from such patterns we can derive that for all NP_i, $1 \leq i \leq n$, $hypernym(NP_i, NP_0)$. For example, for the expression: *Bruises, wounds, broken bones or other injuries*, we would extract: *hypernym(bruise,injury)*, *hypernym(broken bone,injury)* and *hypernym(wound,injury)*. However, it is well known that Hearst-style patterns occur rarely, such that it seems intuitive to match them on the Web. So in our case we are looking not only for the hypernym relation (comparable to the *Formal*-Relation) but for similar patterns conveying a *Constitutive*, *Telic* or *Agentive* relation. As currently there is no support for searching using regular expressions in standard search engines such as Google or Altavista[3], our approach consists of 5 phases (compare Figure 1):

1. generate for each qualia role a set of so called *clues*, i.e. search engine queries indicating the relation of interest
2. download the snippets of the 10 first Google hits matching the generated clues [4]
3. part-of-speech-tagging of the downloaded snippets
4. match regular expressions conveying the qualia role of interest
5. weight the returned qualia elements according to some measure

The outcome of this process are then so called *Weighted Qualia Structures* (WQSs) in which every

[2] NP_i stands for a noun phrase.

[3] An exception is certainly the Linguist's Search Engine (Resnik and Elkiss, 2003)

[4] The reason for using only the 10 first hits is to maintain efficiency. With the current setting the systems needs between 3 and 10 minutes to generate the qualia structure for a given nominal

qualia element in a certain role is weighted according to some measure. The patterns in our pattern library are actually tuples (p, c) where p is a regular expression defined over part-of-speech tags and c a function $c : string \rightarrow string$ called the *clue*. Given a nominal t and a clue c, the query $c(t)$ is sent to the Google API and we download the abstracts of the first n documents matching this query and then process the abstracts to find instances of pattern p. For example, given the clue $f(x) = ”such\ as\ ”\pi(x)$ and the instance *computer* we would download n abstracts matching the query f(computer), i.e. ”such as computers”. Hereby $\pi(x)$ is a function returning the plural form of x. We implemented this function as a lookup in a lexicon in which plural nouns are mapped to their base form. With the use of such clues, we thus download a number of Google-abstracts in which a corresponding pattern will probably be matched thus restricting the linguistic analysis to a few promising pages. The downloaded abstracts are then part-of-speech tagged using QTag (Tufis and Mason, 1998). Then we match the corresponding pattern p in the downloaded snippets thus yielding candidate qualia elements as output. In our approach we then calculate the weight of a candidate qualia element e for the term t we want to compute the qualia structure for by the *Jaccard Coefficient*:

$$\frac{GoogleHits(e+t)}{GoogleHits(e) + GoogleHits(t) - GoogleHits(e+t)}$$

The result is then a *Weighted Qualia Structure* (WQS) in which for each role the qualia elements are weighted according to this Jaccard coefficient. In what follows we describe in detail the procedure for acquiring qualia elements for each qualia role. In particular, we describe in detail the clues and lexico-syntactic patterns used. In general, the patterns have been crafted by hand, testing and refining them in an iterative process, paying attention to maximize their coverage but also accuracy.

In general it is important to mention that by this approach we are not able to detect and separate multiple meanings of words, i.e. to handle polysemy, which is appropriately accounted for in the framework of the Generative Lexicon (Pustejovsky, 1991).

3.1 The Formal Role

To derive qualia elements for the *Formal* role, we first download for each of the clues in Table 1 the first 10 abstracts matching the clue and then process them offline matching the patterns defined over part-of-speech-tags[5] thus yielding up to 10 different qualia element candidates per clue. The patterns are specified in form of regular expressions, whereby the part-of-speech tags are always

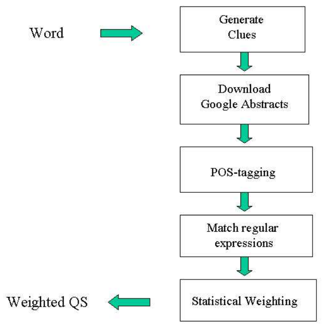


Figure 1: General Approach

given in square brackets after the token. Further, besides using the traditional regular expression operators such as $+$, $*$ and ?, we also use Perl-like symbols such as $\backslash w$ denoting any alphabetic character as well as [a-z] denoting the set of all lower case letters.

As there are 4 different clues for the *Formal* role, we thus yield up to 40 qualia elements as potential candidates to fill the *Formal* role. In general, we paid attention to create clues relying on indefinite articles as we found out that they produce more general and reliable results than when using definite articles. In order to choose the correct indefinite article – *a* or *an* – or even using no article at all, we implemented some ad-hoc heuristics checking if the first letter of the term in question is a vowel and checking if the term is used more often with an article or without an article on the Web by a set of corresponding Google queries. The alternative ’(a/an/?)’ means that we use either the indefinite article ’a’ ’an’ or no article depending on the results of the above mentioned Google queries.

A general question raised also by Hearst (Hearst, 1992) is how to deal with NP modification. Hearst’s conclusion is that this depends on the application. In our case we mainly remove adjective modifiers, keeping only the heads of noun phrases as candidate qualia elements. The lemmatized heads of the NP$_F$ noun phrase are then regarded as qualia role candidates for the *Formal* role. These candidates are then weighted using the above defined *Jaccard Coefficient* measure. Hereby, a noun phrase is an instance matching the following regular expression:

NP:=[a-z]+[DT]? ([a-z]+[JJ])+? ([a-z]+[NN(S?)])+,

where the head is the underlined expression, which is lemmatized and considered as a candidate qualia element. After some initial experiments we decided not to use the patterns ’X is Y’ and ’X is a kind of Y’ such as in *a book is an item* or *a book is a kind of publication*

[5] We use the well-known Penn Treebank tagset described at http://www.computing.dcu.ie/∼acahill/tagset.html.

as well as the pattern ’Y, including X’ (compare (Hearst, 1992)) as we found that in our settings they delivered quite spurious results.

Clue	Pattern
such as $\pi(t)$	NP$_F$,? such[DT] as[IN] NP
especially $\pi(t)$	NP$_F$,? especially[RB] NP
$\pi(t)$ or other	NP or[CC] other[JJ] NP$_F$
$\pi(t)$ and other	NP and[CC] other[JJ] NP$_F$

Table 1: Clues and Patterns for the *Formal* role

3.2 The Constitutive Role

The procedure for finding elements of the *Constitutive* role is similar to the one described above for the *Formal* role. The corresponding clues and patterns are given in Table 2. As above, the candidate qualia elements are then the lemmatized heads of the noun phrase NP$_C$.

Clue	Pattern
(a/an)? t is made up of	NP is[VBZ] made[VBN] up[RP] of[IN] NP$_C$
$\pi(t)$ are made up of	NP are[VBP] made[VBN] up[RP] of[IN] NP$_C$
(a/an)? t is made of	NP are[VBP] made[VBN] of[IN] NP$_C$
$\pi(t)$ are made of	NP are[VBP] made[VBN] of[IN] NP$_C$
(a/an)? t comprises	NP comprises[VBZ] NP$_C$
$\pi(t)$ comprise	NP comprise[VBP] NP$_C$
(a/an)? t consists of	NP consists[VBZ] of[IN] NP$_C$
$\pi(t)$ consist of	NP consist[VBP] of[IN] NP$_C$

Table 2: Clues and Patterns for the *Constitutive* Role

As an additional heuristic, we test if the lemmatized head of NP$_C$ is an element of the following list containing nouns denoting an indication of amount: {*variety, bundle, majority, thousands, million, millions, hundreds, number, numbers, set, sets, series, range*} and furthermore this NP$_C$ is followed by the preposition ’of’. In that case we would take the head of the noun phrase after the preposition ’of’ as potential candidate of the *Constitutive* role. For example, when considering *a conversation is made up of a series of observable interpersonal exchanges*, we would take *exchange* as a potential qualia element candidate instead of *series*.

3.3 The Telic Role

The *Telic* Role is in principle acquired in the same way as the *Formal* and *Constitutive* roles with the exception that the qualia element is not only the head of a noun phrase, but also a verb or a verb followed by a noun phrase. Table 3 gives the corresponding clues and patterns. In particular, the returned candidate qualia elements are the lemmatized underlined expressions in PURP:=\w+[VB] NP | NP | be[VB] \w+[VBD]).

Clue	Pattern
purpose of a t is	purpose[NN] of[IN] NP$_0$ is[VBZ] (to[TO])? PURP
purpose of $\pi(t)$ is	purpose[NN] of[IN] NP$_0$ is[VBZ] (to[TO])? PURP
(a/an)? t is used to	(A|a|An|an) NP$_0$ is[VBZ] used[VBN] to[TO] PURP
$\pi(t)$ are used to	NP$_0$ are[VBZ] used[VBN] to[TO] PURP

Table 3: Clues and Patterns for the *Telic* Role

3.4 The Agentive Role

As mentioned in (Hearst, 1992), it is not always as straightforward to find lexico-syntactic patterns reliably conveying a certain relation. In fact, we did not find any patterns reliably identifying qualia elements for the *Agentive* role. Certainly, it would have been possible to find the source of the creation by using patterns such as *X is made by Y* or *X is produced by Y*. However, we found that these patterns do not reliably convey a verb describing how an object is brought into existence. The fact that it is far from straightforward to find patterns indicating an *Agentive* role is further corroborated by the research in (Yamada and Baldwin, 2004), in which only one pattern indicating a qualia relation is used, namely ’NN BE V[+en]’ in order to match passive constructions such as *the book was written*. On the other hand it is clear that constructing a reliable clue for this pattern is not straightforward given the current state-of-the-art concerning search engine queries. Nevertheless, in order to also get results for the *Agentive* role, we apply a different method here. Instead of issuing a query which is used to search for possible candidates for the role, we take advantage of the fact that the verbs which describe how something comes into being, particularly artificial things, are often quite general phrases like ”make, produce, write, build...”. So instead of generating clues as above, we calculate the value $\frac{GoogleHits(<AGENTIVE\_VERB>\ a\ t)}{GoogleHits(t)}$ for the nominal we want to acquire a qualia structure for as well as the following verbs: *build, produce, make, write, plant, elect, create, cook, construct* and *design*. If this value is over a threshold (0.0005 in our case), we assume that it is a valid filler of the *Agentive* qualia role.

4 Evaluation

We evaluate our approach for the lexical elements *knife, beer, book*, which are also discussed in (Johnston and

Busa, 1996) or (Pustejovsky, 1991), as well as *computer*, an abstract noun, i.e. *conversation*, as well as two very specific multi-term words, i.e. *natural language processing* and *data mining*. We give the automatically learned weighted Qualia Structures for these entries in Figures 3, 4, 5 and 6. The evaluation of our approach consists on the one hand of a discussion of the weighted qualia structures, in particular comparing them to the ideal structures form the literature. On the other hand, we also asked a student at our institute to assign credits to each of the qualia elements from 0 (incorrect) to 3 (totally correct) whereby 1 credit meaning 'not totally wrong' and 2 meaning 'still acceptable'.

4.1 Quantitative Evaluation

The distribution of credits for each qualia role and term is given in Table 4. It can be seen that with three exceptions: *beer→formal*, *book→agentive* as well as *beer→constitutive*, '3' is the mark assigned in most cases to the automatically learned qualia elements. Further, for almost every query term and qualia role, at least 50% of the automatically learned qualia structures have a mark of '2' or '3' – the only exceptions being *beer→formal* with 45.45%, *book→agentive* with 33.33% and *beer→constitutive* with 28.57%. In general this shows that the automatically learned qualia roles are indeed reasonable. Considering the average over all the terms ('All' in the table), we observe that the qualia role which is recognized most reliably is the *Telic* one with 73.15% assignments of credit '3' and 75.93% of credits '2' or '3', followed by the *Agentive* role with 71.43% assignments of credit 3. The results for the *Formal* and *Constitutive* role are still reasonable with 62.09% assignments of credit '3' and 66.01% assignments of credits '2' or '3' for the *Formal role*; and respectively 61.61% and 64.61% for the *Constitutive* role. The worst results are achieved for the *Constitutive* role due to the fact that 26.26% of the qualia elements are regarded as totally wrong. Table 5 supports the above claims and shows the average credits assigned by the human evaluator per query term and role. It shows again that the roles with the best results are the *Agentive* and *Telic* roles, while the *Formal* and *Constitutive* roles are not identified as accurately. This is certainly due to the fact that the patterns for the *Telic* role are much less ambiguous than the ones for the *Formal* and *Constitutive* roles. Finally, we also discuss the correlation between the credits assigned and the *Jaccard Coefficient*. Figure 2 shows this correlation. While for the *Formal* role the correlation is as expected, i.e. the higher the credit assigned, the higher also the Jaccard Coefficient, for the *Constitutive* and *Telic* roles this correlation is unfortunately less clear, thus making the task of finding a cut-off threshold more difficult.

4.2 Qualitative Evaluation & Discussion

In this section we provide a more subjective evaluation of the automatically learned qualia structures by comparing them to ideal qualia structures discussed in the literature wherever possible. In particular, we discuss more in detail the qualia structure for *book*, *knife* and *beer* and leave the detailed assessment of the qualia structures for *computer*, *natural language processing*, *data mining* and *conversation* to the interested reader.

For book, the first four candidates of the *Formal* role, i.e. *product, item*, *publication* and *document* are very appropriate, but alluding to the *physical object* meaning of book as opposed to the meaning in the sense of *information container* (compare (Pustejovsky, 1991). As candidates for the *Agentive* role we have *make*, *write* and *create* which are appropriate, *write* being the ideal filler of the *Agentive* role according to (Pustejovsky, 1991). For the *Constitutive* role of *book* we get – besides *it* at the first position which could be easily filtered out – *sign* (2nd position), *letter* (3rd position) and *page* (6th position), which are quite appropriate. The top four candidates for the *Telic* role are *give, select*, *read* and *purchase*. It seems that *give* is emphasizing the role of a book as a gift, *read* is referring to the most obvious purpose of a book as specified in the ideal qualia structures of (Pustejovsky, 1991) as well as (Johnston and Busa, 1996) and *purchase* denotes the more general purpose of a book, i.e. to be bought.

The first element of the *Formal* role of *knife* unfortunately denotes the material it is typically made of, i.e. *steel*, but the next 5 elements are definitely appropriate: *weapon, item*, *kitchenware*, *object* and *instrument*. The ideal element *artifact_tool* (compare (Johnston and Busa, 1996)) can be found at the 10th position. The results are interesting in that on the one hand the most prominent meaning of *knife* according to the web is the one of a *weapon*. On the other hand our results are more specific, classifying a knife as *kitchenware* instead of merely as an *artifact_tool*. Very interesting are the specific and accurate results at the end of the list. The reason why they appear at the end is that the Jaccard Coefficient ranks them lower because they are more specific, thus appearing less frequently. This shows that using some other measure less sensitive to frequency could yield more accurate results. The fillers of the *Agentive* role *produce, make* and *create* seem all appropriate, whereby *make* corresponds exactly to the ideal filler for the *Agentive* role as mentioned in (Johnston and Busa, 1996). The results for the *Constitutive* role contain not only parts but also materials a knife is made of and thus contain more information than the typical qualia structures assumed in the literature. The best results are (in this order) *blade, metal, steel, wood* and *handle* at the 6th position. In fact, in the ideal qualia structure in (Johnston and Busa, 1996) *blade* and *han-*

	Formal			
	0	1	2	3
Book	2/17 (11.76%)	4/17 (23.52%)	1/17 (5.88%)	10/17 (58.82%)
Computer	8/28 (28.57%)	1/28 (3.57%)	2/28 (7.14%)	17/28 (60.71%)
Knife	3/16 (18.75%)	0/16 (0%)	0/16 (0%)	13/16 (81.25%)
Beer	12/22 (54.54%)	0/22 (0%)	2/22 (9.09%)	8/22 (36.36%)
Data Mining	6/25 (24%)	0/25 (0%)	0/25 (0%)	19/25 (76%)
Natural Language Processing	2/15 (13.33%)	1/15 (6.66%)	0/15 (0%)	12/15 (80%)
Conversation	10/30 (33.33%)	4/30 (13.33%)	0/30 (0%)	16/30 (53.33%)
All	43/153 (28.10%)	11/153 (7.19%)	6/153 (3.92%)	95/153 (62.09%)
	Agentive			
Book	0/3 (0%)	2/3 (66.66%)	0/3 (0%)	1/3 (33.33%)
Computer	0/1 (0%)	0/1 (0%)	0/1 (0%)	1/1 (100%)
Knife	0/3 (0%)	0/3 (0%)	0/3 (0%)	3/3 (100%)
Beer	0/3 (0%)	1/3 (33.33%)	0/3 (0%)	2/3 (66.66%)
Data Mining	0/1 (0%)	0/1 (0%)	0/1 (0%)	1/1 (100%)
Natural Language Processing	0/1 (0%)	0/1 (0%)	0/1 (0%)	1/1 (100%)
Conversation	1/2 (50%)	0/2 (0%)	0/2 (0%)	1/2 (50%)
All	1/14 (7.14%)	3/14 (21.43%)	0/14 (0%)	10/14 (71.43%)
	Constitutive			
Book	8/29 (27.58%)	4/29 (13.79%)	1/29 (3.44%)	16/29 (55.17%)
Computer	6/26 (23.07%)	1/26 (3.84%)	0/26 (0%)	19/26 (73.07%)
Knife	4/15 (26.66%)	0/15 (0%)	0/15 (0%)	11/15 (73.33%)
Beer	5/7 (71.42%)	0/7 (0%)	0/7 (0%)	2/7 (28.57%)
Data Mining	0/1 (0%)	0/1 (0%)	0/1 (0%)	1/1 (100%)
Natural Language Processing				
Conversation	3/21 (14.28%)	4/21 (19.04%)	0/21 (0%)	14/21 (66.66%)
All	26/99 (26.26%)	9/99 (9%)	3/99 (3%)	61/99 (61.61%)
	Telic			
Book	3/22 (13.63%)	2/22 (9.09%)	3/22 (13.63%)	14/22 (63.63%)
Computer	0/27 (0%)	3/27 (11.11%)	0/27 (0%)	24/27 (88.88%)
Knife	5/18 (27.77%)	0/18 (0%)	0/18 (0%)	13/18 (72.22%)
Beer				
Data Mining	2/22 (9.09%)	4/22 (18.18%)	0/22 (0%)	16/22 (72.72%)
Natural Language Processing	1/6 (16.66%)	0/6 (0%)	0/6 (0%)	5/6 (83.33%)
Conversation	6/13 (46.15%)	0/13 (0%)	0/13 (0%)	7/13 (53.84%)
All	17/108 (15.74%)	9/108 (8.33%)	3/108 (2.78%)	79/108 (73.15%)

Table 4: Distribution of credits for each role and term

	Formal	Agentive	Constitutive	Telic
Book	2.12	1.67	1.86	2.27
Computer	2	3	2.23	2.78
Knife	2.44	3	2.2	2.17
Beer	1.27	2.33	0.96	n.a.
Data Mining	2.28	3	3	2.36
Natural Language Processing	2.47	3	n.a.	2.5
Conversation	1.73	1.5	2.19	1.62
All	1.99	2.36	2.02	2.33

Table 5: Average credits for each role

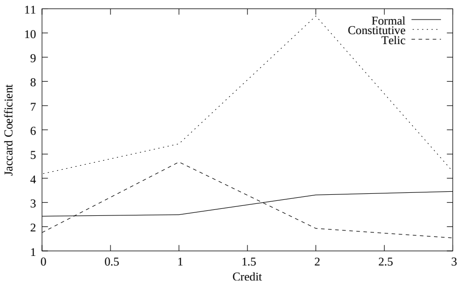


Figure 2: Average Jaccard Coefficient value per credit

dle are mentioned as fillers of the *Constitutive* role, while there are no elements describing the materials of which a knife is made of. Finally, the top four candidates for the *Telic* role are *kill*, *slit*, *cut* and *slice*, whereby *cut* corresponds to the ideal filler of the qualia structure for *knife* as mentioned in (Johnston and Busa, 1996).

Considering the qualia structure for *beer*, it is surprising that no purpose has been found. The reason is that currently no results are returned by Google for the clue *a beer is used to* and the four snippets returned for *the purpose of a beer* contain expressions of the form *the purpose of a beer is to drink it* which is not matched by our patterns as *it* is a pronoun and not matched by our NP pattern (unless it is matched by an error as in the Qualia Structure for *book* in Figure 4). Considering the results for the *Formal* role, the elements *drink* (1st), *alcohol* (2nd) and *beverage* (4th) are much more specific than *liquid* as given in (Pustejovsky, 1991), while *thing* at the 3rd position is certainly too general. Furthermore, according to the automatically learned qualia structure, *beer* is made of *rice*, *malt* and *hop*, which are perfectly reasonable results. Very interesting are the results *concoction* and *libation* for the *Formal* role of beer, which unfortunately were rated low by our evaluator (compare Figure 3).

Overall, the discussion has shown that the results produced by our method are reasonable when compared to the qualia structures from the literature. In general, our method produces in some cases additional qualia candidates, such as the ones describing the material a knife is typically made of. In other cases it discovers more specific candidates, such as for example *weapon* or *kitchenware* as elements of the *Formal* role for knife instead of the general term *artifact_tool*.

5 Related Work

There is quite a lot of work related to the use of linguistic patterns to discover certain ontological relations from text. Hearst's (Hearst, 1992) seminal work had the aim of discovering taxonomic relations from electronic dictionaries. The precision of the *is-a*-relations learned is 61/106 (57.55%) when measured against WordNet as gold standard, which is comparable to our results. Hearst's idea has been reapplied by different researchers with either slight variations in the patterns used (Iwanska et al., 2000), to acquire knowledge for anaphora resolution (Poesio et al., 2002), or to discover other kinds of semantic relations such as part-of relations (Charniak and Berland, 1999) or causation relations (Girju and Moldovan, 2002).

Instead of matching these patterns in a large text collection, some researchers have recently turned to the Web to match these patterns such as in (Cimiano and Staab, 2004) or (Markert et al., 2003). (Cimiano and Staab, 2004) for example aim at learning instance-of as well as taxonomic (*is-a*) relations. This is very related to the acquisition of the *Formal* role proposed here. (Markert et al., 2003) aim at acquiring knowledge for anaphora resolution, while (Etzioni et al., 2004) aim at learning the complete extension of a certain concept. For example, they aim at finding all the actors in the world.

Our approach goes further in that it not only learns typing, superconcept or instance-of relations, but also *Constitutive* and *Telic* relations.

There also exist approaches specifically aiming at learning qualia elements from corpora based on machine learning techniques. (Claveau et al., 2003) for example use Inductive Logic Programming to learn if a given verb is a qualia element or not. However, their approach goes not as far as learning the complete qualia structure for a lexical element in an unsupervised way as presented in our approach. In fact, in their approach they do not distinguish between different qualia roles and restrict themselves to verbs as potential fillers of qualia roles. (Yamada and Baldwin, 2004) present an approach to learning *Telic* and *Agentive* relations from corpora analyzing two different approaches: one relying on matching certain lexico-syntactic patterns as in the work presented here, but also a second approach consisting in training a maximum entropy model classifier. Their conclusion is that the results produced by the classification approach correlate better with two hand-crafted gold standards.

The patterns used by (Yamada and Baldwin, 2004) differ substantially from the ones used in this paper, which is mainly due to the fact that search engines do not provide support for regular expressions and thus instantiating a pattern as ’V[+ing] Noun’ is impossible in our approach as the verbs are unknown a priori.

Finally, (Pustejovsky et al., 1993) present an interesting framework for the acquisition of semantic relations from corpora not only relying on statistics, but guided by theoretical lexicon principles.

6 Conclusion

We have presented an approach to automatically learning Qualia Structures from the Web. Such an approach is especially interesting either for lexicographers aiming at constructing lexicons, but even more for natural language processing systems relying on deep lexical knowledge as represented by qualia structures. We have in particular shown that the qualia structures learned by our system are reasonable. In general, it is valid to claim that our system is the first one automatically producing complete qualia structures for a given nominal.

Our system can be tested online at http://km.aifb.uni-karlsruhe.de/pankow/qualia/. Further work will aim at improving the system but also at using the automatically learned structures within NLP applications.

Acknowledgments The work reported in this paper has been partially supported by the SmartWeb project[6], funded by the German Ministry of Research. Thanks also to Laura Goebes for assisting in the evaluation of the system.

Knife		
Formal		
steel	3.8666	3
weapon	3.4876	3
item	1.7458	3
kitchenware	1.6840	3
object	1.6025	3
instrument	1.2963	3
utensil	1.2886	3
court	1.1441	0
equipment	0.9479	3
tool	0.7090	3
action	0.7028	0
time	0.6590	0
cutting instrument	0.0739	3
cutting instruments	0.0551	3
emergency items	0.0383	3
cutting weapons	0.0232	3
Agentive		
produce		3
make		3
create		3
Constitutive		
blade	5.4618	3
metal	5.0205	3
steel	3.8666	3
wood	2.9699	3
person	2.6829	0
handle	1.9223	3
tang	1.6784	3
gold	1.6609	0
alloy	1.2466	3
dragonfly	0.8742	3
model	0.7513	3
tool	0.7090	0
quality	0.6575	3
group	0.5764	0
rotating discs	0.0062	3
Telic		
kill	3.7626	3
slit	3.4829	3
cut	3.4373	3
slice	3.2499	3
begin	2.4192	0
split	1.7241	3
avoid	1.3190	0
score	1.0204	0
an instrument	0.8137	0
process	0.5327	3
prune	0.4505	3
incise	0.0573	3
cut things	0.0545	3
remove moisture	0.0479	3
add details	0.0361	0
cut a flap	0.0264	3
split a cake	0.0010	3
slit a wide variety	0.0004	3

Beer		
Formal		
drink	9.6677	3
alcohol	4.6006	3
thing	4.0028	3
beverage	3.6182	3
adventure	3.0825	0
mistake	2.7014	0
matter	2.6533	0
style	2.1583	0
delight	1.9198	3
people	1.4465	0
creation	1.2201	0
can	0.9433	3
list	0.8432	0
product	0.8224	3
refreshment	0.5328	3
concoction	0.4851	0
libation	0.1147	0
summery	0.0872	0
adult beverages	0.0848	2
speciality beers	0.0269	2
looney things	0.0002	0
Agentive		
produce		3
make		3
create		1
Constitutive		
rice	2.9871	0
malt	2.5724	3
hop	2.1744	3
bottom	2.1179	0
continuum	0.4808	0
puree	0.3563	0
stoneware	0.3325	0

Figure 3: Weighted Qualia Structure for *knife* and *beer*

References

J. Bos, P. Buitelaar, and M. Mineur. 1995. Bridging as coercive accomodation. In E. Klein, S. Manandhar, W. Nutt, and J. Siekmann, editors, *Working Notes of the Edinburgh Conference on Computational Logic and Natural Language Processing (CLNLP-95)*.

E. Charniak and M. Berland. 1999. Finding parts in very large corpora. In *Proceedings of the 37th Annual Meeting of the ACL*, pages 57–64.

P. Cimiano and S. Staab. 2004. Learning by googling. *SIGKDD Explorations*, 6(2), December.

V. Claveau, P. Sebillot, C. Fabre, and P. Bouillon. 2003. Learning semantic lexicons from a part-of-speech and semantically tagged corpus using inductive logic programming. *Journal of Machine Learning Research*, (4):493–525.

O. Etzioni, M. Cafarella, D. Downey, S. Kok, A.-M. Popescu, T. Shaked, S. Soderland, D.S. Weld, and A. Yates. 2004. Web-scale information extraction in KnowItAll (preliminary results). In *Proceedings of the 13th World Wide Web Conference*, pages 100–109.

C. Fellbaum. 1998. *WordNet, an electronic lexical database*. MIT Press.

R. Girju and M. Moldovan. 2002. Text mining for causal relations. In *Proceedings of the FLAIRS Conference*, pages 360–364.

G. Grefenstette. 1999. The WWW as a resource for example-based MT tasks. In *Proceedings of ASLIB’99 Translating and the Computer 21*.

M.A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th In-*

[6] http://www.smartweb-projekt.de/

Book		
Formal		
product	34.6238	3
item	33.8573	3
publication	20.2621	3
document	14.4778	3
history	12.7262	1
project	8.9809	2
material	8.6704	3
reader	8.3890	0
resource	7.7259	3
source	7.6739	3
piece	7.6131	3
format	7.2203	0
tool	6.1124	1
object	3.7705	3
specifics	0.5374	1
library materials	0.1468	3
library property	0.0026	1
Agentive		
make		1
write		3
create		1
Constitutive		
it	21.5785	0
sign	21.0870	3
letter	18.7778	3
part	11.7830	1
individual	11.4043	0
page	10.9202	3
collection	10.7901	0
teaching	10.7004	2
language	9.6041	1
period	9.4002	0
paper	9.3551	3
table	8.7089	3
material	8.6704	3
word	8.1424	3
piece	7.6131	0
chapter	7.4746	3
presentation	7.0955	3
detail	6.8218	3
minute	5.3550	0
sheet	4.4369	3
lie	3.0866	1
ticket	2.3198	0
ink	2.2769	3
dot	1.7427	3
leather	1.1162	1
leaf	1.0266	3
title page	0.3639	3
peice	0.0530	0
dedication page	0.0076	3
Telic		
give	14.8954	1
select	12.9594	0
read	12.4937	3
purchase	9.0372	3
support	8.0204	3
identify	7.9388	1
represent	5.7829	2
inspire	1.7292	3
convey	1.3940	3
present information	0.0728	3
provide additional information	0.0368	3
convey information	0.0260	3
filch	0.0101	3
share a story	0.0081	3
commit crime	0.0061	0
contain words	0.0055	3
introduce concepts	0.0038	2
traprock	0.0015	0
stock libraries	0.0009	3
hold a collection	0.0008	3
fund special projects	0.0007	2
support teachings	0.0001	3

Computer		
Formal		
technology	20.3667	3
information	20.2418	0
network	14.8052	3
hardware	14.6539	3
service	13.9161	3
office	12.2881	0
equipment	7.4594	2
machine	7.0099	3
item	6.7469	3
device	5.6259	3
medium	4.0503	3
fix	3.9188	0
piece	3.5898	3
notebook	2.1126	3
circuit	1.8663	0
consumer electronics	1.1544	0
appliance	1.0045	3
toy	0.7934	3
office equipment	0.4055	3
datum	0.3262	0
computer clipart	0.3156	1
mentality	0.1158	0
network device	0.0343	3
artefact	0.0339	3
data stores	0.0133	3
display screen equipment	0.0042	2
library equipment	0.0037	3
complex computer processes	0.0001	0
Agentive		
build		3
Constitutive		
software	25.5230	3
hardware	14.6539	3
part	14.6224	1
electronics	9.6139	3
individual	9.3791	0
memory	8.9683	3
man	5.9584	0
device	5.6259	3
unit	5.2078	3
component	4.3808	3
switch	4.2159	3
mix	3.8996	0
string	1.8896	3
circuit	1.8663	0
silicon	1.7717	3
actor	1.2127	0
processing unit	0.1444	3
individual components	0.1122	3
hardware components	0.1087	3
centra	0.0530	0
computer codes	0.0463	3
plastic case	0.0167	3
data storage device	0.0077	3
transitors	0.0022	3
Telic		
make	16.9616	1
access	15.5691	3
control	12.2216	3
run	8.6411	3
assist	4.1410	3
publish	3.0015	3
solve	2.9701	3
facilitate	2.8860	3
insight	2.2718	3
combine	1.9592	1
calculate	1.2977	3
execute	1.2792	3
translate	1.2530	3
suppose	1.1340	3
provide information	0.8969	3
access data	0.1025	3
imitate	0.0998	1
provide feedback	0.0900	3
human freedom	0.0065	3
teach children	0.0266	3
enable people	0.0255	3
manage information	0.0231	3
process words	0.0009	3
support program goals	0.0003	3
reduce analysis time	0.0002	3
perform useful computations	0.0001	3

Conversation		
Formal		
concept	6.6834	3
expression	5.8487	3
context	5.2338	3
object	4.6343	0
sound	4.4566	0
function	4.1414	0
material	4.1324	0
place	3.7806	0
employee	3.4710	0
skill	3.3323	3
interaction	3.1092	3
communication	3.0006	3
activity	2.9859	3
people	2.9027	0
label	2.7427	3
time	2.6158	1
source	1.6782	0
text	1.5877	1
transmission	1.2251	3
information	1.2182	3
contact	1.1309	3
utterance	0.9499	1
transaction	0.9412	3
school activities	0.2094	3
datum	0.1462	3
mannerism	0.0635	0
communication difficulties	0.0412	1
ambient audio	0.0148	3
official forms	0.0140	3
priceless tidbits	0.0002	0
Agentive		
make		3
create		0
Constitutive		
relationship	6.1848	3
silence	5.7213	3
answer	5.6855	3
question	4.8714	3
sentence	4.8663	3
story	4.4669	3
laughter	3.1766	1
unit	2.9359	1
tree	2.7633	0
contribution	2.6421	3
world	2.1804	0
sequence	1.8986	3
requests	1.4969	3
repetition	1.4267	3
token	1.2746	1
bonus	1.2155	1
pauses	1.1568	3
utterance	0.9499	0
cliches	0.2556	3
interpersonal exchanges	0.0082	3
brief debates	0.0003	3
Telic		
exchange	4.2769	3
establish	3.3530	3
further	3.2694	0
allow	3.2489	3
create	2.7141	0
generate	2.0107	0
get	1.9484	0
gloss	0.4780	0
exchange information	0.2313	3
exchange ideas	0.1896	3
enable people	0.1151	3
pass time	0.0469	0
teach skills	0.0171	3

Figure 4: Weighted Qualia Structures for *book*, *computer* and *conversation*

Data Mining		
Formal		
data analysis	2.1492	3
intelligence	1.4242	0
analysis	1.2009	3
tool	1.1987	3
prediction	0.9682	3
approach	0.7279	3
speciality	0.6245	3
system	0.6018	3
application	0.5209	3
functionality	0.3974	3
process	0.3840	3
mechanism	0.3503	3
type	0.3372	0
practice	0.3310	3
technology	0.3240	3
activity	0.3207	3
employment	0.2565	0
use	0.2128	3
name	0.1944	3
area	0.1856	0
datum	0.1701	0
data warehousing technologies	0.1497	3
subject	0.1403	0
information process	0.0498	3
information process techniques	0.0005	3
Agentive		
design		3
Constitutive		
knowledge	0.7062	3
Telic		
connect	0.5949	0
achieve	0.3651	3
uncover	0.3460	3
research	0.3374	3
answer	0.2122	3
support	0.2025	3
look	0.1834	0
provide information	0.1527	3
search	0.1451	3
tell	0.1099	1
identify patterns	0.0959	3
discover patterns	0.0934	3
identify trends	0.0765	3
provide a foundation	0.0620	1
improve services	0.0559	3
gain business intelligence	0.0048	3
explore knowledge	0.0045	3
detect dependencies	0.0036	3
gain business	0.0223	1
analyse large volumes	0.0022	1
find new prospects	0.0011	3
analyze disparate customer data	0.0002	3

Figure 5: Weighted Qualia Structure for *data mining*

Natural Language Processing		
Formal		
linguistics	1.0047	3
technique	0.4983	3
intelligence	0.3559	3
method	0.2748	3
model	0.1847	3
aspect	0.1380	3
scheme	0.1258	3
system	0.0750	1
research	0.0636	3
application	0.0603	3
science	0.0536	3
technology	0.0414	3
area	0.0373	0
product	0.0337	0
document processing applications	0.0174	3
Agentive		
design		3
Constitutive		
Telic		
build	0.1037	3
keep track	0.0820	3
understand	0.0662	3
soften	0.0501	0
provide	0.0384	3
build tailored knowledge base	0.0008	3

Figure 6: Weighted Qualia Structure for *natural language processing*

ternational Conference on Computational Linguistics, pages 539–545.

L.M. Iwanska, N. Mata, and K. Kruger. 2000. Fully automatic acquisition of taxonomic knowledge from large corpora of texts. In L.M. Iwanksa and S.C. Shapiro, editors, *Natural Language Processing and Knowledge Processing*, pages 335–345. MIT/AAAI Press.

M. Johnston and F. Busa. 1996. Qualia structure and the compositional interpretation of compounds.

F. Keller, M. Lapata, and O. Ourioupina. 2002. Using the web to overcome data sparseness. In *Proceedings of EMNLP-02*, pages 230–237.

F. Kronlid. 2003. Modes of explanation - aristotelian philosophy and pustejovskyan linguistics. Ms. University of Gteborg.

K. Markert, N. Modjeska, and M. Nissim. 2003. Using the web for nominal anaphora resolution. In *EACL Workshop on the Computational Treatment of Anaphora*.

M. Poesio, T. Ishikawa, S. Schulte im Walde, and R. Viera. 2002. Acquiring lexical knowledge for anaphora resolution. In *Proceedings of the 3rd Conference on Language Resources and Evaluation*.

J. Pustejovsky, P. Anick, and S. Bergler. 1993. Lexical semantic techniques for corpus analysis. *Computational Lingustics, Special Issue on Using Large Corpora II*, 19(2):331–358.

J. Pustejovsky. 1991. The generative lexicon. *Computational Linguistics*, 17(4):209–441.

P. Resnik and A. Elkiss. 2003. The linguist's search engine: Getting started guide. Technical Report LAMP-TR-108/CS-TR-4541/UMIACS-TR-2003-109, University of Maryland, College Park, November.

P. Resnik and N. Smith. 2003. The web as a parallel corpus. *Computational Lingusitics*, 29(3):349–380.

D. Tufis and O. Mason. 1998. Tagging Romanian Texts: a Case Study for QTAG, a Language Independent Probabilistic Tagger. In *Proceedings of the First International Conference on Language Resources and Evaluation (LREC)*, pages 589–96.

E.M. Voorhees. 1994. Query expansion using lexical-semantic relations. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 61–69.

I. Yamada and T. Baldwin. 2004. Automatic discovery of telic and agentive roles from corpus data. In *Proceedings of the The 18th Pacific Asia Conference on Language, Information and Computation (PACLIC 18)*.

Automatically Distinguishing Literal and Figurative Usages of Highly Polysemous Verbs

Afsaneh Fazly and Ryan North and Suzanne Stevenson
Department of Computer Science
University of Toronto
{afsaneh,ryan,suzanne}@cs.toronto.edu

Abstract

We investigate the meaning extensions of very frequent and highly polysemous verbs, both in terms of their compositional contribution to a light verb construction (LVC), and the patterns of acceptability of the resulting LVC. We develop compositionality and acceptability measures that draw on linguistic properties specific to LVCs, and demonstrate that these statistical, corpus-based measures correlate well with human judgments of each property.

1 Introduction

Due to a cognitive priority for concrete, easily visualizable entities, abstract notions are often expressed in terms of more familiar and concrete things and situations (Newman, 1996; Nunberg et al., 1994). This gives rise to a widespread use of metaphor in language. In particular, certain verbs easily undergo a process of metaphorization and meaning extension (e.g., Pauwels, 2000; Newman and Rice, 2004). Many such verbs refer to states or acts that are central to human experience (e.g., *sit*, *put*, *give*); hence, they are often both highly polysemous and highly frequent. An important class of verbs prone to metaphorization are **light verbs**, on which we focus in this paper.

A light verb, such as *give*, *take*, or *make*, combines with a wide range of complements from different syntactic categories (including nouns, adjectives, and prepositions) to form a new predicate called a **light verb construction** (LVC). Examples of LVCs include:

1. (a) Azin *took a walk* along the river.
   (b) Sam *gave a speech* to a few students.
   (c) Joan *takes care* of him when I am away.
   (d) They *made good* on their promise to win.
   (e) You should always *take* this *into account*.

The light verb component of an LVC is "semantically bleached" to some degree; consequently, the semantic content of an LVC is assumed to be determined primarily by the complement (Butt, 2003). Nevertheless, light verbs exhibit meaning variations when combined with different complements. For example, *give* in *give (someone) a present* has a literal meaning, i.e., "transfer of possession" of a THING to a RECIPIENT. In *give a speech*, *give* has a figurative meaning: an abstract entity (*a speech*) is "transferred" to the audience, but no "possession" is involved. In *give a groan*, the notion of transfer is even further diminished.

Verbs exhibiting such meaning variations are widespread in many languages. Hence, successful NLP applications—especially those requiring some degree of semantic interpretation—need to identify and treat them appropriately. While figurative uses of a light verb are indistinguishable on the surface from a literal use, this distinction is essential to a machine translation system, as Table 1 illustrates. It is therefore important to determine automatic mechanisms for distinguishing literal and figurative uses of light verbs.

Moreover, in their figurative usages, light verbs tend to have similar patterns of cooccurrence with semantically similar complements (e.g., Newman, 1996). Each similar group of complement nouns can even be viewed as a possible meaning extension for a light verb. For example, in *give advice*, *give orders*, *give a speech*, etc., *give* contributes a notion of

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 38–47,
Ann Arbor, June 2005.

Sentence in English	Intermediate semantics	Translation in French
Azin gave Sam a book.	(e1/give :agent (a1/“Azin”) :theme (b1/“book”) :recepient (s1/“Sam”))	Azin a donné un livre à Sam. Azin gave a book to Sam.
Azin gave the lasagna a try.	(e2/give-a-try≈try :agent (a1/“Azin”) :theme (l1/“lasagna”))	Azin a essayé le lasagne. Azin tried the lasagna.

Table 1: Sample sentences with literal and figurative usages of *give*.

“abstract transfer”, while in *give a groan*, *give a cry*, *give a moan*, etc., *give* contributes a notion of “emission”. There is much debate on whether light verbs have one highly abstract (underspecified) meaning, further determined by the context, or a number of identifiable (related) subsenses (Pustejovsky, 1995; Newman, 1996). Under either view, it is important to elucidate the relation between possible interpretations of a light verb and the sets of complements it can occur with.

This study is an initial investigation of techniques for the automatic discovery of meaning extensions of light verbs in English. As alluded to above, we focus on two issues: (i) the distinction of literal versus figurative usages, and (ii) the role of semantically similar classes of complements in refining the figurative meanings.

In addressing the first task, we note the connection between the literal/figurative distinction and the degree to which a light verb contributes compositionally to the semantics of an expression. In Section 2, we elaborate on the syntactic properties that relate to the compositionality of light verbs, and propose a statistical measure incorporating these properties, which places light verb usages on a continuum of meaning from literal to figurative. Figure 1(a) depicts such a continuum in the semantic space of *give*, with the literal usages represented as the core.

The second issue above relates to our long-term goal of dividing the space of figurative uses of a light verb into semantically coherent segments, as shown in Figure 1(b). Section 3 describes our hypothesis on the class-based nature of the ability of potential complements to combine with a light verb. At this point we cannot spell out the different figurative meanings of the light verb associated with such classes. We take a preliminary step in proposing a statistical measure of the acceptability of a combination of a light verb and a class of complements, and explore the extent to which this measure can reveal class-based behaviour.

Subsequent sections of the paper present the corpus extraction methods for estimating our compositionality and acceptability measures, the collection of human judgments to which the measures will be compared, experimental results, and discussion.

2 Compositionality of Light Verbs

2.1 Linguistic Properties: Syntactic Flexibility

We focus on a broadly-documented subclass of light verb constructions, in which the complement is an activity noun that is often the main source of semantic predication (Wierzbicka, 1982). Such complements are assumed to be indefinite, non-referential **predicative nominals** (PNs) that are often morphologically related to a verb (see the complements in examples (1a–c) above). We refer to this class of light verb constructions as “LV+PN” constructions, or simply LVCs.

There is much linguistic evidence that semantic properties of a lexical item determine, to a large extent, its syntactic behaviour (e.g., Rappaport Hovav and Levin, 1998). In particular, the degree of compositionality (decomposability) of a multiword expression has been known to affect its participation in syntactic transformations, i.e., its syntactic flexibility (e.g., Nunberg et al., 1994). English “LV+PN” constructions enforce certain restrictions on the syntactic freedom of their noun components (Kearns, 2002). In some, the noun may be introduced by a definite article, pluralized, passivized, relativized, or even *wh*-questioned:

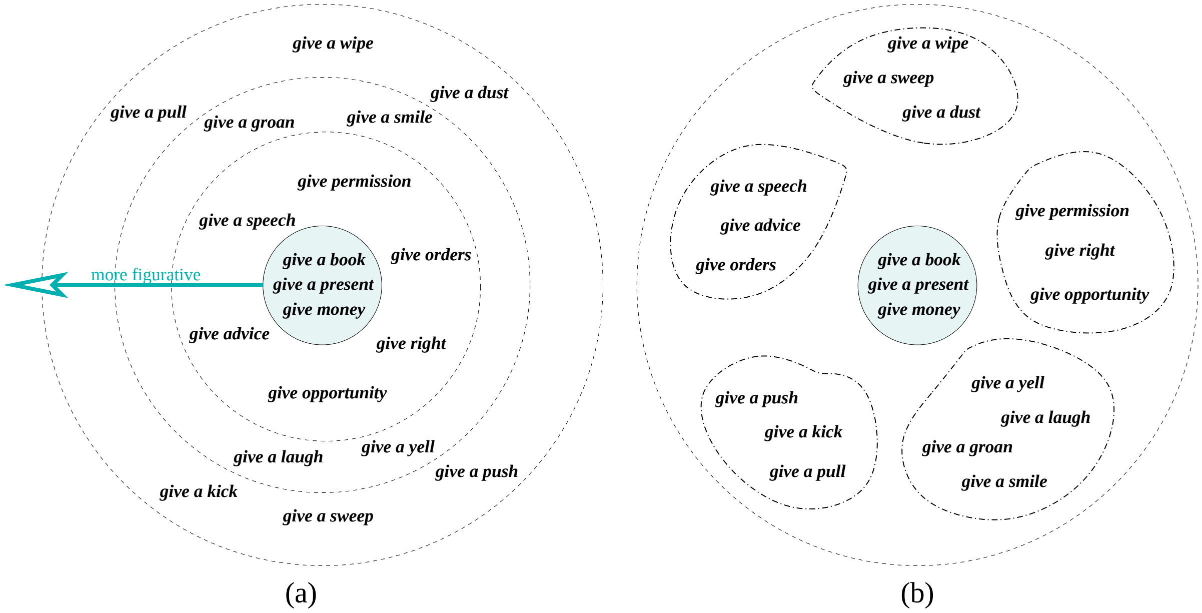


Figure 1: Two possible partitionings of the semantic space of *give*.

2. (a) Azin *gave a speech* to a few students.
   (b) Azin *gave the speech* just now.
   (c) Azin *gave* a couple of *speeches* last night.
   (d) *A speech* was *given* by Azin just now.
   (e) *Which speech* did Azin *give*?

Others have little or no syntactic freedom:

3. (a) Azin *gave a groan* just now.
   (b) * Azin *gave the groan* just now.
   (c) ? Azin *gave* a couple of *groans* last night.
   (d) * *A groan* was *given* by Azin just now.
   (e) * *Which groan* did Azin *give*?

Recall that *give* in *give a groan* is presumed to be a more abstract usage than *give* in *give a speech*. In general, the degree to which the light verb retains aspects of its literal meaning—and contributes them compositionally to the LVC—is reflected in the degree of syntactic freedom exhibited by the LVC. We exploit this insight to devise a statistical measure of compositionality, which uses evidence of syntactic (in)flexibility of a potential LVC to situate it on a scale of literal to figurative usage of the light verb: i.e., the more inflexible the expression, the more figurative (less compositional) the meaning.

2.2 A Statistical Measure of Compositionality

Our proposed measure quantifies the degree of syntactic flexibility of a light verb usage by looking at its frequency of occurrence in any of a set of relevant syntactic patterns, such as those in examples (2) and (3). The measure, $\textsc{Comp}(LV,N)$, assigns a score to a given combination of a light verb (LV) and a noun (N):

$$\begin{aligned}&\textsc{Comp}(LV,N) = \\ &\ \textsc{Assoc}(LV;N) + \\ &\ \textsc{Diff}\ (\textsc{Assoc}(LV;N,PS_{pos}),\textsc{Assoc}(LV;N,PS_{neg}))\end{aligned}$$

That is, the greater the association between LV and N, and the greater the difference between their association with positive syntactic patterns and negative syntactic patterns, the more figurative the meaning of the light verb, and the higher the score.

The strength of the association between the light verb and the complement noun is measured using pointwise mutual information (PMI) whose standard formula is given here:[1]

$$\begin{aligned}\textsc{Assoc}(LV;N) &= \log\frac{\Pr(LV,N)}{\Pr(LV)\ \Pr(N)} \\ &\approx \log\frac{n\, f(LV,N)}{f(LV)\ f(N)}\end{aligned}$$

where n is an estimate of the total number of verb and object noun pairs in the corpus.

[1] PMI is subject to overestimation for low frequency items (Dunning, 1993), thus we require a minimum frequency of occurrence for the expressions under study.

PS_{pos} represents the set of syntactic patterns preferred by less-compositional (more figurative) LVCs (e.g., as in (3a)), and PS_{neg} represents less preferred patterns (e.g., those in (3b–e)). Typically, these patterns most affect the expression of the complement noun. Thus, to measure the strength of association between an expression and a set of patterns, we use the PMI of the light verb, and the complement noun appearing in all of the patterns in the set, as in:

$$\begin{aligned}\textsc{Assoc}(LV;N,PS_{pos}) &= \mathrm{PMI}\,(LV;N,PS_{pos}) \\ &= \log\frac{\Pr(LV,N,PS_{pos})}{\Pr(LV)\ \Pr(N,PS_{pos})} \\ &\approx \log\frac{n\,f(LV,N,PS_{pos})}{f(LV)\ f(N,PS_{pos})}\end{aligned}$$

in which counts of occurrences of N in syntactic contexts represented by PS_{pos} are summed over all patterns in the set. $\textsc{Assoc}(LV;N,PS_{neg})$ is defined analogously using PS_{neg} in place of PS_{pos}.

DIFF measures the difference between the association strengths of the positive and negative pattern sets, referred to as $\textsc{Assoc}_{pos}$ and $\textsc{Assoc}_{neg}$, respectively. Our calculation of ASSOC uses maximum likelihood estimates of the true probabilities. To account for resulting errors, we compare the two confidence intervals, $[\textsc{Assoc}_{pos} \pm \Delta\textsc{Assoc}_{pos}]$ and $[\textsc{Assoc}_{neg} \pm \Delta\textsc{Assoc}_{neg}]$, as in Lin (1999). We take the minimum distance between the two as a conservative estimate of the true difference:

$$\begin{aligned}&\textsc{Diff}(\textsc{Assoc}(LV;N,PS_{pos}),\ \textsc{Assoc}(LV;N,PS_{neg})) \approx \\ &\quad (\textsc{Assoc}_{pos} - \Delta\textsc{Assoc}_{pos}) \\ &\quad -(\textsc{Assoc}_{neg} + \Delta\textsc{Assoc}_{neg})\end{aligned}$$

Taking the difference between confidence intervals lessens the effect of differences that are not statistically significant. (The confidence level, $1-\alpha$, is set to 95% in all experiments.)

3 Acceptability Across Semantic Classes

3.1 Linguistic Properties: Class Behaviour

In this aspect of our work, we narrow our focus onto a subclass of "LV+PN" constructions that have a PN complement in a stem form identical to a verb, preceded (typically) by an indefinite determiner (as in (1a–b) above). Kearns (2002), Wierzbicka (1982), and others have noted that the way in which LVs combine with such PNs to form acceptable LVCs is semantically patterned—that is, PNs with similar semantics appear to have the same trends of cooccurrence with an LV.

Our hypothesis is that semantically similar LVCs—i.e., those formed from an LV plus any of a set of semantically similar PNs—distinguish a figurative subsense of the LV. In the long run, if this is true, it could be exploited by using class information to extend our knowledge of acceptable LVCs and their likely meaning (cf. such an approach to verb particle constructions by Villavicencio, 2003).

As steps to achieving this long-term goal, we must first devise an acceptability measure which determines, for a given LV, which PNs it successfully combines with. We can even use this measure to provide evidence on whether the hypothesized class-based behaviour holds, by seeing if the measure exhibits differing behaviour across semantic classes of potential complements.

3.2 A Statistical Measure of Acceptability

We develop a probability formula that captures the likelihood of a given LV and PN forming an acceptable LVC. The probability depends on both the LV and the PN, and on these elements being used in an LVC:

$$\begin{aligned}&\textsc{Acpt}(LV,PN) \\ &\quad = \Pr(LV,PN,LVC) \\ &\quad = \Pr(PN)\ \Pr(LVC|PN)\ \Pr(LV|PN,LVC)\end{aligned}$$

The first factor, $\Pr(PN)$, reflects the linguistic observation that higher frequency words are more likely to be used as LVC complements (Wierzbicka, 1982). We estimate this factor by $f(PN)/n$, where n is the number of words in the corpus.

The probability that a given LV and PN form an acceptable LVC further depends on how likely it is that the PN combines with *any* light verbs to form an LVC. The frequency with which a PN forms LVCs is estimated as the number of times we observe it in the prototypical "LV a/an PN" pattern across LVs. (Note that such counts are an overestimate, since we cannot determine which usages are indeed LVCs vs. literal uses of the LV.) Since these counts consider the PN only in the context of an indefinite determiner, we normalize over counts of "a/an PN" (noted as *aPN*) to form the conditional probability estimate of the second factor:

$$\Pr(LVC|PN) \approx \frac{\sum_{i=1}^{v} f(LV_i, aPN)}{f(aPN)}$$

where v is the number of light verbs considered.

The third factor, $\Pr(LV|PN, LVC)$, reflects that different LVs have varying degrees of acceptability when used with a given PN in an LVC. We similarly estimate this factor with counts of the given LV and PN in the typical LVC pattern: $f(LV, aPN)/f(aPN)$.

Combining the estimates of the three factors yields:

$$\textsc{Acpt}(LV, PN) \approx$$
$$\frac{f(PN)}{n} \times \frac{\sum_{i=1}^{v} f(LV_i, aPN)}{f(aPN)} \times \frac{f(LV, aPN)}{f(aPN)}$$

4 Materials and Methods

4.1 Light Verbs

Common light verbs in English include *give*, *take*, *make*, *get*, *have*, and *do*, among others. We focus here on two of them, i.e., *give* and *take*, that are frequently and productively used in light verb constructions, and are highly polysemous. The WordNet polysemy count (number of different senses) of *give* and *take* are 44 and 42, respectively.

4.2 Experimental Expressions

Experimental expressions—i.e., potential LVCs using *give* and *take*—are drawn from two sources. The development and test data used in experiments of compositionality (bncD and bncT, respectively) are randomly extracted from the BNC (BNC Reference Guide, 2000), yielding expressions covering a wide range of figurative usages of *give* and *take*, with complements from different semantic categories. In contrast, in experiments that involve acceptability, we need figurative usages of "the same type", i.e., with semantically similar complement nouns, to further examine our hypothesis on the class-based behaviour of light verb combinations. Since in these LVCs the complement is a predicative noun in stem form identical to a verb, we form development and test expressions by combining *give* or *take* with verbs from selected semantic classes of Levin (1993), taken from Stevenson et al. (2004).

4.3 Corpora

We gather estimates for our COMP measure from the BNC, processed using the Collins parser (Collins, 1999) and TGrep2 (Rohde, 2004). Because some LVCs can be rare in classical corpora, our ACPT estimates are drawn from the World Wide Web (the subsection indexed by AltaVista). In our comparison of the two measures, we use web data for both, using a simplified version of COMP. The high level of noise on the web will influence the performance of both measures, but COMP more severely, due to its reliance on comparisons of syntactic patterns.

Web counts are based on an exact-phrase query to AltaVista, with the number of pages containing the search phrase recorded as its frequency.[2] The size of the corpus is estimated at 3.7 billion, the number of hits returned in a search for *the*. These counts are underestimates of the true frequencies, as a phrase may appear more than once in a web page, but we assume all counts to be similarly affected.

4.4 Extraction

Most required frequencies are simple counts of a word or string of words, but the syntactic patterns used in the compositionality measure present some complexity. Recall that PS_{pos} and PS_{neg} are pattern sets representing the syntactic contexts of interest. Each pattern encodes several syntactic attributes: v, the voice of the extracted expression (active or passive); d, the type of the determiner introducing N (definite or indefinite); and n, the number of N (singular or plural). In our experiments, the set of patterns associated with less-compositional use, PS_{pos}, consists of the single pattern with values active, indefinite, and singular, for these attributes. PS_{neg} consists of all patterns with at least one of these attributes having the alternative value.

While our counts on the BNC can use syntactic mark-up, it is not feasible to collect counts on the web for some of the pattern attributes, such as voice. We develop two different variations of the measure, one for BNC counts, and a simpler one for

[2] All searches were performed March 15–30, 2005.

	give		*take*	
Human Ratings	bncD	bncT	bncD	bncT
‘low’	20	10	36	19
‘medium’	35	16	9	5
‘high’	24	10	27	10
Total	79	36	72	34

Table 2: Distribution of development and test expressions with respect to human compositionality ratings.

web counts. We thus subscript COMP with abbreviations standing for each attribute in the measure: COMP_{vdn} for a measure involving all three attributes (used on BNC data), and COMP_{d} for a measure involving determiner type only (used on web data).

5 Human Judgments

5.1 Judgments of Compositionality

To determine how well our proposed measure of compositionality captures the degree of literal/figurative use of a light verb, we compare its scores to human judgments on compositionality. Three judges (native speakers of English with sufficient linguistic knowledge) answered yes/no questions related to the contribution of the literal meaning of the light verb within each experimental expression. The combination of answers to these questions is transformed to numerical ratings, ranging from 0 (fully non-compositional) to 4 (largely compositional). The three sets of ratings yield linearly weighted Kappa values of .34 and .70 for *give* and *take*, respectively. The ratings are averaged to form a consensus set to be used for evaluation.[3]

The lists of rated expressions were biased toward figurative usages of *give* and *take*. To achieve a spectrum of literal to figurative usages, we augment the lists with literal expressions having an average rating of 5 (fully compositional). Table 2 shows the distribution of the experimental expressions across three intervals of compositionality degree, ‘low’ (ratings ≤ 1), ‘medium’ ($1 <$ ratings < 3), and ‘high’ (ratings ≥ 3). Table 3 presents sample expressions with different levels of compositionality ratings.

	Sample Expressions	
Human Ratings	*give*	*take*
‘low’	*give a squeeze*	*take a shower*
‘medium’	*give help*	*take a course*
‘high’	*give a dose*	*take an amount*

Table 3: Sample expressions with different levels of compositionality ratings.

5.2 Judgments of Acceptability

Our acceptability measure is compared to the human judgments gathered by Stevenson et al. (2004). Two expert native speakers of English rated the acceptability of each potential “LV+PN” construction generated by combining *give* and *take* with candidate complements from the development and test Levin classes. Ratings were from 1 (unacceptable) to 5 (completely natural; this was capped at 4 for test data), allowing for “in-between” ratings as well, such as 2.5. On test data, the two sets of ratings yielded linearly weighted Kappa values of .39 and .72 for *give* and *take*, respectively. (Interestingly, a similar agreement pattern is found in our human compositionality judgments above.) The consensus set of ratings was formed from an average of the two sets of ratings, once disagreements of more than one point were discussed.

6 Experimental Results

To evaluate our compositionality and acceptability measures, we compare them to the relevant consensus human ratings using the Spearman rank correlation coefficient, r_s. For simplicity, we report the absolute value of r_s for all experiments. Since in most cases, correlations are statistically significant ($p \ll .01$), we omit p values; those r_s values for which p is marginal (i.e., $.01 \leq p \leq .10$) are subscripted with an “m” in the tables. Correlation scores in boldface are those that show an improvement over the baseline, PMI_{LVC}.

The PMI_{LVC} measure is an informed baseline, since it draws on properties of LVCs. Specifically, PMI_{LVC} measures the strength of the association between a light verb and a noun appearing in syntactic patterns preferred by LVCs, i.e., $\text{PMI}_{\text{LVC}} = \text{PMI}\,(LV; N, PS_{pos})$. Assuming that an acceptable LVC forms a detectable collocation, PMI_{LVC} can be interpreted as an informed baseline for degree of acceptability. PMI_{LVC} can also

[3] We asked the judges to provide short paraphrases for each expression, and only use those expressions for which the majority of judges expressed the same sense.

			$\textsc{pmi}_{\textsc{lvc}}$	$\textsc{Comp}_{vdn}$
LV	Data Set	n	r_s	r_s
give	bncT	36	.62	.57
	bncDT	114	.68	**.70**
	bncDT/a	79	.68	**.75**
take	bncT	34	.51	**.59**
	bncDT	106	.52	**.61**
	bncDT/a	68	.63	**.72**

Table 4: Correlations (r_s; n = # of items) between human compositionality ratings and $\textsc{Comp}$ measure (counts from BNC).

	Levin class:	18.1,2	30.3	43.2
LV		n=35	n=18	n=35
give	% fair/good ratings	51	44	54
	log of mean $\textsc{Acpt}$	-6	-4	-5
take	% fair/good ratings	23	28	3
	log of mean $\textsc{Acpt}$	-4	-3	-6

Table 5: Comparison of the proportion of human ratings considered “fair” or “good” in each class, and the $\log_{10}$ of the mean $\textsc{Acpt}$ score for that class.

be considered as a baseline for the degree of compositionality of an expression (with respect to the light verb component), under the assumption that the less compositional an expression, the more its components appear as a fixed collocation.

6.1 Compositionality Results

Table 4 displays the correlation scores of the human compositionality ratings with $\textsc{Comp}_{vdn}$, our compositionality measure estimated with counts from the BNC. Given the variety of light verb usages in expressions used in the compositionality data, we report correlations not only on test data (bncT), but also on development and test data combined (bncDT) to get more data points and hence more reliable correlation scores. Compared to the baseline, $\textsc{Comp}_{vdn}$ has generally higher correlations with human ratings of compositionality.

There are two different types of expressions among those used in compositionality experiments: expressions with an indefinite determiner *a* (e.g., *give a kick*) and those without a determiner (e.g., *give guidance*). Despite shared properties, the two types of expressions may differ with respect to syntactic flexibility, due to differing semantic properties of the noun complements in the two cases. We thus calculate correlation scores for expressions with the indefinite determiner only, from both development and test data (bncDT/a). We find that $\textsc{Comp}_{vdn}$ has higher correlations (and larger improvements over the baseline) on this subset of expressions. (Note that there are comparable numbers of items in bncDT and bncDT/a, and the correlation scores are highly significant—very small *p* values—in both cases.)

To explore the effect of using a larger but noisier corpus, we compare the performance of $\textsc{Comp}_{vdn}$ with $\textsc{Comp}_{d}$, the compositionality measure using web data. The correlation scores for $\textsc{Comp}_{d}$ on bncDT are .41 and .35, for *give* and *take*, respectively, compared to a baseline (using web counts) of .37 and .32. We find that $\textsc{Comp}_{vdn}$ has significantly higher correlation scores (larger r_s and much smaller *p* values), as well as larger improvements over the baseline. This is a confirmation that using more syntactic information, from less noisy data, improves the performance of our compositionality measure.[4]

6.2 Acceptability Results

We have two goals in assessing our $\textsc{Acpt}$ measure: one is to demonstrate that the measure is indeed indicative of the level of acceptability of an LVC, and the other is to explore whether it helps to indicate class-based patterns of acceptability.

Regarding the latter, Stevenson et al. (2004) found differing overall levels of (human) acceptability for different Levin classes combined with *give* and *take*. This indicates a strong influence of semantic similarity on the possible LV and complement combinations. Our $\textsc{Acpt}$ measure also yields differing patterns across the semantic classes. Table 5 shows, for each light verb and test class, the proportion of acceptable LVCs according to human ratings, and the log of the mean $\textsc{Acpt}$ score for that LV and class combination. For *take*, the $\textsc{Acpt}$ score generally reflects the difference in proportion of accepted expressions according to the human ratings, while for *give*, the measure is less consistent. (The three development classes show the same pattern.) The $\textsc{Acpt}$ measure thus appears to reflect the differing patterns of acceptability across the classes, at least

[4] Using the automatically parsed BNC as a source of less noisy data improves performance. However, since these constructions may be infrequent with any particular complement, we do not expect the use of cleaner but more plentiful text (such as existing treebanks) to improve the performance any further.

LV	Levin Class	n	PMI$_{\mathrm{LVC}}$ r_s	ACPT r_s
give	18.1,2	35	$.39_{\mathrm{m}}$	**.55**
	30.3	18	$.38_{\mathrm{m}}$	**.73**
	43.2	35	$.30_{\mathrm{m}}$	$\mathbf{.34}_{\mathrm{m}}$
take	18.1.2	35	.57	**.61**
	30.3	18	.55	**.64**
	43.2	35	.43	**.47**

Table 6: Correlations (r_s; n = # of items) between acceptability measures and consensus human ratings (counts from web).

Human Ratings	LV	n	PMI$_{\mathrm{LVC}}$ r_s	ACPT r_s	COMP$_d$ r_s
accept.	*give*	88	.31	**.42**	**.40**
(Levin)	*take*	88	.58	**.61**	.56
compos.	*give*	114	.37	$.21_{\mathrm{m}}$	**.41**
(bncDT)	*take*	106	.32	.30	**.35**

Table 7: Correlations (r_s; n = # of items) between each measure and each set of human ratings (counts from web).

for *take*.

To get a finer-grained notion of the degree to which ACPT conforms with human ratings, we present correlation scores between the two, in Table 6. The results show that ACPT has higher correlation scores than the baseline—substantially higher in the case of *give*. The correlations for *give* also vary more widely across the classes.

These results together indicate that the acceptability measure may be useful, and indeed taps into some of the differing levels of acceptability across the classes. However, we need to look more closely at other linguistic properties which, if taken into account, may improve the consistency of the measure.

6.3 Comparing the Two Measures

Our two measures are intended for different purposes, and indeed incorporate differing linguistic information about LVCs. However, we also noted that PMI$_{\mathrm{LVC}}$ can be viewed as a baseline for both, indicating some underlying commonality. It is worth exploring whether each measure taps into the different phenomena as intended. To do so, we correlate COMP with the human ratings of acceptability, and ACPT with the human ratings of compositionality, as shown in Table 7. (The formulation of the ACPT measure here is adapted for use with determiner-less LVCs.) For comparability, both measures use counts from the web. The results confirm that COMP$_d$ correlates better than does ACPT with compositionality ratings, while ACPT correlates best with acceptability ratings.

7 Discussion and Concluding Remarks

Recently, there has been increasing awareness of the need for appropriate handling of multiword expressions (MWEs) in NLP tasks (Sag et al., 2002). Some research has concentrated on the automatic acquisition of semantic knowledge about certain classes of MWEs, such as compound nouns or verb particle constructions (VPCs) (e.g., Lin, 1999; McCarthy et al., 2003; Villavicencio, 2003). Previous research on LVCs, on the other hand, has primarily focused on their automatic extraction (e.g., Grefenstette and Teufel 1995; Dras and Johnson 1996; Moirón 2004; though see Stevenson et al. 2004).

Like most previous studies that focus on semantic properties of MWEs, we are interested in the issue of compositionality. Our COMP measure aims to identify a continuum along which a light verb contributes to the semantics of an expression. In this way, our work combines aspects of earlier work on VPC semantics. McCarthy et al. (2003) determine a continuum of compositionality of VPCs, but do not distinguish the contribution of the individual components. Bannard et al. (2003), on the other hand, look at the separate contribution of the verb and particle, but assume that a binary decision on the compositionality of each is sufficient.

Previous studies determine compositionality by looking at the degree of distributional similarity between an expression and its component words (e.g., McCarthy et al., 2003; Bannard et al., 2003; Baldwin et al., 2003). Because light verbs are highly polysemous and frequently used in LVCs, such an approach is not appropriate for determining their contribution to the semantics of an expression. We instead examine the degree to which a light verb usage is "similar" to the prototypical LVC, through a statistical comparison of its behaviour within different syntactic patterns. Syntactic flexibility and semantic compositionality are known to be strongly correlated for many types of MWEs (Nunberg et al., 1994). We thus intend to extend our approach to include other polysemous verbs with metaphorical extensions.

Our compositionality measure correlates well with the literal/figurative spectrum represented in human judgments. We also aim to determine finer-grained distinctions among the identified figurative usages of a light verb, which appear to relate to the semantic class of its complement. Semantic class knowledge may enable us to elucidate the types of relations between a light verb and its complement such as those determined in the work of Wanner (2004), but without the need for the manually labelled training data which his approach requires. Villavicencio (2003) used class-based knowledge to extend a VPC lexicon, but assumed that an unobserved VPC is not acceptable. We instead believe that more robust application of class-based knowledge can be achieved with a better estimate of the acceptability of various expressions.

Work indicating acceptability of MWEs is largely limited to collocational analysis using PMI-based measures (Lin, 1999; Stevenson et al., 2004). We instead use a probability formula that enables flexible integration of LVC-specific linguistic properties. Our ACPT measure yields good correlations with human acceptability judgments; indeed, the average increase over the baseline is about twice as high as that of the acceptability measure proposed by Stevenson et al. (2004). Although ACPT also somewhat reflects different patterns across semantic classes, the results clearly indicate the need for incorporating more knowledge into the measure to capture class-based behaviour more consistently.

The work presented here is preliminary, but is the first we are aware of to tie together the two issues of compositionality and acceptability, and relate them to the notion of class-based meaning extensions of highly polysemous verbs. Our on-going work is focusing on the role of the noun component of LVCs, to determine the compositional contribution of the noun to the semantics of the expression, and the role of noun classes in influencing the meaning extensions of light verbs.

References

Baldwin, T., Bannard, C., Tanaka, T., and Widdows, D. (2003). An empirical model of multiword expression decomposability. In *Proceedings of the ACL-SIGLEX Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 89–96.

Bannard, C., Baldwin, T., and Lascarides, A. (2003). A statistical approach to the semantics of verb-particles. In *Proceedings of the ACL-SIGLEX Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 65–72.

BNC Reference Guide (2000). *Reference Guide for the British National Corpus (World Edition)*, second edition.

Butt, M. (2003). The light verb jungle. Workshop on Multi-Verb Constructions.

Collins, M. (1999). *Head-Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania.

Dras, M. and Johnson, M. (1996). Death and lightness: Using a demographic model to find support verbs. In *Proceedings of the Fifth International Conference on the Cognitive Science of Natural Language Processing*.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Grefenstette, G. and Teufel, S. (1995). Corpus-based method for automatic identification of support verbs for nominalization. In *Proceedings of the 7th Meeting of the EACL*.

Kearns, K. (2002). Light verbs in English. manuscript.

Levin, B. (1993). *English Verb Classes and Alternations: A Preliminary Investigation*. The University of Chicago Press.

Lin, D. (1999). Automatic identification of non-compositional phrases. In *Proceedings of the 37th Annual Meeting of the ACL*, pages 317–324.

McCarthy, D., Keller, B., and Carroll, J. (2003). Detecting a continuum of compositionality in phrasal verbs. In *Proceedings of the ACL-SIGLEX Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*.

Moirón, M. B. V. (2004). Discarding noise in an automatically acquired lexicon of support verb constructions. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*.

Newman, J. (1996). *Give: A Cognitive Linguistic Study*. Mouton de Gruyter.

Newman, J. and Rice, S. (2004). Patterns of usage for English SIT, STAND, and LIE: A cognitively inspired exploration in corpus linguistics. *Cognitive Linguistics*, 15(3):351–396.

Nunberg, G., Sag, I. A., and Wasow, T. (1994). Idioms. *Language*, 70(3):491–538.

Pauwels, P. (2000). *Put, Set, Lay and Place: A Cognitive Linguistic Approach to Verbal Meaning*. LINCOM EUROPA.

Pustejovsky, J. (1995). *The Generative Lexicon*. MIT Press.

Rappaport Hovav, M. and Levin, B. (1998). Building verb meanings. In Butt and Geuder, editors, *The Projection of Arguments: Lexical and Computational Factors*, pages 97–134. CSLI Publications.

Rohde, D. L. T. (2004). TGrep2 User Manual.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for NLP. In *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLING'02)*, pages 1–15.

Stevenson, S., Fazly, A., and North, R. (2004). Statistical measures of the semi-productivity of light verb constructions. In *Proceedings of the ACL-04 Workshop on Multiword Expressions: Integrating Processing*, pages 1–8.

Villavicencio, A. (2003). Verb-particle constructions and lexical resources. In *Proceedings of the ACL-SIGLEX Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 57–64.

Wanner, L. (2004). Towards automatic fine-grained semantic classification of verb-noun collocations. *Natural Language Engineering*, 10(2):95–143.

Wierzbicka, A. (1982). Why can you Have a Drink when you can't *Have an Eat? *Language*, 58(4):753–799.

Automatic Extraction of Idioms using Graph Analysis and Asymmetric Lexicosyntactic Patterns

Dominic Widdows
MAYA Design, Inc.
Pittsburgh, Pennsylvania
widdows@maya.com

Beate Dorow
Institute for Natural Language Processing
University of Stuttgart
dorowbe@IMS.Uni-Stuttgart.DE

Abstract

This paper describes a technique for extracting idioms from text. The technique works by finding patterns such as "thrills and spills", whose reversals (such as "spills and thrills") are never encountered.

This method collects not only idioms, but also many phrases that exhibit a strong tendency to occur in one particular order, due apparently to underlying semantic issues. These include hierarchical relationships, gender differences, temporal ordering, and prototype-variant effects.

1 Introduction

Natural language is full of idiomatic and metaphorical uses. However, language resources such as dictionaries and lexical knowledge bases give at best poor coverage of such phenomena. In many cases, knowledge bases will mistakenly 'recognize' a word and this can lead to more harm than good: for example, a typical mistake of blunt logic would be to assume that "somebody let the cat out of the bag" implied that "somebody let some mammal out of some container."

Idiomatic generation of natural language is, if anything, an even greater challenge than idiomatic language understanding. As pointed out decades ago by Fillmore (1967), a complete knowledge of English requires not only an understanding of the semantics of the word *good*, but also an awareness that this special adjective (alone) can occur with the word *any* to construct phrases like *"Is this paper any good at all?"*, and traditional lexical resources were not designed to provide this information. There are many more general examples occur: for example, "the big bad wolf" sounds right and the "the bad big wolf" sounds wrong, even though both versions are syntactically and semantically plausible. Such examples are perhaps 'idiomatic', though we would perhaps not call them 'idioms', since they are compositional and can sometimes be predicted by general pattern of word-ordering.

In general, the goal of manually creating a complete lexicon of idioms and idiomatic usage patterns in any language is unattainable, and automatic extraction and modelling techniques have been developed to fill this ever-evolving need. Firstly, automatically identifying potential idioms and bringing them to the attention of a lexicographer can be used to improve coverage and reduce the time a lexicographer must spend in searching for such examples. Secondly and more ambitiously, the goal of such work is to enable computers to recognize idioms independently so that the inevitable lack of coverage in language resources does not impede their ability to respond intelligently to natural language input.

In attempting a first-pass at this task, the experiments described in this paper proceed as follows. We focus on a particular class of idioms that can be extracted using *lexicosyntactic patterns* (Hearst, 1992), which are fixed patterns in text that suggest that the words occurring in them have some interesting relationship. The patterns we focus on are occurrences of the form "A and/or B", where A and

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 48–56,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

B are both nouns. Examples include “football and cricket” and “hue and cry.” From this list, we extract those examples for which there is a strong preference on the *ordering* of the participants. For example, we do see the pattern “cricket and football,” but rarely if ever encounter the pattern “cry and hue.” Using this technique, 4173 potential idioms were extracted. This included a number of both true idioms, and words that have regular semantic relationships but do appear to have interesting orderings on these relationships (such as earlier before later, strong before weak, prototype before variant).

The rest of this paper is organized as follows. Section 2 elaborates on some of the previous works that motivate the techniques we have used. Section 3 describes the precise method used to extract idioms through their asymmetric appearance in a large corpus. Section 4 presents and analyses several classes of results. Section 5 describes the methods attempted to filter these results into pairs of words that are more and less contextually related to one another. These include a statistical method that analyses the original corpus for evidence of semantic relatedness, and a combinatoric method that relies on link-analysis on the resulting graph structure.

2 Previous and Related Work

This section describes previous work in extracting information from text, and inferring semantic or idiomatic properties of words from the information so derived.

The main technique used in this paper to extract groups of words that are semantically or idiomatically related is a form of lexicosyntactic pattern recognition. Lexicosyntactic patterns were pioneered by Marti Hearst (Hearst, 1992; Hearst and Schütze, 1993) in the early 1990’s, to enable the addition of new information to lexical resources such as WordNet (Fellbaum, 1998). The main insight of this sort of work is that certain regular patterns in word-usage can reflect underlying semantic relationships. For example, the phrase “France, Germany, Italy, and other European countries” suggests that *France*, *Germany* and *Italy* are part of the class of *European countries*. Such hierarchical examples are quite sparse, and greater coverage was later attained by Riloff and Shepherd (1997) and Roark and Charniak (1998) in extracting relations not of hierarchy but of *similarity*, by finding conjunctions or co-ordinations such as “cloves, cinammon, and nutmeg” and “cars and trucks.” This work was extended by Caraballo (1999), who built classes of related words in this fashion and then reasoned that if a hierarchical relationship could be extracted for *any* member of this class, it could be applied to *all* members of the class. This technique can often mistakenly reason across an ambiguous middle-term, a situation that was improved upon by Cederberg and Widdows (2003), by combining pattern-based extraction with contextual filtering using latent semantic analysis.

Prior work in discovering non-compositional phrases has been carried out by Lin (1999) and Baldwin et al. (2003), who also used LSA to distinguish between compositional and non-compositional verb-particle constructions and noun-noun compounds.

At the same time, work in analyzing idioms and asymmetry within linguistics has become more sophisticated, as discussed by Benor and Levy (2004), and many of the semantic factors underlying our results can be understood from a sophisticated theoretical perspective.

Other motivating and related themes of work for this paper include collocation extraction and example based machine translation. In the work of Smadja (1993) on extracting collocations, preference was given to constructions whose constituents appear in a fixed order, a similar (and more generally implemented) version of our assumption here that asymmetric constructions are more idiomatic than symmetric ones. Recent advances in example-based machine translation (EBMT) have emphasized the fact that examining patterns of language use can significantly improve idiomatic language generation (Carl and Way, 2003).

3 The Symmetric Graph Model as used for Lexical Acquisition and Idiom Extraction

This section of the paper describes the techniques used to extract potentially idiomatic patterns from text, as deduced from previously successful experiments in lexical acquisition.

The main extraction technique is to use lexicosyntactic patterns of the form “A, B and/or C” to find nouns that are linked in some way. For example, consider the following sentence from the British National Corpus (BNC).

> **Ships** laden with **nutmeg**, **cinnamon**, **cloves** or **coriander** once battled the Seven **Seas** to bring **home** their precious **cargo**.

Since the BNC is tagged for parts-of-speech, we know that the words highlighted in bold are nouns. Since the phrase “nutmeg, cinnamon, cloves or coriander” fits the pattern “A, B, C or D”, we create nodes for each of these nouns and create links between them all. When applied to the whole of the BNC, these links can be aggregated to form a graph with 99,454 nodes (nouns) and 587,475 links, as described by Widdows and Dorow (2002). This graph was originally used for lexical acquisition, since clusters of words in the graph often map to recognized semantic classes with great accuracy ($> 80\%$, (Widdows and Dorow, 2002)).

However, for the sake of smoothing over sparse data, these results made the assumption that the links between nodes were *symmetric*, rather than *directed*. In other words, when the pattern “A and/or B” was encountered, a link from A to B *and* a link from B to A was introduced. The nature of symmetric and antisymmetric relationships is examined in detail by Widdows (2004). For the purposes of this paper, it suffices to say that the assumption of symmetry (like the assumption of transitivity) is a powerful tool for improving recall in lexical acquisition, but also leads to serious lapses in precision if the directed nature of links is overlooked, especially if symmetrized links are used to infer semantic similarity.

This problem was brought strikingly to our attention by the examples in Figure 1. In spite of appearing to be a circle of related concepts, many of the nouns in this group are not similar at all, and many of the links in this graph are derived from very very different contexts. In Figure 1, *cat* and *mouse* are linked (they are re both animals and the phrase “cat and mouse” is used quite often): but then *mouse* and *keyboard* are also linked because they are both objects used in computing. A *keyboard*, as well as being a typewriter or computer keyboard, is also used to mean (part of) a musical instrument such as an organ or piano, and *keyboard* is linked to *violin*. A *violin* and a *fiddle* are the same instrument (as often happens with synonyms, they don’t appear together often but have many neighbours in common). The unlikely circle is completed (it turns out) because of the phrase from the nursery rhyme

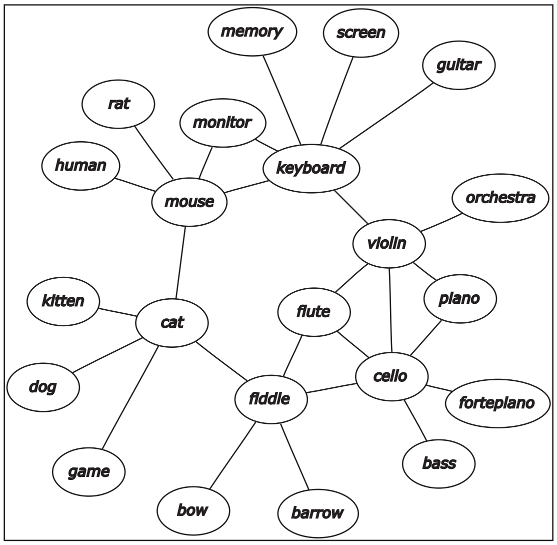


Figure 1: A cluster involving several idiomatic links

Hey diddle diddle,
The cat and the fiddle,
The cow jumped over the moon;

It became clear from examples such as these that idiomatic links, like ambiguous words, were a serious problem when using the graph model for lexical acquisition. However, with ambiguous words, this obstacle has been gradually turned into an opportunity, since we have also developed ways to used the apparent flaws in the model to detect which words are ambiguous in the first place (Widdows, 2004, Ch 4). It is now proposed that we can take the same opportunity for certain idioms: that is, to use the properties of the graph model to work out which links arise from idiomatic usage rather than semantic similarity.

3.1 Idiom Extraction by Recognizing Asymmetric Patterns

The link between the *cat* and *fiddle* nodes in Figure 1 arises from the phrase “the cat and the fiddle.”

Table 1: Sample of asymmetric pairs extracted from the BNC.

First word	Second word
highway	byway
cod	haddock
composer	conductor
wood	charcoal
element	compound
assault	battery
north	south
rock	roll
god	goddess
porgy	bess
middle	class
war	aftermath
god	hero
metal	alloy
salt	pepper
mustard	cress
stocking	suspender
bits	bobs
stimulus	response
committee	subcommittee
continent	ocean

However, no corpus examples were ever found of the converse phrase, "the fiddle and the cat." In cases like these, it may be concluded that placing a *symmetric* link between these two nodes is a mistake. Instead, a *directed* link may be more appropriate.

We therefore formed the hypothesis that if the phrase "A and/or B" occurs frequently in a corpus, but the phrase "B and/or A" is absent, then the link between A and B should be attributed to idiomatic usage rather than semantic similarity.

The next step was to rebuild, finding those relationships that have a strong preference for occurring in a fixed order. Sure enough, several British English idioms were extracted in this way. However, several other kinds of relationships were extracted as well, as shown in the sample in Table 1.[1]

After extracting these pairs, groups of them were gathered together into *directed subgraphs*.[2] Some of these directed subgraphs are reporduced in the analysis in the following section.

4 Analysis of Results

The experimental results include representatives of several types of asymmetric relationships, including the following broad categories.

'True' Idioms

There are many results that display genuinely idiomatic constructions. By this, we mean phrases that have an explicitly lexicalized nature that a native speaker may be expected to recognize as having a special reference or significance. Examples include the following:

thrills and spills
bread and circuses
Punch and Judy
Porgy and Bess
lies and statistics
cat and fiddle
bow and arrow
skull and crossbones

This category is quite loosely defined. It includes

1. historic quotations such as "lies, damned lies and statistics"[3] and "bread and circuses."[4]

2. titles of well-known works.

3. colloquialisms.

4. groups of objects that have become fixed nominals in their own right.

All of these types share the common property that any NLP system that encounters such groups, in order to behave correctly, should recognize, generate, or translate them as phrases rather than words.

Hierarchical Relationships

Many of the asymmetric relationships follow some pattern that may be described as roughly hierarchical. A cluster of examples from two domains is shown in Figure 2. In chess, a rook outranks a bishop, and the phrase "rook and bishop" is encountered much more often than the phrase "bishop and

[1] The sample chosen here was selected by the authors to be representative of some of the main types of results. The complete list can be found at `http://infomap.stanford.edu/graphs/idioms.html`.

[2] These can be viewed at `http://infomap.stanford.edu/graphs/directed_graphs.html`

[3] Attributed to Benjamin Disraeli, certainly popularized by Mark Twain.

[4] A translation of "panem et circenses," from the Roman satirist Juvenal, 1st century AD.

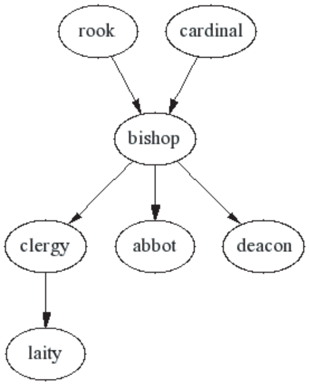


Figure 2: Asymmetric relationships in the chess and church hierarchies

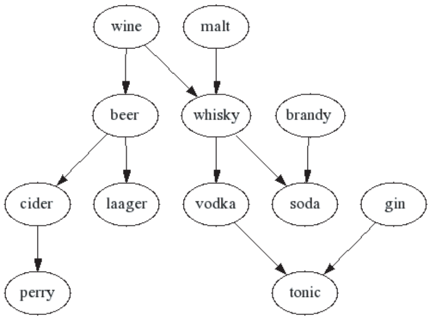


Figure 3: Different beverages, showing their directed relationships

rook.” In the church, a cardinal outranks a bishop, a bishop outranks most of the rest of the clergy, and the clergy (in some senses) outrank the laity.

Sometimes these relationships coincide with figure / ground and agent / patient distinctions. Examples of this kind, as well as “clergy and laity”, include “landlord and tenant”, “employer and employee”, “teacher and pupil”, and “driver and passengers”. An interesting exception is “passengers and crew”, for which we have no semantic explanation.

Pedigree and potency appear to be two other dimensions that can be used to establish the directedness of an idiomatic construction. For example, Figure 3 shows that alcoholic drinks normally appear before their cocktail mixers, but that wine outranks some stronger drinks.

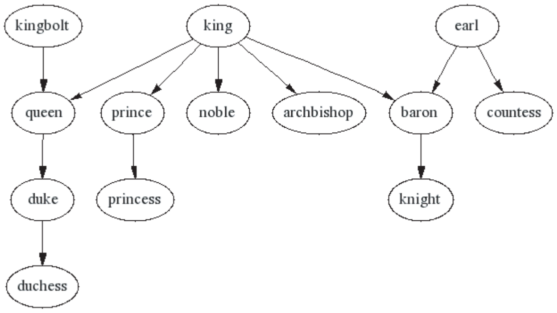


Figure 4: Hierarchical relationships between aristocrats, some of which appear to be gender based

Gender Asymmetry

The relationship between corresponding concepts of different genders also appear to be heavily biased towards appearing in one direction. Many of these relationships are shown in Figure 4. This shows that, in cases where one class outranks another, the higher class appears first, but if the classes are identical, then the male version tends to appear before the female. This pattern is repeated in many pairs of words such as “host and hostess”, “god and goddess”, etc. One exception appears to be in parenting relationships, where female precedes male, as in “mother and father”, “mum and dad”, “grandma and grandpa”.

Temporal Ordering

If one word refers to an event that precedes another temporally or logically, it almost always appears first. The examples in Table 2 were extracted by our experiment. It has been pointed out that for cyclical events, it is perfectly possible that the order of these pairs may be reversed (e.g., “late night and early morning”), though the data we extracted from the BNC showed strong tendencies in the directions given.

A directed subgraph showing many events in human lives in shown in Figure 5.

Prototype precedes Variant

In cases where one participant is regarded as a ‘pure’ substance and the other is a variant or mixture, the pure substance tends to come first. These occur particularly in scientific writing, examples including “element and compound”, “atoms and

Table 2: Pairs of events that have a strong tendency to occur in asymmetric patterns.

Before	After
spring	autumn
morning	afternoon
morning	evening
evening	night
morning	night
beginning	end
question	answer
shampoo	conditioner
marriage	divorce
arrival	departure
eggs	larvae

molecules", "metals and alloys". Also, we see "apples and pears", "apple and plums", and "apples and oranges", suggesting that an apple is a prototypical fruit (in agreement with some of the results of prototype theory; see Rosch (1975)).

Another possible version of this tendency is that core precedes periphery, which may also account for asymmetric ordering of food items such as "fish and chips", "bangers and mash", "tea and coffee" (in the British National Corpus, at least!) In some cases such as "meat and vegetables", a hierarchical or figure / ground distinction may also be argued.

Mistaken extractions

Our preliminary inspection has shown that the extraction technique finds comparatively few genuine mistakes, and the reader is encouraged to follow the links provided to check this claim. However, there are some genuine errors, most of which could be avoided with more sophisticated preprocessing.

To improve recall in our initial lexical acquisition experiments, we chose to strip off modifiers and to stem plural forms to singular forms, so that "apples and green pears" would give a link between *apple* and *pear*.

However, in many cases this is a mistake, because the bracketing should not be of the form "A and (B C)," but of the form "(A and B) C." Using part-of-speech tags alone, we cannot recover this information. One example is the phrase "hardware and software vendors," from which we obtain a link between *hardware* and *vendors*, instead of a link between *hardware* and *software*. A fuller degree of syntactic analysis would improve this situation. For extracting semantic relationships, Cederberg and Widdows (2003) demonstrated that nounphrase chunking does this work very satisfactorily, while being much more tractable than full parsing.

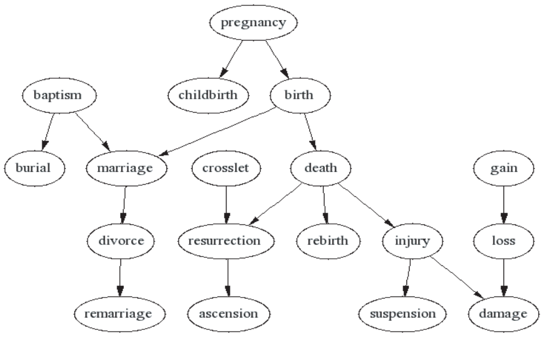


Figure 5: Directed graph showing that life-events are usually ordered temporally when they occur together

The mistaken pair *middle* and *class* shown in Table 1 is another of these mistakes, arising from phrases such as "middle and upper class" and "middle and working class." These examples could be avoided simply by more accurate part-of-speech tagging (since the word "middle" should have been tagged as an adjective in these examples).

This concludes our preliminary analysis of results.

5 Filtering using Latent Semantic Analysis and Combinatoric Analysis

From the results in the previous section, the following points are clear.

1. It is possible to extract many accurate examples of asymmetric constructions, that would be necessary knowledge for generation of natural-sounding language.

2. Some of the pairs extracted are examples of general semantic patterns, others are examples of genuinely idiomatic phrases.

Even for semantically predictable phrases, the fact that the words occur in fixed patterns can be very useful for the purposes of disambiguation, as demonstrated by (Yarowsky, 1995). However, it would be useful to be able to tell which of the asymmetric patterns extracted by our experiments correspond to semantically regular phrases which happen to have a conventional ordering preference, and which phrases correspond to genuine idioms. This final section demonstrates two techniques for performing this filtering task, which show promising results for improving our classification, though should not yet be considered as reliable.

5.1 Filtering using Latent Semantic Analysis

Latent semantic analysis or LSA (Landauer and Dumais, 1997) is by now a tried and tested technique for determining semantic similarity between words by analyzing large corpus (Widdows, 2004, Ch 6). Because of this, LSA can be used to determine whether a pair of words is likely to participate in a regular semantic relationship, even though LSA may not contribute specific information regarding the *nature* of the relationship. However, once a relationship is expected, LSA can be used to predict whether this relationship is used in contexts that are typical uses of the words in question, or whether these uses appear to be anomalies such as rare senses or idioms. This technique was used successfully by (Cederberg and Widdows, 2003) to improve the accuracy of hyponymy extraction. It follows that it should be useful to tell the difference between regularly related words and idiomatically related words.

To test this hypothesis, we used an LSA model built from the BNC using the Infomap NLP software.[5] This was used to measure the LSA similarity between the words in each of the pairs extracted by the techniques in Section 4. In cases where a word was too infrequent to appear in the LSA model, we used ‘folding in,’ which assigns a word-vector ‘on the fly’ by adding together the vectors of any surrounding words of a target word that are in the model.

Table 3: Ordering of results from semantically similar to semantically dissimilar using LSA

Word pair	**LSA similarity**
north south	0.931
middle class	0.834
porgy bess	0.766
war aftermath	0.676
salt pepper	0.672
bits bobs	0.671
mustard cress	0.603
composer conductor	0.588
cod haddock	0.565
metal alloy	0.509
highway byway	0.480
committee subcommittee	0.479
god goddess	0.456
rock roll	0.398
continent ocean	0.300
wood charcoal	0.273
stimulus response	0.261
stocking suspender	0.177
god hero	0.115
element compound	0.044
assault battery	-0.068

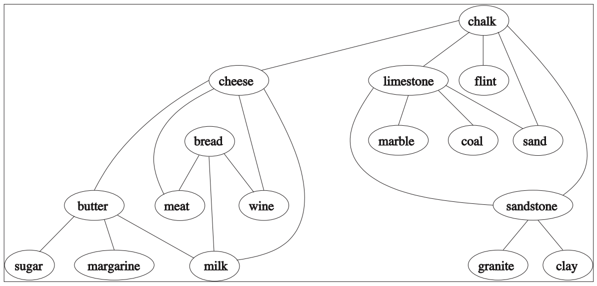


Figure 6: Nodes in the original symmetric graph in the vicinity of *chalk* and *cheese*

The results are shown in Table 3. The hypothesis is that words whose occurrence is purely idiomatic would have a low LSA similarity score, because they are otherwise not closely related. However, this hypothesis does not seem to have been confirmed, partly due to the effects of overall frequency. For example, the word *Porgy* only occurs in the phrase “Porgy and Bess,” and the word *bobs* almost always occurs in the phrase “bits and bobs.” A more effective filtering technique would need to normalize to account for these effects. However, there are some good results: for example, the low score between *assault* and *battery* reflects the fact that this usage, though compositional, is a rare meaning of the word *battery*, and the same argument can be made for *element* and *compound*. Thus LSA might be a better guide for recognizing rarity in meaning of individual words than it is for idiomaticity of phrases.

5.2 Link analysis

Another technique for determining whether a link is idiomatic or not is to check whether it connects two

[5] Freely available from `http://infomap-nlp.sourceforge.net/`

areas of meaning that are otherwise unconnected. A hallmark example of this phenomenon is the "chalk and cheese" example shown in Figure 6. [6] Note that none of the other members of the rock-types clusters is linked to any of the other foodstuffs. We may be tempted to conclude that the single link between these clusters is an idiomatic phenomenon. This technique shows promise, but has yet to be explored in detail.

6 Conclusions and Further Work

It is possible to extract asymmetric constructions from text, some of which correspond to idioms which are indecomposable (in the sense that their meaning cannot be decomposed into a combination of the meanings of their constituent words).

Many other phrases were extracted which exhibit a typical directionality that follows from underlying semantic principles. While these are sometimes not defined as 'idioms' (because they are still composable), knowledge of their asymmetric behaviour is necessary for a system to generate natural language utterances that would sound 'idiomatic' to native speakers.

While all of this information is useful for correctly interpreting and generating natural language, further work is necessary to distinguish accurately between these different categories. The first step in this process will be to manually classify the results, and evaluate the performance of different classification techniques to see if they can reliably identify different types of idiom, and also distinguish these cases from false positives that were mistakenly extracted. Once some of these techniques have been evaluated, we will be in a better position to broaden our techniques by turning to larger corpora such as the Web.

References

Timothy Baldwin, Colin Bannard, Takaaki Tanaka, and Dominic Widdows. 2003. An empirical model of multiword expression decomposability. In *Proceedings of the ACL-2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, Sapporo, Japan.

Sarah Bunin Benor and Roger Levy. 2004. The chicken or the egg? a probabilistic analysis of english binomials. `http://www.stanford.edu/~rog/papers/binomials.pdf`.

Sharon Caraballo. 1999. Automatic construction of a hypernym-labeled noun hierarchy from text. In *37th Annual Meeting of the Association for Computational Linguistics: Proceedings of the Conference*, pages 120–126.

M Carl and A Way, editors. 2003. *Recent Advances in Example-Based Machine Translation*. Kluwer.

Scott Cederberg and Dominic Widdows. 2003. Using LSA and noun coordination information to improve the precision and recall of automatic hyponymy extraction. In *Conference on Natural Language Learning (CoNNL)*, Edmonton, Canada.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge MA.

Charles J. Fillmore. 1967. The grammar of hitting and breaking. In R. Jacobs, editor, *In Readings in English: Transformational Grammar*, pages 120–133.

Marti Hearst and Hinrich Schütze. 1993. Customizing a lexicon to better suit a computational task. In *ACL SIGLEX Workshop*, Columbus, Ohio.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING*, Nantes, France.

Thomas Landauer and Susan Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of acquisition. *Psychological Review*, 104(2):211–240.

Dekang Lin. 1999. Automatic identification of non-compositional phrases. In *ACL:1999*, pages 317–324.

Ellen Riloff and Jessica Shepherd. 1997. A corpus-based approach for building semantic lexicons. In Claire Cardie and Ralph Weischedel, editors, *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 117–124. Association for Computational Linguistics, Somerset, New Jersey.

Brian Roark and Eugene Charniak. 1998. Noun-phrase co-occurence statistics for semi-automatic semantic lexicon construction. In *COLING-ACL*, pages 1110–1116.

Eleanor Rosch. 1975. Cognitive representations of semantic categories. *Journal of Experimental Psychology: General*, 104:192–233.

[6] "Chalk and cheese" is a widespread idiom in British English, used to contrast two very different objects, e.g. "They are as different as chalk and cheese." A roughly corresponding (though more predictable) phrase in American English might be "They are as different as night and day."

Frank Smadja. 1993. Retrieving collocations from text: Xtract. *Computational Linguistics*, 19(1):143–177.

Dominic Widdows and Beate Dorow. 2002. A graph model for unsupervised lexical acquisition. In *19th International Conference on Computational Linguistics*, pages 1093–1099, Taipei, Taiwan, August.

Dominic Widdows. 2004. *Geometry and Meaning*. CSLI publications, Stanford, California.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196.

Frame Semantic Enhancement of Lexical-Semantic Resources

Rebecca Green
Institute of Advanced Computer Studies
College of Information Studies
University of Maryland
College Park, MD 20742, USA
rgreen@umd.edu

Bonnie J. Dorr
Institute of Advanced Computer Studies
Department of Computer Science
University of Maryland
College Park, MD 20742, USA
bonnie@umiacs.umd.edu

Abstract

SemFrame generates FrameNet-like frames, complete with semantic roles and evoking lexical units. This output can enhance FrameNet by suggesting new frames, as well as additional lexical units that evoke existing frames. SemFrame output can also support the addition of frame semantic relationships to WordNet.

1 Introduction

The intuition that semantic analysis can make a positive contribution to language-based applications has motivated the development of a number of lexical-semantic resources. Prominent among them are WordNet,[1] PropBank,[2] and FrameNet.[3] The potential contribution of these resources is constrained by the information they contain and the level of effort involved in their development.

For example, semantic annotation tasks (Baker et al., 2004) typically assign semantic roles to the arguments of predicates. The benefit of the semantic annotation is constrained by the presence and quality of semantic roles in the lexical-semantic resource(s) used. Gildea and Jurafsky (2002) suggest that the availability of semantic annotation of this sort is useful for information extraction, word sense disambiguation, machine translation, text summarization, text mining, and speech recognition.

Other tasks rely on the identification of semantic relationships to recognize lexical chains (sets of semantically related words that enable a text to be cohesive) (Morris and Hirst, 1991). The success of this work is constrained by the set of semantic relationship types and instantiations underlying the recognition of lexical chains. As Stokes's dissertation (2004) notes, lexical cohesion has been used in discourse analysis, text segmentation, word sense disambiguation, text summarization, topic detection and tracking, and question answering.

Unfortunately, most lexical-semantic resources, including those previously mentioned, are the product of considerable ongoing human effort. Given the high development costs associated with these resources, the possibility of enhancing them on the basis of complementary resources that are produced automatically is welcome.

This paper demonstrates several of the characteristics and benefits of SemFrame (Green et al., 2004; Green and Dorr, 2004), a system that produces such a resource.

1. SemFrame generates semantic frames in a form like those of FrameNet, the ostensible gold standard for semantic frames.

2. Some SemFrame frames correspond to FrameNet frames. When SemFrame identifies additional lexical units that evoke the frame, it bolsters the use of semantic frames for identifying lexical chains.

3. Some SemFrame frames cover semantic space not yet investigated in FrameNet, which, be-

[1] http://www.cogsci.princeton.edu/˜wn
[2] http://www.cis.upenn.edu/˜ace
[3] http://framenet.icsi.berkeley.edu

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 57–66,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

cause of the labor-intensive nature of its development, is incomplete. The identification of new frames thus helps fill in gaps in FrameNet.

4. In addition to complementing FrameNet, SemFrame could be used as a more systematic source of semantic roles for PropBank or could serve as the basis for adding frame semantic relationships to WordNet.

The rest of the paper is organized as follows: Section 2 discusses lexical-semantic resources that could be enhanced by using SemFrame's output. Section 3 sets out how SemFrame works, with Subsections 3.1 and 3.2 explaining, respectively, the identification of lexical units that evoke shared semantic frames and the generation of the internal structure of those frames. Section 4 discusses how we evaluate SemFrame's output. Finally, Section 5 summarizes SemFrame's contributions and sketches future directions in its development.

2 Lexical-Semantic Resources

Lexical-semantic resources, such as FrameNet and PropBank, which involve semantic frames and/or semantic roles, are one kind of resource that SemFrame's output can enhance. SemFrame could also benefit a resource like WordNet that captures different kinds of semantic relationships. Here we discuss characteristics of these resources that make them amenable to enhancement through SemFrame.

2.1 FrameNet

FrameNet documents the semantic and syntactic behavior of words with respect to frames. A frame characterizes a conventional conceptual structure, for instance, a situation involving risk, a hitting event, a commercial transaction. Lexical units are said to evoke a frame. For example, use of the literal sense of *buy* introduces into a discourse an expectation that some object or service (the Goods) passes from one person (the Seller) to another (the Buyer) in exchange for something of (presumably equivalent) value (typically Money).

A significant contribution of the FrameNet project is the creation of frames, which involves the enumeration both of participant roles in the frame (a.k.a, frame elements, frame slots) and of lexical units that evoke the frame. As of May 2005, 657 frames have been defined in FrameNet; approximately 8600 lexical unit/frame associations have been made.

FrameNet's approach to identifying frames is "opportunistic" and driven by the corpus data being annotated. Thus the FrameNet team does not expect to have a full inventory of frames until a substantial proportion of the general-purpose vocabulary of English has been analyzed. As the development of FrameNet is labor-intensive, supplementing FrameNet's frames and evoking lexical units using data from SemFrame would be beneficial.

2.2 PropBank

Like FrameNet, PropBank (Kingsbury et al., 2002) is a project aimed at semantic annotation, in this case of the Penn English Treebank.[4] The intent of PropBank is to provide for "automatic extraction of relational data" on the basis of consistent labeling of predicate argument relationships. Typically the labels/semantic roles are verb-specific (but are often standardized across synonyms). For example, the set of semantic arguments for *promise*, *pledge*, etc. (its 'roleset') includes the promiser, the person promised to, and the promised thing or action. These correspond respectively to FrameNet's Speaker, Addressee, and Message elements within the Commitment frame.

The more general labels used in FrameNet and SemFrame give evidence of a more systematic approach to semantic argument structure, more easily promoting the discovery of relationships among frames. It can be seen from the terminology used that PropBank is more focused on the individual arguments of the semantic argument structure, while FrameNet and SemFrame are more focused on the overall gestalt of the argument structure, that is, the frame. The use of FrameNet and SemFrame to suggest more generic (that is, frame-relevant) roleset labels would help move PropBank toward greater systematicity.

[4] The semantic annotation tasks in the FrameNet and PropBank projects enable them to link semantic roles and syntactic behavior. Enhancing and stabilizing its semantic frame inventory must precede the inclusion of such linkage in SemFrame.

2.3 WordNet

WordNet is a lexical database for English nouns, verbs, adjectives, and adverbs. Fine-grained sense distinctions are recognized and organized into synonym sets ('synsets'), WordNet's basic unit of analysis; each synset has a characterizing gloss, and most are exemplified through one or more phrases or sentences.

In addition to the synonymy relationship at the heart of WordNet, other semantic relationships are referenced, including, among others, antonymy, hyponymy, troponymy, partonomy, entailment, and cause-to. On the basis of these relationships, Fellbaum (1998) noted that WordNet reflected the structure of frame semantics to a degree, but suggested that its organization by part of speech would preclude a full frame-semantic approach.

With release 2.0, WordNet added morphological and topical category relationships that cross over part-of-speech boundaries. This development relates to incorporating a full frame-semantic approach in WordNet in two ways.

First, since the lexical units that evoke a frame are not restricted to a single part of speech, the ability to create links between parts of speech is required in order to encode frame semantic relationships.

Second, topical categories (e.g., slang, meat, navy, Arthurian legend, celestial body, historical linguistics, Mafia) have a kinship with semantic frames, but are not the same. While topical category domains map between categories and lexical items—as do semantic frames—it is often not clear what internal structure might be posited for a category domain. What, for example, would the participant structure of 'meat' look like?

Should WordNet choose to adopt a full frame-semantic approach, FrameNet and SemFrame are natural starting points for identifying frame-semantic relationships between synsets. The most beneficial enhancement would involve WordNet's incorporating FrameNet and/or SemFrame frames as a separate resource, with a mapping between WordNet's synsets and the semantic frame inventory. SemFrame has the extra advantage that its lexical units are already identified as WordNet synsets.

3 Development of SemFrame

There are two main processing stages in producing SemFrame output: The first establishes verb classes, while the second generates semantic frames. The next two subsections describe these stages.

3.1 Establishing Verb Classes

SemFrame adopts a multistep approach to identifying sets of frame-semantically related verb senses. The basic steps involved in the current version[5] of SemFrame are:

1. Building a graph with WordNet verb synsets as vertices and semantic relationships as edges
2. Identifying for each vertex a maximal highly connected component (HCC) (i.e., a highly interconnected subgraph that the vertex is part of)
3. Eliminating HCC's with undesirable qualities
4. Forming preliminary verb semantic classes by supplementing HCC's with reliable semantic relationships
5. Merging verb semantic classes with a high degree of overlap

Building the Relationships Graph

WordNet 2.0 includes a vast array of semantic relationships between synsets of the same part of speech and has now been enhanced with relationships linking synsets of different parts of speech. Some of these relationships are almost guaranteed to link synsets that evoke the same frame, while others operate within the bounds of a semantic frame on some occasions, but not others. Among the relationship types in WordNet most fruitful for identifying verb synsets within the same frame semantic verb class are: synonymy (e.g., *buy, purchase*, as collocated within synsets), antonymy (e.g., *buy,*

[5] The process of establishing verb classes has been redesigned. All that has been carried over from the previous/initial version of SemFrame is the use of some of the same WordNet relationships. New in the current version are: the use of relationship types first implemented in WordNet 2.0, the predominant and exclusive use of WordNet as the source of data (the previous version used WordNet as a source secondary to the Longman Dictionary of Contemporary English), and modeling the identification of classes of related verbs as a graph, specifically through the use of highly connected components.

sell), cause-to (e.g., *transfer, change hands*), entailment (e.g., *buy, pay*), verb group (e.g., different commercial senses of *buy*, morphological derivation (e.g., *buy, buyer*),[6] and "see also" (e.g., *buy, buy out*). Instances of these relationship types for all verb synsets in WordNet 2.0 are represented as edges within the graph.

Additional edges are inserted between any two synsets/vertices related by two or more of the following: clustering of synsets based on the occurrence of word stems in their glosses and example sentences;[7] hyperonymy/hyponymy relationships; and category domain relationships. These three relationship types are too noisy to be used on their own for identifying frame semantic relationships among synsets, but when a relationship is verified by two or more of these relationships, the likelihood that the related synsets evoke the same frame is considerably higher. Table 1 summarizes the number of edges in the graph supported by each relationship type.

Relationship Type	Count
Antonymy	502
Cause-to	218
Entailment	409
Verb group	874
Morphological derivation	8,986
See also	539
Two of:	2,223
Clustering	54,298
Hyperonymy/hyponymy	12,985
Category domain	18,482
Total	13,751

Table 1: Relationship Counts in WordNet 2.0

Identifying Highly Connected Components (HCC's)

Step 1 constructs a graph interconnecting thousands of WordNet verb synsets. Identifying sets of verb synsets likely to evoke the same semantic frame requires identifying subgraphs with a high degree of interconnectivity. Empirical investigation has shown that "highly connected components" (Hartuv and Shamir, 2000)—induced subgraphs of size k in which every vertex's connectivity exceeds $\frac{k}{2}$ vertices—identify such sets of verb synsets.[8] For example, in a 5-vertex highly connected component, each vertex is related to at least 3 other vertices. Figure 1 shows a portion of the original graph in which relationship arcs constituting an HCC are given as solid lines, while those that fail the interconnectivity threshold are given as dotted lines.

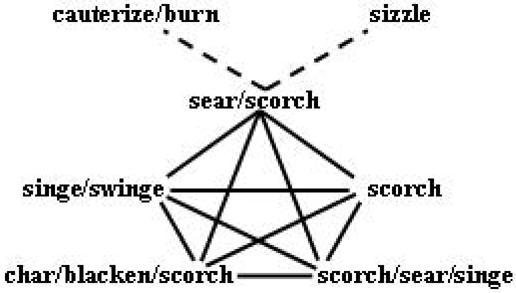


Figure 1: Relationships Subgraph with HCC

Given an undirected graph, the Hartuv-Shamir algorithm for identifying HCC's returns zero or more non-overlapping subgraphs (including zero or more singleton vertices). But it is inaccurate to assume that verb synsets evoke only a single frame, as is suggested by non-overlapping subgraphs.[9] For this reason, we have modified the Hartuv-Shamir algorithm to identify a maximal HCC, if one exists, for (i.e., that includes) each vertex of the graph. This modification reduces the effort involved in identifying any single HCC: Since the diameter of a HCC is no greater than two, only those vertices who are neighbors of the source vertex, or neighbors of those neighbors, need to be examined.

[6] SemFrame relates verb synsets with a morphological derivation relationship to a common noun synset. This includes verbs related to different members of the shared noun synset.

[7] Voorhees' (1986) hierarchical agglomerative clustering algorithm was implemented.

[8] The algorithm for computing HCC's first finds the minimum cut for a (sub)graph. If the graph meets the highly connected component criterion, the graph is returned, else the algorithm is called recursively on each of the subgraphs created by the cut. The Stoer-Wagner (1997) algorithm has been implemented for finding the minimum cut.

[9] Semantic frames can be defined at varying levels of generality; thus, a given synset may evoke a set of hierarchically related frames. Words/Synsets may also evoke multiple, unrelated frames simultaneously; *criticize*, for example, evokes both a Judging frame and a Communication frame.

Eliminating Duplicates

Because HCC's were generated for each vertex in the relationships graph, considerable duplication and overlap existed in the output. The output of step 2 was cleaned up using three filters. First, duplicate HCC's were eliminated. Second, any HCC wholly included within another HCC was deleted.[10] Third, any HCC based only on morphological derivation relationships was deleted. In SemFrame, all verb synsets morphologically derived from the same noun synset were related to each other. Thus all verb synsets derived from a common noun synset are guaranteed to generate an HCC. If only such relationships support an HCC, the likelihood that all of the interrelated verb synsets evoke the same semantic frame is much lower than if other types of relationships also provide evidence for their interrelationship.

Supplementing HCC's

The HCC's generated in step 2 that survived the filters implemented in step 3 form the basis of verb *framesets*, that is, sets of verb senses that evoke the same semantic frame. Specifically, all the synsets represented by vertices in a single HCC form a frameset.

The connectivity threshold imposed by HCC's helps maintain reasonably high precision of the resulting framesets, but is too strict for high recall. Some types of relationships known to operate within frame-semantic boundaries generally do not survive the connectivity threshold cutoff. For example, for frames of a certain level of generality, if a specific verb evokes that frame, it is also the case that its antonym evokes the frame, as antonyms operate against the backdrop of the same situational context; that is, they share participant structure.[11] However, since antonymy is (only) a *lexical* relationship between two word senses, A and B, the tight coupling of A and B is unlikely to be reflected in A's being directly related to other synsets that are related to B and vice-versa. Thus, antonyms are unlikely to be highly connected through WordNet to other words/synsets that evoke the frame and thus fail the HCC connectivity threshold. The same argument can be made for causatively related verbs. A post-processing step was required therefore to add to a frameset any verb synsets related through WordNet's antonymy or cause-to relationships to a member of the frameset. Similarly, any verb synset entailed by a member of a verb frameset was added to the frameset.

Other verb synsets fail to survive the connectivity threshold cutoff because they enter into few relationships of any kind. If a verb synset is related to only one other verb synset, the assumption is made that it evokes the same frame as that one other synset; it is then added to the corresponding frameset.

Lastly, if a synset is related to two or more members of a frameset, the likelihood that it evokes the same semantic frame is reasonably high. Such verb synsets were added to the frameset if not already present.

At the end of this phase, any framesets wholly included within another frameset were again deleted.

Merging Overlapping Verb Classes

The preceding processes produced many framesets with a significant degree of overlap. For any two framesets, if at least half of the verb synsets in both framesets were also members of the other, the two framesets were merged into a single frameset.

Summary of Stage 1 Results

The above steps generated 1434 framesets, varying in size from 2 to 25 synsets (see Table 2). Small framesets dominate the results, with over 60% of the framesets including only 2 or 3 synsets.

Representative examples of these framesets are given in Appendix A, where members of each synset appear in parentheses, followed by the synset's gloss. (Examples are ordered by frameset size.) Smaller and medium-sized framesets generally enjoy high precision, but many of the largest framesets would be better split into two or more framesets.

3.2 Generating Semantic Frames

Generating frames from verb framesets relies on the insight that the semantic arguments of a frame are largely drawn from nouns associated with verb

[10] Given the interest in generating semantic frames of varying levels of generality, this filter may itself be eliminated in the future.

[11] Identifying antonyms is especially helpful in the case of conversives, as with *buy* and *sell*; the inclusion of both in the frameset promotes discovery of all relevant frame participants, in this case, both buyer and seller.

Frameset Size	Count
2	536
3	346
4-5	309
6-8	169
9-12	54
13-25	20
Total	1434

Table 2: Count of Frameset Sizes

synsets in the frameset. In SemFrame's processing, these include nouns in the gloss of a verb synset or in the gloss of its corresponding LDOCE verb sense(s), as well as nouns (that is, noun synsets) to which a verb synset is morphologically related and those naming the category domain to which a verb synset belongs. In the latter two cases, the nouns come disambiguated within WordNet, but nouns from glosses must undergo disambiguation. The set of noun senses associated with a verb frameset is then analyzed against the WordNet noun hierarchy, using an adaptation of Agirre and Rigau's (1995) conceptual density measure. This analysis identifies a frame name and a set of frame participants, all of which correspond to nodes in the WordNet noun hierarchy.

Disambiguating Nouns from Glosses

First we consider how nouns from WordNet and LDOCE verb glosses are disambiguated.[12] This step involves looking for matches between the stems of words in the glosses of WordNet noun synsets that include the noun needing to be disambiguated, on the one hand, and the stems of words in the glosses of all WordNet verb synsets (and corresponding LDOCE verb senses) in the frameset, on the other hand.

A similarity score is computed by dividing the match count by the number of non-stop-word stems in the senses under consideration. SemFrame favors predominant senses by examining word senses in frequency order. Any sense with a non-zero similarity score that is the highest score yet seen is chosen as an appropriate word sense.

The various nodes within WordNet's noun network that correspond to a verb frameset—either through morphological derivation or category domain relationships in WordNet or through the disambiguation of nouns from the glosses of verbs in the frameset—constitute 'evidence synsets' for the participant structure of the corresponding semantic frame and form the input for the conceptual density calculation.

In preparation for use in calculating conceptual density, evidence synsets are given weights that take into account the source and basis of the disambiguation. In the current implementation, noun synsets related to the frameset through morphological derivation or shared category domain are given a weight of 4.0 (the nouns are guaranteed to be related to the verbs, and disambiguation of the nouns is built into the fact that relationships are given between synsets); disambiguated noun synsets coming from WordNet verb synsets receive a weight of 2.0 (since the original framesets contain WordNet synsets, and the disambiguation strategy is fairly conservative); non-disambiguated nouns coming from LDOCE verbs related to the frameset have a weight of 0.5 (LDOCE verbs are a step removed from the original framesets, and the nouns have not been disambiguated); all other nouns receive a weight of 1.0. The weight for non-disambiguated nouns is ultimately distributed across the noun's senses, with higher proportions of the weight being assigned to more frequent senses.

Computing Conceptual Density

The overall idea behind transforming the list of evidence synsets into a list of participants involves using the relationship structure of WordNet to identify an appropriately small set of concepts (i.e., synsets) within WordNet that account for (i.e., are superordinate to) as many of the evidence synsets as possible; such synsets will be referred to as 'covering synsets'.

This task relies on the hypothesis that a frame's evidence synsets will not be randomly distributed across WordNet, but will be clustered in various subtrees within the hierarchy. Intuitively, when evidence synsets cluster together, the subtrees in which they occur will be more dense than those subtrees where few or no evidence synsets occur. It is hypothesized that the WordNet subtrees with the high-

[12] Identification of LDOCE verb senses that correspond to WordNet verb synsets is carried out using a similar strategy.

est density are the most likely to correspond to frame slots. Thus, the task is to identify such clusters/subtrees and then to designate the nodes at the roots of the subtrees as covering synsets (subject to certain constraints).

The conceptual density measure we have used has been inspired by the measure of the same name in Agirre and Rigau (1995). The conceptual density, CD(n), of a node *n* is computed as follows:

$$CD(n) = \frac{\sum_{i \epsilon descendants_n}(wgt_i * treesize_i)}{treesize_n}$$

Both frame names and frame slots are identified on the basis of this conceptual density measure, with the frame name being taken from the node with the highest conceptual density from a specified group of subnetworks within the WordNet noun network (including abstractions, actions, events, phenomena, psychological features, and states). Frame slots are subject to a density threshold (based on mean density and variance), an evidence-synset-support threshold, and a constraint on the number of possible slots to be taken from specific subnetworks within WordNet. Further details on the computation and interpretation of conceptual density are given in (Green and Dorr, 2004).

Frame names and frame structures for the framesets in Appendix A are given in Appendix B. The full set of SemFrame's frames (including ca. 30,000 lexical unit/frame associations) is publicly available at: http://www.cs.umd.edu/˜rgreen/semframe2.tar.gz.

The correspondence between frameset sizes and the number of slots generated for the frame is worth noting, since we have independent evidence about the number of slots that should be generated. Frames in FrameNet generally have from 1 to 5 slots (occasionally more). Over 70% of SemFrame's frames contain from 1 to 5 frame slots. Of course, generating an appropriate number of frame slots is not the same as generating the right frame slots, a determination that requires empirical investigation.

4 Evaluation

Three student judges evaluated SemFrame's results, with 200 frames each assessed by two judges, and 1234 frames each assessed by one judge.

In evaluating a frame, judges began by examining the set of verb synsets deemed to evoke a common frame and identified from among them the largest subset of synsets they considered to evoke the same frame. This frame—designated the 'target frame'—was simply a mental construct in the judge's mind. For only 9% of the frame judgments were the judges unable to identify a target frame.

If a target frame was discerned, judges were then asked to evaluate whether the WordNet verb synsets and LDOCE verb senses listed by SemFrame could be used to communicate about the frame the judge had in mind. This evaluation step applied to 6147 WordNet verb synsets and 7148 LDOCE verb senses; in the judges' views, 78% of the synsets and 68% of the verb senses evoke the target frame.

Judges were asked how well the frame names generated by SemFrame capture the overall target frame. Some 53% of the names were perceived to be satisfactory (good or excellent), with another 25% of the names in the right hierarchy. Only 11% of the names were deemed to be only mediocre and 9% to be unrelated.

Judges were also asked how well the frame element names generated by SemFrame named a participant or attribute of the target frame. Here 46% of the names were found satisfactory, with another 18% of the names consistent with a target frame participant, but either too general or too narrow. Another 5% of the names were regarded as mediocre and 30% as unrelated.

Lastly, judges were asked to look for correspondences between target frames and FrameNet frames. While only 17% of the target frames were considered equivalent to a FrameNet frame, many were judged to be hierarchically related; 51% of the FrameNet frames were judged more general than the corresponding SemFrame frame, while 8% were judged more specific. This reflects the need to combine some number of SemFrame frames. For 23% of the SemFrame frames, even the best FrameNet match was considered only mediocre. These may represent viable frames not yet recognized by FrameNet. Judges also found 3668 verbs in SemFrame that could be appropriately listed for a corresponding frame in FrameNet, but were not.

These results reveal SemFrame's strengths in inducing frames by enumerating sets of verbs that evoke a shared frame and in naming such frames. SemFrame's ability to postulate names for the elements of a frame is less robust, although results in this area are still noteworthy.

5 Conclusion and Future Work

SemFrame's output can be used to enhance lexical-semantic resources in various ways. For example, WordNet has recently incorporated new relationship types, some of which touch on frame semantic relationships. But frame semantic relationships are as yet only implicit in WordNet; not all morphological derivation relationships, for example, operate within a frame. Should WordNet choose to reflect frame semantic relationships, SemFrame would provide a useful point of departure, since the verb framesets, frame names, and frame slots are all already expressed as WordNet synsets.

SemFrame can also add to FrameNet. The extensive human effort that has gone into FrameNet is overwhelmingly evident in the quality of its frame structures (and attendant annotations). SemFrame is unlikely ever to compete with FrameNet on this score. However, SemFrame has identified frames not recognized in FrameNet, e.g., SemFrame's SOILING frame. SemFrame has likewise identified lexical units appropriate to FrameNet frames that have not yet been incorporated into FrameNet, e.g., *stick to*, *stick with*, and *abide by* in the COMPLIANCE / CONFORMITY frame. These contributions would add as well to the semantic representations in PropBank. Since identifying frames and their evoking lexical units from scratch requires more effort than assessing the general quality of proposed frames and lexical units—indeed, since there is currently no other systematic way in which to identify either a universal set of semantic frames or the set of lexical items that evoke a frame—SemFrame's ability to propose new frames and new evoking lexical units constitutes a major contribution to the development of lexical-semantic resources.

SemFrame's current results might themselves be enhanced by considering data from other parts of speech. For instance, at present SemFrame bases all its frames on verb framesets, but some FrameNet frames list only adjectives as evoking lexical units. At the same time, potentially more can be done in associating verb synsets with frames: Only one-third of WordNet's verb synsets are now included in SemFrame's output. Some of those not now included evoke none of SemFrame's current frames, but some do and have not yet been recognized. Ways of establishing hierarchical and compositional relationships among frames should also be investigated.

The above suggestions for enhancing SemFrame notwithstanding, major progress in improving SemFrame awaits incorporation of corpus data. Relying on data from lexical resources has contributed to SemFrame's precision, but the data sparseness bottleneck that SemFrame faces is nonetheless real. On the basis of the lexical resource data used, verb synsets are related on average to only 5 nouns, many of which closely reflect the participant structure of the corresponding frame. However, it is not uncommon for specific elements of the participant structure to go unrepresented, and any nouns in the dataset that are not particularly reflective of the participant structure carry far too much weight amidst such a paucity of data.

In contrast, the number of nouns that co-occur with a verb in a corpus may be orders of magnitude greater.[13] But the nouns in a corpus are less likely to reflect closely the participant structure of the corresponding frame; many more nouns are thus likely to be needed. Furthermore, word sense disambiguation will be required to assign to a frame only those nouns corresponding to an appropriate sense of the verb.[14] We are optimistic, however, that the presence of additional corpus data will help fill in frame element gaps arising from the sparseness of lexical resource data and can also be used to help reduce the impact of nouns from lexical resource data that are not representative of a frame's participant structure.

Coupled with subject-specific resources, the analysis of corpus data may then lead to the development

[13] We are investigating two levels of noun-verb co-occurrence. The first counts co-occurrences of all nouns and verbs appearing within the same paragraph of newswire texts. The second counts only those nouns related to verbs as their subjects, direct objects, indirect objects, or as objects of prepositional phrases that modify the verb.

[14] We make the simplifying assumption that if a noun occurs with some reasonable percentage of the verbs within a frameset, the desired verb sense is in play.

of subject-specific frame inventories. Such inventories can in turn inform such knowledge-intensive applications as information retrieval, information extraction, and question answering.

Acknowledgements

This work has been supported in part by NSF ITR Grant IIS-0326553.

References

Eneko Agirre and German Rigau. A proposal for word sense disambiguation using conceptual distance. *1st International Conference on Recent Advances in NLP*.

Collin Baker, Jan Hajic, Martha Palmer, and Manfred Pinkal. 2004. Beyond syntax: Predicates, arguments, valency frames and linguistic annotations. Tutorial at *42nd Annual Meeting of the Association of Computational Linguistics*.

Christiane Fellbaum (Ed.) 1998. Introduction. In C. Fellbaum (Ed.), *WordNet: An Electronic Lexical Database.* MIT Press.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3): 245–288.

Rebecca Green, Bonnie J. Dorr, and Philip Resnik. 2004. Inducing frame semantic verb classes from WordNet and LDOCE. *42nd Annual Meeting of the Association of Computational Linguistics*.

Rebecca Green and Bonnie J. Dorr. 2004. Inducing a semantic frame lexicon from WordNet data. *Workshop on Text Meaning and Interpretation, 42nd Annual Meeting of the Association of Computational Linguistics*.

Erez Hartuv and Ron Shamir. 2000. A clustering algorithm based on graph connectivity. *Information Processing Letters*, 76:175–181.

Paul Kingsbury, Martha Palmer, and Mitch Marcus. 2002. Adding Semantic Annotation to the Penn Treebank. *Proceedings of the Human Language Technology Conference*.

Jane Morris and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 18(1):21–48.

Paul Procter (Ed.) 1978. *Longman Dictionary of Contemporary English.* Longman Group Ltd.

Mechthild Stoer and Frank Wagner. 1997. A simple mincut algorithm. *Journal of the ACM*, 44(4):585–591.

Nicola Stokes. 2004. Applications of lexical cohesion: Analysis in the topic detection and tracking domain. Ph.D. dissertation, National University of Ireland, Dublin.

Ellen Voorhees. 1986. Implementing agglomerative hierarchic clustering algorithms for use in document retrieval. *Information Processing & Management*, 22(6):465–476.

A Sample Framesets

(a)
(stick_to stick_with follow) keep to
(comply follow abide_by) act in accordance with someone's rules, commands, or wishes
(b)
(sneer) smile contemptuously
(sneer) express through a scornful smile
(contemn despise scorn disdain) look down on with disdain
(c)
(muck) remove muck, clear away muck, as in a mine
(slime) cover or stain with slime
(clean make_clean) make clean by removing dirt, filth, or unwanted substances from
(dirty soil begrime grime colly bemire) make soiled, filthy, or dirty
(mire muck mud muck_up) soil with mud, muck, or mire
(d)
(federate federalize federalise) unite on a federal basis or band together as a league
(ally) become an ally or associate, as by a treaty or marriage
(confederate) form a confederation with; of nations
(divide split split_up separate dissever carve_up) separate into parts or portions
(unite unify) act in concert or unite in a common purpose or belief
(band_together confederate) form a group or unite
(e)
(fade melt) become less clearly visible or distinguishable; disappear gradually or seemingly
(get_down begin get start_out start set_about set_out commence) take the first step or steps in carrying out an action
(begin lead_off start commence) set in motion, cause to start
(end terminate) bring to an end or halt
(appear come_along) come into being or existence, or appear on the scene
(vanish disappear) cease to exist
(vanish disappear go_away) become invisible or unnoticeable
(begin start) have a beginning, in a temporal, spatial, or evaluative sense
(end stop finish terminate cease) have an end, in a temporal, spatial, or quantitative sense; either spatial or metaphorical

B Sample Frames

(a)
FRAME CONFORMITY (acting according to certain accepted standards):
- ATTRIBUTE (complaisance (a disposition or tendency to yield to the will of others)) [ ]
- COMMUNICATION (law (legal document setting forth rules governing a particular kind of activity)) [ ]
- PSYCH FEATURE (e.g., law (a rule or body of rules of conduct essential to or binding upon human society)) [ ]
- PERSON1/AGENT [ ]
- PERSON2/RECIPIENT OR PATIENT [ ]
- COMMUNICATION (advice (a proposal for an appropriate course of action)) [ ]
- ACT (e.g., accordance (the act of granting rights)) [ ]

(b)
FRAME CONTEMPT (open disrespect for a person or thing):
- COMMUNICATION (scorn (open disrespect for a person or thing)) [ ]
- PERSON1/AGENT [ ]
- PERSON2/RECIPIENT OR PATIENT [ ]

(c)
FRAME SOILING (the act of soiling something):
- ACTION (e.g., soiling (the act of soiling something)) [ ]
- STATE (e.g., soil (the state of being covered with unclean things)) [ ]
- CLEANER (the operator of dry cleaning establishment) [ ]
- CLEANER (someone whose occupation is cleaning) [ ]

(d)
FRAME CONFEDERATION (the act of forming an alliance or confederation):
- ACTION (e.g., division (the act or process of dividing)) [ ]
- SPLITTER (a taxonomist who classifies organisms into many groups on the basis of relatively minor characteristics) [ ]
- STATE (e.g., marriage (the state of being a married couple voluntarily joined for life (or until divorce))) [ ]

(e)
FRAME BEGINNING (the act of starting something):
- ACTION (e.g., beginning (the act of starting something)) [ ]
- COMMUNICATION (conclusion (the last section of a communication)) [ ]

Bootstrapping Deep Lexical Resources: Resources for Courses

Timothy Baldwin
Department of Computer Science and Software Engineering
University of Melbourne, Victoria 3010 Australia
tim@csse.unimelb.edu.au

Abstract

We propose a range of deep lexical acquisition methods which make use of morphological, syntactic and ontological language resources to model word similarity and bootstrap from a seed lexicon. The different methods are deployed in learning lexical items for a precision grammar, and shown to each have strengths and weaknesses over different word classes. A particular focus of this paper is the relative accessibility of different language resource types, and predicted "bang for the buck" associated with each in deep lexical acquisition applications.

1 Introduction

Over recent years, computational linguistics has benefitted considerably from advances in statistical modelling and machine learning, culminating in methods capable of deeper, more accurate automatic analysis, over a wider range of languages. Implicit in much of this work, however, has been the existence of **deep language resources** (DLR hereafter) of ever-increasing linguistic complexity, including lexical semantic resources (e.g. WordNet and FrameNet), precision grammars (e.g. the English Resource Grammar and the various ParGram grammars) and richly-annotated treebanks (e.g. PropBank and CCGbank).

Due to their linguistic complexity, DLRs are invariably constructed by hand and thus restricted in size and coverage. Our aim in this paper is to develop general-purpose automatic methods which can be used to automatically expand the coverage of an existing DLR, through the process of **deep lexical acquisition** (DLA hereafter).

The development of DLRs can be broken down into two basic tasks: (1) design of a data representation to systematically capture the generalisations and idiosyncracies of the dataset of interest (**system design**); and (2) classification of data items according to the predefined data representation (**data classification**). In the case of a deep grammar, for example, system design encompasses the construction of the system of lexical types, templates, and/or phrase structure rules, and data classification corresponds to the determination of the lexical type(s) each individual lexeme conforms to. DLA pertains to the second of these tasks, in automatically mapping a given lexeme onto a pre-existing system of lexical types associated with a DLR.

We propose to carry out DLA through a bootstrap process, that is by employing some notion of word similarity, and learning the lexical types for a novel lexeme through analogy with maximally similar word(s) for which we know the lexical types. In this, we are interested in exploring the impact of different secondary language resources (LRs) on DLA, and estimating how successfully we can expect to learn new lexical items from a range of LR types. That is, we estimate the expected DLA "bang for the buck" from a range of secondary LR types of varying size and complexity. As part of this, we look at the relative impact of different LRs on DLA for different open word classes, namely nouns, verbs, adjectives and adverbs.

We demonstrate the proposed DLA methods relative to the English Resource Grammar (see Section 2.1), and in doing so assume the lexical types of the target DLR to be syntactico-semantic in nature. For example, we may predict that the word *dog* has a usage as an intransitive countable noun (`n_intr_le`,[1] cf. *The dog barked*), and also as a transitive verb (`v_np_trans_le`, cf. *It dogged my every step*).

A secondary interest of this paper is the consideration of how well we could expect to perform DLA for languages of differing density, from "low-

[1] All example lexical types given in this paper are taken directly from the English Resource Grammar – see Section 2.1.

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 67–76,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

density" languages (such as Walpiri or Uighur) for which we have limited LRs, to "high-density" languages (such as English or Japanese) for which we have a wide variety of LRs. To this end, while we exclusively target English in this paper, we experiment with a range of LRs of varying complexity and type, including morphological, syntactic and ontological LRs. Note that we attempt to maintain consistency across the feature sets associated with each, to make evaluation as equitable as possible.

The remainder of this paper is structured as follows. Section 2 outlines the process of DLA and reviews relevant resources and literature. Sections 3, 4 and 5 propose a range of DLA methods based on morphology, syntax and ontological semantics, respectively. Section 6 evaluates the proposed methods relative to the English Resource Grammar.

2 Task Outline

This research aims to develop methods for DLA which can be run automatically given: (a) a pre-existing DLR which we wish to expand the coverage of, and (b) a set of secondary LRs/preprocessors for that language. The basic requirements to achieve this are the discrete inventory of lexical types in the DLR, and a pre-classification of each secondary LR (e.g. as a corpus or wordnet, to determine what set of features to employ). Beyond this, we avoid making any assumptions about the language family or DLR type.

The DLA strategy we propose in this research is to use secondary LR(s) to arrive at a feature signature for each lexeme, and map this onto the system of choice indirectly via supervised learning, i.e. observation of the correlation between the feature signature and classification of bootstrap data. This methodology can be applied to unannotated corpus data, for example, making it possible to tune a lexicon to a particular domain or register as exemplified in a particular repository of text. As it does not make any assumptions about the nature of the system of lexical types, we can apply it fully automatically to any DLR and feed the output directly into the lexicon without manual intervention or worry of misalignment. This is a distinct advantage when the inventory of lexical types is continually undergoing refinement, as is the case with the English Resource Grammar (see below).

A key point of interest in this paper is the investigation of the relative "bang for the buck" when different types of LR are used for DLA. Crucially, we investigate only LRs which we believe to be plausibly available for languages of varying density, and aim to minimise assumptions as to the pre-existence of particular preprocessing tools. The basic types of resources and tools we experiment with in this paper are detailed in Table 1.

Past research on DLA falls into two basic categories: expert system-style DLA customised to learning particular linguistic properties, and DLA via resource translation. In the first instance, a specialised methodology is proposed to (automatically) learn a particular linguistic property such as verb subcategorisation (e.g. Korhonen (2002)) or noun countability (e.g. Baldwin and Bond (2003a)), and little consideration is given to the applicability of that method to more general linguistic properties. In the second instance, we take one DLR and map it onto another to arrive at the lexical information in the desired format. This can take the form of a one-step process, in mining lexical items directly from a DLR (e.g. a machine-readable dictionary (Sanfilippo and Poznański, 1992)), or two-step process in reusing an existing system to learn lexical properties in one format and then mapping this onto the DLR of choice (e.g. Carroll and Fang (2004) for verb subcategorisation learning).

There have also been instances of more general methods for DLA, aligned more closely with this research. Fouvry (2003) proposed a method of token-based DLA for unification-based precision grammars, whereby partially-specified lexical features generated via the constraints of syntactically-interacting words in a given sentence context, are combined to form a consolidated lexical entry for that word. That is, rather than relying on indirect feature signatures to perform lexical acquisition, the DLR itself drives the incremental learning process. Also somewhat related to this research is the general-purpose verb feature set proposed by Joanis and Stevenson (2003), which is shown to be applicable in a range of DLA tasks relating to English verbs.

2.1 English Resource Grammar

All experiments in this paper are targeted at the **English Resource Grammar** (ERG; Flickinger (2002), Copestake and Flickinger (2000)). The ERG is an implemented open-source broad-coverage precision Head-driven Phrase Structure Grammar

Secondary LR type	*Description*	*Preprocessor(s)*
Word list***	List of words with basic POS	—
Morphological lexicon*	Derivational and inflectional word relations	—
Compiled corpus***	Unannotated text corpus	POS tagger** Chunk parser* Dependency parser*
WordNet-style ontology*	Lexical semantic word linkages	—

Table 1: Secondary LR and tool types targeted in this research (*** = high expectation of availability for a given language; ** = medium expectation of availability; * = low expectation of availability)

(HPSG) developed for both parsing and generation. It contains roughly 10,500 lexical items, which, when combined with 59 lexical rules, compile out to around 20,500 distinct word forms.[2] Each lexical item consists of a unique identifier, a lexical type (one of roughly 600 leaf types organized into a type hierarchy with a total of around 4,000 types), an orthography, and a semantic relation. The grammar also contains 77 phrase structure rules which serve to combine words and phrases into larger constituents. Of the 10,500 lexical items, roughly 3,000 are multiword expressions.

To get a basic sense of the syntactico-semantic granularity of the ERG, the noun hierarchy, for example, is essentially a cross-classification of countability/determiner co-occurrence, noun valence and preposition selection properties. For example, lexical entries of `n_mass_count_ppof_le` type can be either countable or uncountable, and optionally select for a PP headed by *of* (example lexical items are *choice* and *administration*).

As our target lexical type inventory for DLA, we identified all open-class lexical types with at least 10 lexical entries, under the assumption that: (a) the ERG has near-complete coverage of closed-class lexical entries, and (b) the bulk of new lexical entries will correspond to higher-frequency lexical types. This resulted in the following breakdown:[3]

Word class	*Lexical types*	*Lexical items*
Noun	28	3,032
Verb	39	1,334
Adjective	17	1,448
Adverb	26	721
Total	110	5,675

Note that it is relatively common for a lexeme to occur with more than one lexical type in the ERG: 22.6% of lexemes have more than one lexical type, and the average number of lexical types per lexeme is 1.12.

In evaluation, we assume we have prior knowledge of the basic word classes each lexeme belongs to (i.e. noun, verb, adjective and/or adverb), information which could be derived trivially from preexisting shallow lexicons and/or the output of a tagger.

Recent development of the ERG has been tightly coupled with treebank annotation, and all major versions of the grammar are deployed over a common set of treebank data to help empirically trace the evolution of the grammar and retrain parse selection models (Oepen et al., 2002). We treat this as a held-out dataset for use in analysis of the *token* frequency of each lexical item, to complement analysis of *type*-level learning performance (see Section 6).

2.2 Classifier design

The proposed procedure for DLA is to generate a feature signature for each word contained in a given secondary LR, take the subset of lexemes contained in the original DLR as training data, and learn lexical items for the remainder of the lexemes through supervised learning. In order to maximise comparability between the results for the different DLRs, we employ a common classifier design wherever possible (in all cases other than ontology-based DLA),

[2] All statistics and analysis relating to the ERG in this paper are based on the version of 11 June, 2004.

[3] Note that all results are over simplex lexemes only, and that we choose to ignore multiword expressions in this research.

using TiMBL 5.0 (Daelemans et al., 2003); we used the IB1 k-NN learner implementation within TiMBL, with k = 9 throughout.[4] We additionally employ the feature selection method of Baldwin and Bond (2003b), which generates a combined ranking of all features in descending order of "informativeness" and skims off the top-N features for use in classification; N was set to 100 in all experiments.

As observed above, a significant number of lexemes in the ERG occur in multiple lexical items. If we were to take all lexical type combinations observed for a single lexeme, the total number of lexical "super"-types would be 451, of which 284 are singleton classes. Based on the sparseness of this data and also the findings of Baldwin and Bond (2003b) over a countability learning task, we choose to carry out DLA via a suite of 110 binary classifiers, one for each lexical type.

We deliberately avoid carrying out extensive feature engineering over a given secondary LR, choosing instead to take a varied but simplistic set of features which is parallelled as much as possible between LRs (see Sections 3–5 for details). We additionally tightly constrain the feature space to a maximum of 3,900 features, and a maximum of 50 feature instances for each feature type; in each case, the 50 feature instances are selected by taking the features with highest saturation (i.e. the highest ratio of non-zero values) across the full lexicon. This is in an attempt to make evaluation across the different secondary LRs as equitable as possible, and get a sense of the intrinsic potential of each secondary LR in DLA. Each feature instance is further translated into two feature values: the raw count of the feature instance for the target word in question, and the relative occurrence of the feature instance over all target word token instances.

One potential shortcoming of our classifier architecture is that a given word can be negatively classified by all unit binary classifiers and thus not assigned any lexical items. In this case, we fall back on the majority-class lexical type for each word class the word has been pre-identified as belonging to.

3 Morphology-based Deep Lexical Acquisition

We first perform DLA based on the following morphological LRs: (1) word lists, and (2) morphological lexicons with a description of derivational word correspondences. Note that in evaluation, we presuppose that we have access to word lemmas although in the first instance, it would be equally possible to run the method over non-lemmatised data.[5]

3.1 Character n-grams

In line with our desire to produce DLA methods which can be deployed over both low- and high-density languages, our first feature representation takes a simple word list and converts each lexeme into a character n-gram representation.[6] In the case of English, we generated all 1- to 6-grams for each lexeme, and applied a series of filters to: (1) filter out all n-grams which occurred less than 3 times in the lexicon data; and (2) filter out all n-grams which occur with the same frequency as larger n-grams they are proper substrings of. We then select the 3,900 character n-grams with highest saturation across the lexicon data (see Section 2.2).

The character n-gram-based classifier is the simplest of all classifiers employed in this research, and can be deployed on any language for which we have a word list (ideally lemmatised).

3.2 Derviational morphology

The second morphology-based DLA method makes use of derivational morphology and analysis of the process of word formation. As an example of how derivational information could assist DLA, knowing that the noun *achievement* is deverbal and incorporates the *-ment* suffix is a strong predictor of it being optionally uncountable and optionally selecting for a PP argument (i.e. being of lexical type `n_mass_count_ppof_le`).

We generate derivational morphological features for a given lexeme by determining its word cluster in CATVAR[7] (Habash and Dorr, 2003) and then for each sister lexeme (i.e. lexeme occurring in the

[4] We also experimented with bsvm and SVMLight, and a maxent toolkit, but found TiMBL to be superior overall, we hypothesise due to the tight integration of continuous features in TiMBL.

[5] Although this would inevitably lose lexical generalisations among the different word forms of a given lemma.

[6] We also experimented with syllabification, but found the character n-grams to produce superior results.

[7] In the case that the a given lemma is not in CATVAR, we attempt to dehyphenate and then deprefix the word to find a match, failing which we look for the lexeme of smallest edit distance.

same cluster as the original lexeme with the same word stem), determine if there is a series of edit operations over suffixes and prefixes which maps the lexemes onto one another. For each sister lexeme where such a correspondence is found to exist, we output the nature of the character transformation and the word classes of the lexemes involved. E.g., the sister lexemes for *achievement*$_{\mathsf{N}}$ in CATVAR are *achieve*$_{\mathsf{V}}$, *achiever*$_{\mathsf{N}}$, *achievable*$_{\mathsf{Adj}}$ and *achievability*$_{\mathsf{N}}$; the mapping between *achievement*$_{\mathsf{N}}$ and *achiever*$_{\mathsf{N}}$, e.g., would be analysed as:

$$\texttt{N} \; -\texttt{ment\$} \rightarrow \texttt{N} \; +\texttt{r\$}$$

Each such transformation is treated as a single feature.

We exhaustively generate all such transformations for each lexeme, and filter the feature space as for character n-grams above.

Clearly, LRs which document derivational morphology are typically only available for high-density languages. Also, it is worth bearing in mind that derivational morphology exists in only a limited form for certain language families, e.g. agglutinative languages.

4 Syntax-based Deep Lexical Acquisition

Syntax-based DLA takes a raw text corpus and preprocesses it with either a tagger, chunker or dependency parser. It then extracts a set of 39 feature types based on analysis of the token occurrences of a given lexeme, and filters over each feature type to produce a maximum of 50 feature instances of highest saturation (e.g. if the feature type is the word immediately proceeding the target word, the feature instances are the 50 words which proceed the most words in our lexicon). The feature signature associated with a word for a given preprocessor type will thus have a maximum of 3,900 items ($39 \times 50 \times 2$).[8]

4.1 Tagging

The first and most basic form of syntactic preprocessing is part-of-speech (POS) tagging. For our purposes, we use a Penn treebank-style tagger custom-built using fnTBL 1.0 (Ngai and Florian, 2001), and further lemmatise the output of the tagger using morph (Minnen et al., 2000).

The feature types used with the tagger are detailed in Table 2, where the position indices are relative to the target word (e.g. the word at position -2 is two words to the left of the target word, and the POS tag at position 0 is the POS of the target word). All features are relative to the POS tags and words in the immediate context of each token occurrence of the target word. "Bi-words" are word bigrams (e.g. biword $(1, 3)$ is the bigram made up of the words one and three positions to the right of the target word); "bi-tags" are, similarly, POS tag bigrams.

4.2 Chunking

The second form of syntactic preprocessing, which builds directly on the output of the POS tagger, is CoNLL 2000-style full text chunking (Tjong Kim Sang and Buchholz, 2000). The particular chunker we use was custom-built using fnTBL 1.0 once again, and operates over the lemmatised output of the POS tagger.

The feature set for the chunker output includes a subset of the POS tagger features, but also makes use of the local syntactic structure in the chunker input in incorporating both intra-chunk features (such as modifiers of the target word if it is the head of a chunk, or the head if it is a modifier) and inter-chunk features (such as surrounding chunk types when the target word is chunk head). See Table 2 for full details.

Note that while chunk parsers are theoretically easier to develop than full phrase-structure or treebank parsers, only high-density languages such as English and Japanese have publicly available chunk parsers.

4.3 Dependency parsing

The third and final form of syntactic preprocessing is dependency parsing, which represents the pinnacle of both robust syntactic sophistication and inaccessibility for any other than the highest-density languages.

The particular dependency parser we use is RASP[9] (Briscoe and Carroll, 2002), which outputs head–modifier dependency tuples and further classifies each tuple according to a total of 14 relations; RASP also outputs the POS tag of each word token. As our features, we use both local word and POS features, for comparability with the POS tagger

[8] Note that we will have less than 50 feature instances for some feature types, e.g. the POS tag of the target word, given that the combined size of the Penn POS tagset is 36 elements (not including punctuation).

[9] RASP is, strictly speaking, a full syntactic parser, but we use it in dependency parser mode

Feature type	*Positions/description*	*Total*
TAGGER		***39***
POS tag	$(-4,-3,-2,-1,0,1,2,3,4)$	9
Word	$(-4,-3,-2,-1,1,2,3,4)$	8
POS bi-tag	$((-4,-1),(-4,0),(-3,-2),(-3,-1),(-3,0),(-2,-1),(-2,0),$ $(-1,0),(0,1),(0,2),(0,3),(0,4),(1,2),(1,3),(1,4),(2,3))$	16
Bi-word	$((-3,-2),(-3,-1),(-2,-1),(1,2),(1,3),(2,3))$	6
CHUNKER		***39***
Modifier$_{\text{head}}$	Chunk heads when target word is modifier	1
Modifier$_{\text{chunk}}$	Chunk types when target word is modifier	1
Modifiee$_{\text{word}}$	Modifiers when target word is chunk head	1
Modifiee$_{\text{POS}}$	POS tag of modifiers when target word is chunk head	1
Modifiee$_{\text{word+POS}}$	Word + POS tag of modifiers when target word is chunk head	1
POS tag	$(-3,-2,-1,0,1,2,3)$	7
Word	$(-3,-2,-1,1,2,3)$	6
Chunk	$(-4,-3,-2,-1,0,1,2,3,4)$	9
Chunk head	$(-3,-2,-1,1,2,3)$	6
Bi-chunk	$((-2,-1),(-2,0),(-1,0),(0,1),(0,2),(1,2))$	6
DEPENDENCY PARSER		***39***
POS tag	$(-2,-1,0,1,2)$	5
Word	$(-2,-1,1,2)$	4
Conj$_{\text{word}}$	Words the target word coordinates with	1
Conj$_{\text{POS}}$	POS of words the target word coordinates with	1
Head	Head word when target word modifier in dependency relation (× 14)	14
Modifier	Modifier when target word head of dependency relation (× 14)	14

Table 2: Feature types used in syntax-based DLA for the different preprocessors

and chunker, and also dependency-derived features, namely the modifier of all dependency tuples the target word occurs as head of, and conversely, the head of all dependency tuples the target word occurs as modifier in, along with the dependency relation in each case. See Table 2 for full details.

4.4 Corpora

We ran the three syntactic preprocessors over a total of three corpora, of varying size: the Brown corpus (∼460K tokens) and Wall Street Journal corpus (∼1.2M tokens), both derived from the Penn Treebank (Marcus et al., 1993), and the written component of the British National Corpus (∼98M tokens: Burnard (2000)). This selection is intended to model the effects of variation in corpus size, to investigate how well we could expect syntax-based DLA methods to perform over both smaller and larger corpora.

Note that the only corpus annotation we make use of is sentence tokenisation, and that all preprocessors are run automatically over the raw corpus data. This is in an attempt to make the methods maximally applicable to lower-density languages where annotated corpora tend not to exist but there is at least the possibility of accessing raw text collections.

5 Ontology-based Deep Lexical Acquisition

The final DLA method we explore is based on the hypothesis that there is a strong correlation between the semantic and syntactic similarity of words, a claim which is best exemplified in the work of Levin (1993) on diathesis alternations. In our case, we take word similarity as given and learn the syntactic behaviour of novel words relative to semantically-similar words for which we know the lexical types. We use WordNet 2.0 (Fellbaum, 1998) to determine word similarity, and for each sense of the target word in WordNet: (1) construct the set of "semantic neighbours" of that word sense, comprised of all synonyms, direct hyponyms and direct hypernyms; and (2) take a majority vote across the lexical types of the semantic neighbours which occur in the training data. Note that this diverges from the learning paradigm adopted for the morphology- and syntax-based DLA methods in that we use a simple voting strategy rather than relying on an external learner to carry out the classification. The full set of lexical entries for the target word is generated by taking the union of the majority votes across all senses of the word, such that a polysemous lexeme can potentially give rise to multiple lexical entries. This learning procedure is based on the method used by van der Beek and Baldwin (2004) to learn Dutch countability.

As for the suite of binary classifiers, we fall back on the majority class lexical type as the default in the instance that a given lexeme is not contained in WordNet 2.0 or no classification emerges from the set of semantic neighbours. It is important to realise that WordNet-style ontologies exist only for the highest-density languages, and that this method will thus have very limited language applicability.

6 Evaluation

We evaluate the component methods over the 5,675 open-class lexical items of the ERG described in Section 2.1 using 10-fold stratified cross-validation. In each case, we calculate the **type precision** (the proportion of correct hypothesised lexical entries) and **type recall** (the proportion of gold-standard lexical entries for which we get a correct hit), which we roll together into the **type F-score** (the harmonic mean of the two) relative to the gold-standard ERG lexicon. We also measure the **token accuracy** for the lexicon derived from each method, relative to the Redwoods treebank of Verbmobil data associated with the ERG (see Section 2.1).[10] The token accuracy represents a weighted version of type precision, relative to the distribution of each lexical item in a representative text sample, and provides a crude approximation of the impact of each DLA method on parser coverage. That is, it gives more credit for a method having correctly hypothesised a commonly-occurring lexical item than a low-frequency lexical item, and no credit for having correctly identified a lexical item not occurring in the corpus.

The overall results are presented in Figure 1, which are then broken down into the four open word classes in Figures 2–5. The baseline method (*Base*) in each case is a simple majority-class classifier, which generates a unique lexical item for each lexeme pre-identified as belonging to a given word class of the following type:

Word class	Majority-class lexical type
Noun	`n_intr_le`
Verb	`v_np_trans_le`
Adjective	`adj_intrans_le`
Adverb	`adv_int_vp_le`

In each graph, we present the type F-score and token accuracy for each method, and mark the best-performing method in terms of each of these evaluation measures with a star ($\star$). The results for syntax-based DLA (S_{POS}, S_{CHUNK} and S_{PARSE}) are based on the BNC in each case. We return to investigate the impact of corpus size on the performance of the syntax-based methods below.

Looking first at the combined results over all lexical types (Figure 1), the most successful method in terms of type F-score is syntax-based DLA, with chunker-based preprocessing marginally outperforming tagger- and parser-based preprocessing (type F-score = 0.641). The most successful method in terms of token accuracy is ontology-based DLA (token accuracy = 0.544).

The figures for token accuracy require some qualification: ontology-based DLA tends to be liberal in its generation of lexical items, giving rise to over 20% more lexical items than the other methods (7,307 vs. 5-6000 for the other methods) and proportionately low type precision. This correlates with an inherent advantage in terms of token accuracy, which we have no way of balancing up in our token-based evaluation, as the treebank data offers no insight into the true worth of false negative lexical items (i.e. have no way of distinguishing between unobserved lexical items which are plain wrong from those which are intuitively correct and could be expected to occur in alternate sets of treebank data). We leave investigation of the impact of these extra lexical items on the overall parser performance (in terms of chart complexity and parse selection) as an item for future research.

The morphology-based DLA methods were around baseline performance overall, with character n-grams marginally more successful than derivational morphology in terms of both type F-score and token accuracy.

Turning next to the results for the proposed methods over nouns, verbs, adjectives and adverbs (Figures 2–5, respectively), we observe some interesting effects. First, morphology-based DLA hovers around baseline performance for all word classes except adjectives, where character n-grams produce the highest F-score of all methods, and nouns, where derivational morphology seems to aid DLA slightly (providing weak support for our original hypothesis in Section 3.2 relating to deverbal nouns and affixation).

[10] Note that the token accuracy is calculated only over the open-class lexical items, not the full ERG lexicon.

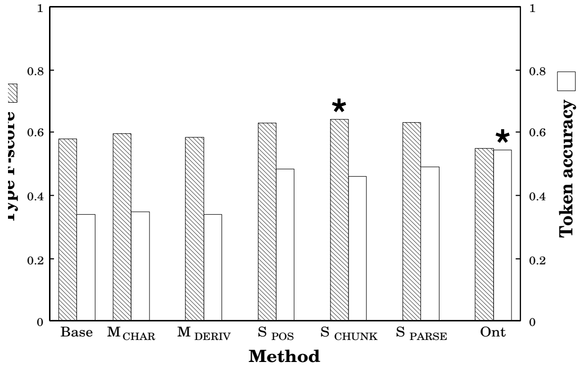


Figure 1: Results for the proposed deep lexical acquisition methods over ALL lexical types

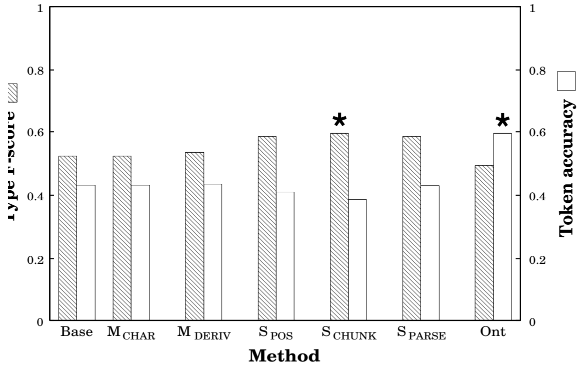


Figure 2: Results for the proposed deep lexical acquisition methods over NOUN lexical types

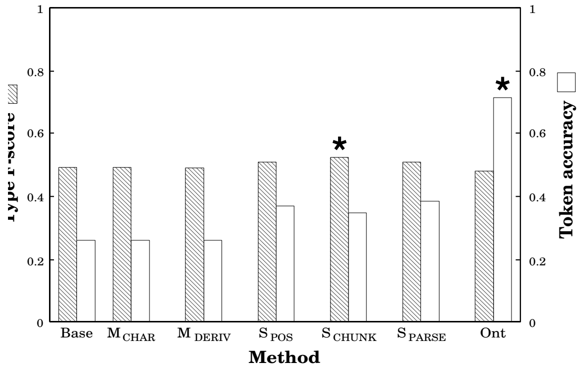


Figure 3: Results for the proposed deep lexical acquisition methods over VERB lexical types

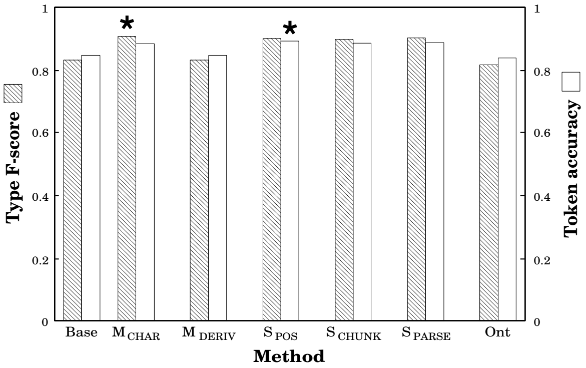


Figure 4: Results for the proposed deep lexical acquisition methods over ADJECTIVE lexical types

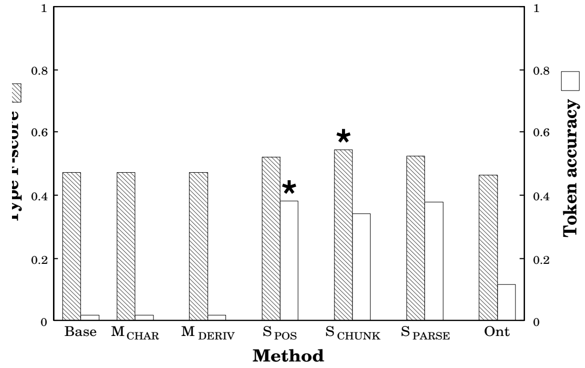


Figure 5: Results for the proposed deep lexical acquisition methods over ADVERB lexical types

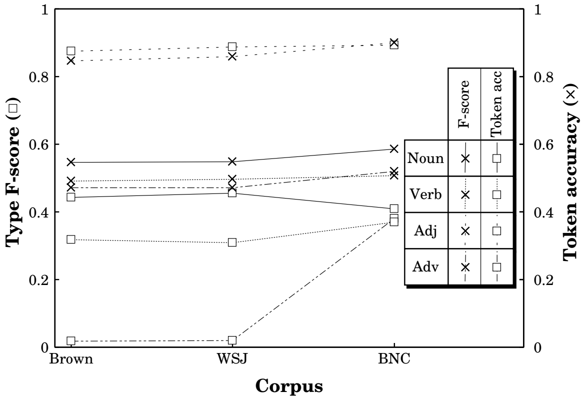


Figure 6: Results for the syntax-based deep lexical acquisition methods over corpora of differing size

Note: Base = baseline, $\mathrm{M}_{\mathrm{CHAR}}$ = morphology-based DLA with character n-grams, $\mathrm{M}_{\mathrm{DERIV}}$ = derivational morphology-based DLA, $\mathrm{S}_{\mathrm{POS}}$ = syntax-based DLA with POS tagging, $\mathrm{S}_{\mathrm{CHUNK}}$ = syntax-based DLA with chunking, $\mathrm{S}_{\mathrm{PARSE}}$ = syntax-based DLA with dependency parsing, and Ont = ontology-based DLA

Syntax-based DLA leads to the highest type F-score for nouns, verbs and adverbs, and the highest token accuracy for adjectives and adverbs. The differential in results between syntax-based DLA and the other methods is particularly striking for adverbs, with a maximum type F-score of 0.544 (for chunker-based preprocessing) and token accuracy of 0.340 (for tagger-based preprocessing), as compared to baseline figures of 0.471 and 0.017 respectively. There is relatively little separating the three styles of preprocessing in syntax-based DLA, although chunker-based preprocessing tends to have a slight edge in terms of type F-score, and tagger-based preprocessing generally produces the highest token accuracy.[11] This suggests that access to a POS tagger for a given language is sufficient to make syntax-based DLA work, and that syntax-based DLA thus has moderately high applicability across languages of different densities.

Ontology-based DLA is below baseline in terms of type F-score for all word classes, but results in the highest token accuracy of all methods for nouns and verbs (although this finding must be taken with a grain of salt, as noted above).

Another noteworthy feature of Figures 2–5 is the huge variation in absolute performance across the word classes: adjectives are very predictable, with a majority class-based baseline type F-score of 0.832 and token accuracy of 0.847; adverbs, on the other hand, are similar to verbs and nouns in terms of their baseline type F-score (at 0.471), but the adverbs that occur commonly in corpus data appear to belong to less-populated lexical types (as seen in the baseline token accuracy of a miniscule 0.017). Nouns appear the hardest to learn in terms of the relative increment in token accuracy over the baseline. Verbs are extremely difficult to get right at the type level, but it appears that ontology-based DLA is highly adept at getting the commonly-occurring lexical items right.

To summarise these findings, adverbs seem to benefit the most from syntax-based DLA. Adjectives, on the other hand, can be learned most effectively from simple character n-grams, i.e. similarly-spelled adjectives tend to have similar syntax, a somewhat surprising finding. Nouns are surprisingly hard to learn, but seem to benefit to some degree from corpus data and also ontological similarity. Lastly, verbs pose a challenge to all methods at the type level, but ontology-based DLA seems to be able to correctly predict the commonly-occurring lexical entries.

Finally, we examine the impact of corpus size on the performance of syntax-based DLA with tagger-based preprocessing.[12] In Figure 6, we examine the relative change in type F-score and token accuracy across the four word classes as we increase the corpus size (from 0.5m words to 1m and finally 100m words, in the form of the Brown corpus, WSJ corpus and BNC, respectively). For verbs and adjectives, there is almost no change in either type F-score or token accuracy when we increase the corpus size, whereas for nouns, the token accuracy actually drops slightly. For adverbs, on the other hand, the token accuracy jumps up from 0.020 to 0.381 when we increase the corpus size from 1m words to 100m words, while the type F-score rises only slightly. It thus seems to be the case that large corpora have a considerable impact on DLA for commonly-occurring adverbs, but that for the remaining word classes, it makes little difference whether we have 0.5m or 100m words. This can be interpreted either as evidence that modestly-sized corpora are good enough to perform syntax-based DLA over (which would be excellent news for low-density languages!), or alternatively that for the simplistic syntax-based DLA methods proposed here, more corpus data is not the solution to achieving higher performance.

Returning to our original question of the “bang for the buck” associated with individual LRs, there seems to be no simple answer: simple word lists are useful in learning the syntax of adjectives in particular, but offer little in terms of learning the other three word classes. Morphological lexicons with derivational information are moderately advantageous in learning the syntax of nouns but little else. A POS tagger seems sufficient to carry out syntax-based DLA, and the word class which benefits the most from larger amounts of corpus data is adverbs, otherwise the proposed syntax-based DLA methods don’t seem to benefit from larger-sized corpora. Ontologies have the greatest impact on verbs and, to a lesser degree, nouns. Ultimately, this seems to lend weight to a “horses for courses”, or perhaps “resources for courses” approach to DLA.

[11] This trend was observed across all three corpora, although we do no present the full results here.

[12] The results for chunker- and parser-based preprocessing are almost identical, and this omitted from the paper.

7 Conclusion

We have proposed three basic paradigms for deep lexical acquisition, based on morphological, syntactic and ontological language resources, and demonstrated the effectiveness of each strategy at learning lexical items for the lexicon of a precision English grammar. We discovered surprising variation in the results for the different DLA methods, with each learning method performing particularly well for at least one basic word class, but the best overall methods being syntax- and ontology-based DLA.

The results presented in this paper are based on one particular language (English) and a very specific style of DLR (a precision grammar, namely the English Resource Grammar), so some caution must be exercised in extrapolating the results too liberally over new languages/DLA tasks. In future research, we are interested in carrying out experiments over other languages and alternate DLRs to determine how well these results generalise and formulate alternate strategies for DLA.

Acknowledgements

This material is based upon work supported in part by NTT Communication Science Laboratories, Nippon Telegraph and Telephone Corporation. We would like to thank the members of the University of Melbourne LT group and the three anonymous reviewers for their valuable input on this research.

References

Timothy Baldwin and Francis Bond. 2003a. Learning the countability of English nouns from corpus data. In *Proc. of the 41st Annual Meeting of the ACL*, pages 463–70, Sapporo, Japan.

Timothy Baldwin and Francis Bond. 2003b. A plethora of methods for learning English countability. In *Proc. of the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP 2003)*, pages 73–80, Sapporo, Japan.

Ted Briscoe and John Carroll. 2002. Robust accurate statistical annotation of general text. In *Proc. of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, pages 1499–1504, Las Palmas, Canary Islands.

Lou Burnard. 2000. *User Reference Guide for the British National Corpus*. Technical report, Oxford University Computing Services.

John Carroll and Alex Fang. 2004. The automatic acquisition of verb subcategorisations and their impact on the performance of an HPSG parser. In *Proc. of the First International Joint Conference on Natural Language Processing (IJCNLP-04)*, pages 107–14, Sanya City, China.

Ann Copestake and Dan Flickinger. 2000. An open-source grammar development environment and broad-coverage English grammar using HPSG. In *Proc. of the 2nd International Conference on Language Resources and Evaluation (LREC 2000)*, Athens, Greece.

Walter Daelemans, Jakub Zavrel, Ko van der Sloot, and Antal van den Bosch. 2003. *TiMBL: Tilburg Memory Based Learner, version 5.0, Reference Guide*. ILK Technical Report 03-10.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, USA.

Dan Flickinger. 2002. On building a more efficient grammar by exploiting types. In Stephan Oepen, Dan Flickinger, Jun'ichi Tsujii, and Hans Uszkoreit, editors, *Collaborative Language Engineering*. CSLI Publications, Stanford, USA.

Frederik Fouvry. 2003. *Robust Processing for Constraint-based Grammar Formalisms*. Ph.D. thesis, University of Essex.

Nizar Habash and Bonnie Dorr. 2003. CATVAR: A database of categorial variations for English. In *Proc. of the Ninth Machine Translation Summit (MT Summit IX)*, pages 471–4, New Orleans, USA.

Eric Joanis and Suzanne Stevenson. 2003. A general feature space for automatic verb classification. In *Proc. of the 10th Conference of the EACL (EACL 2003)*, pages 163–70, Budapest, Hungary.

Anna Korhonen. 2002. *Subcategorization Acquisition*. Ph.D. thesis, University of Cambridge.

Beth Levin. 1993. *English Verb Classes and Alterations*. University of Chicago Press, Chicago, USA.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn treebank. *Computational Linguistics*, 19(2):313–30.

Guido Minnen, John Carroll, and Darren Pearce. 2000. Robust, applied morphological generation. In *Proceedings of the first International Natural Language Genration Conference*, Mitzpe Ramon, Israel.

Grace Ngai and Radu Florian. 2001. Transformation-based learning in the fast lane. In *Proc. of the 2nd Annual Meeting of the North American Chapter of Association for Computational Linguistics (NAACL2001)*, pages 40–7, Pittsburgh, USA.

Stephan Oepen, Dan Flickinger, Kristina Toutanova, and Christoper D. Manning. 2002. LinGO Redwoods: A rich and dynamic treebank for HPSG. In *Proc. of The First Workshop on Treebanks and Linguistic Theories (TLT2002)*, Sozopol, Bulgaria.

Antonio Sanfilippo and Victor Poznański. 1992. The acquisition of lexical knowledge from combined machine-readable dictionary sources. In *Proc. of the 3rd Conference on Applied Natural Language Processing (ANLP)*, pages 80–7, Trento, Italy.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 shared task: Chunking. In *Proc. of the 4th Conference on Computational Natural Language Learning (CoNLL-2000)*, Lisbon, Portugal.

Leonoor van der Beek and Timothy Baldwin. 2004. Crosslingual countability classification with EuroWordNet. In *Papers from the 14th Meeting of Computational Linguistics in the Netherlands*, pages 141–55, Antwerp, Belgium. Antwerp Papers in Linguistics.

Morphology vs. Syntax in Adjective Class Acquisition

Gemma Boleda
GLiCom
Pompeu Fabra University
Barcelona
gemma.boleda@upf.edu

Toni Badia
GLiCom
Pompeu Fabra University
Barcelona
toni.badia@upf.edu

Sabine Schulte im Walde
Computational Linguistics
Saarland University
Saarbrücken
schulte@CoLi.Uni-SB.DE

Abstract

This paper discusses the role of morphological and syntactic information in the automatic acquisition of semantic classes for Catalan adjectives, using decision trees as a tool for exploratory data analysis. We show that a simple mapping from the derivational type to the semantic class achieves 70.1% accuracy; syntactic function reaches a slightly higher accuracy of 73.5%. Although the accuracy scores are quite similar with the two resulting classifications, the kinds of mistakes are qualitatively very different. Morphology can be used as a baseline classification, and syntax can be used as a clue when there are mismatches between morphology and semantics.

1 Introduction

This paper fits into a broader effort addressing the automatic acquisition of semantic classes for Catalan adjectives. So far, no established standard of such semantic classes is available in theoretical or empirical linguistic research. Our aim is to reach a classification that is empirically adequate and theoretically sound, and we use computational techniques as a means to explore large amounts of data which would be impossible to explore by hand to help us define and characterise the classification.

In previous research (Boleda et al., 2004), we developed a three-way classification according to generally accepted adjective properties (see Section 2), and applied a cluster analysis to further examine the classes. While the cluster analysis confirmed our classification to a large extent, it was clear that one of the classes needed further exploration. Also, we used only syntactic features modelled as pairs of POS-bigrams; we explored neither other syntactic features nor the role of morphological evidence for the classification.

In this paper we apply a supervised classification technique, decision trees, for exploratory data analysis. Our aim is to explore the linguistic features and description levels that are relevant for the semantic classification, focusing on morphology and syntax. We check how far we get with morphological information, and whether syntax is helpful to overcome the ceiling reached with morphology.

Decision trees are appropriate for our task, to test and compare sets of features, based on our gold standard. They are also known for their easy interpretation, by reading feature combinations off the tree paths. This property will help us get insight into relevant characteristics of our adjective classes, and in the error analysis.

The paper is structured as follows: Section 2 presents the adjective classification and the gold standard used for the experiments. Sections 3 and 4 explore the morphology-semantics interface and the syntax-semantics interface with respect to the classification proposed, and Section 5 focuses on the differences in the kind of information each level provides for the classification. Sections 6 and 7 are devoted to discussion of related work and conclusions.

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 77–86,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

2 Classification and gold standard

2.1 Classification proposal

To date, no semantic classification of adjectives is generally accepted in theoretical linguistics. Much research in formal semantics has focused on entailment properties , while other kinds of lexical semantic properties are left uncovered. Standard descriptive grammars propose broader classifications (see Picallo (2002) for Catalan), but these usually do not follow a single classification parameter: they mix morphological, syntactic and semantic criteria and end up with classifications that are not consistent.

We are interested in properties of the lexical semantics of adjectives that have a bearing on their syntactic behaviour. This property makes the semantic distinctions traceable at another linguistic level, which is desirable to ensure falsability of the classification criteria. On more practical terms, it also allows the exploitation of the syntax-semantics interface as is usual in Lexical Acquisition, to automate the acquisition of the relevant classes.

Our proposal is largely inspired by the Ontological Semantics framework (Raskin and Nirenburg, 1995). The assumption of an ontology as a model of the world allows us to distinguish linguistic aspects (e.g. syntactic properties) from the actual content of the lexical entries, formalised as a link to an element the ontology. We assume an ontology of basic denotations composed of properties (or attributes), objects (or entities), and events. Adjectives participate in each of these possible denotations, and can be *basic*, *object*-related or *event*-related, depending on their lexical meaning.[1] We next characterise each class.

Basic adjectives are the prototypical adjectives, which denote attributes or properties which cannot be decomposed (*bonic* 'beautiful', *sòlid* 'solid'). Event adjectives have an event component in their meaning. For instance, if something is *tangible* ('tangible'), then it can be touched: *tangible* necessarily evokes a potential event of touching which is embedded in the meaning of the adjective. Other examples are *alterat* ('altered') and *ofensiu* ('offensive'). Similarly, object adjectives have an embedded object component in their meaning: *deformació nasal* ('nasal deformity') can be paraphrased as *deformity that affects the nose*, so that *nasal* evokes the object *nose*. Other examples are *peninsular* ('peninsular') and *sociolingüístic* ('sociolinguistic').

This proposal shares many aspects with discussions in descriptive grammar (2002) and proposals in other lexical resources, such as WordNet (Miller, 1998). In particular, the distinction between prototypical, attribute-denoting adjectives and object-related adjectives is found both in descriptive grammar and in WordNet. As for event-related adjectives, they are not usually found as a class in Romance descriptive grammar, and in WordNet they are distinguished but only if they are participial; other kinds of deverbal adjectives are considered basic (in our terminology). More on the morphology-semantics relationship in Section 3.

Our classification focuses on the semantic content of adjectives, rather than on formal properties such as entailment patterns (contrary to the tradition in formal semantics). The semantic distinctions proposed have an effect on the syntactic distribution of adjectives, as will be shown throughout the paper, and can be exploited in low-level NLP tasks (POS-tagging), and also in more demanding tasks, such as paraphrase detection and generation (e.g. exploiting the relationship *tangible* → *can be touched*, or *deformació nasal* → *deformity affecting the nose*).

2.2 Gold standard

To perform the experiments, we built a set of annotated data based on this classification (*gold standard* from now on). We extracted the lemmata and data for the gold standard from a 16.5 million word Catalan corpus (Rafel, 1994), lemmatised, POS-tagged and shallow parsed with the CatCG tool (Alsina et al., 2002). The shallow parser gives information on the syntactic function of each word (subject, object, etc.), not on phrase structure.

186 lemmata were randomly chosen among all 2564 adjectives occuring more than 25 times in the corpus. 86 of the 186 lemmata were classified by 3 human judges into each of the classes (basic, object, event).[2] In case of polysemy affecting the class as-

[1] Raskin and Nirenburg (1995) account separately for other kinds of adjectives, such as membership adjectives ('fake'). We will abstract away from these less numerous classes.

[2] The 3 human judges were PhD students with training in linguistics, one of which had done research on adjectives. As it was defined, the level of training in linguistics needed for the

signment, the judges were instructed to return the class for the most frequent sense as the primary class, and a secondary class for the other sense.

Polysemy typically arises in cases where an adjective has developed a *noncompositional* sense. One of these cases would be the adjective *puntual*, a denominal adjective (derived from *punt*, ‘point’). Its most frequent sense is ‘punctual’ as in ‘I expect Mary to be punctual for this meeting’. This is a basic meaning, noncompositional in the sense that it cannot be predicted from the meaning of the originating noun in combination with the suffix.

The adjective has a compositional sense, namely, ‘related to a point’ (usually, a point in time), as in *això va ser un esdeveniment puntual*, ‘this was a once-occuring event’. This is the meaning we would expect from the derivation *punt* (‘point’) + *al*, and is an object meaning. In this case, the judge should assign the adjective to two classes, primary basic, secondary object. Compositional meanings are thus those corresponding to active morphological processes, and can be predicted from the meaning of the noun and the derivation with the suffix (be it denominal, deverbal or participial).

The judges had an acceptable 0.74 mean κ agreement (Carletta, 1996) for the assignment of the primary class, but a meaningless 0.21 for the secondary class (they did not even agree on which lemmata were polysemous). As a reaction to the low agreement about polysemy, we incorporated polysemy information from a Catalan dictionary (DLC, 1993). This information was incorporated only in addition to the gathered gold standard: In many cases the dictionary only lists the compositional sense. We added it as a second reading if our judges considered the noncompositional one as most frequent.

One of the authors of the paper classified the remaining 100 lemmata according to the same criteria. For our experiment, we use the complete gold standard containing 186 lemmata (87 basic, 46 event, and 53 object adjectives).

3 Morphological evidence

There is an obvious relationship between the derivational type of an adjective (whether it is denominal, deverbal, or not derived) and the semantic classification we have put forth: Usually, a denominal adjective has an object embedded in its meaning (corresponding to the object denoted by the noun from which it is derived). Similarly, a deverbal or participial adjective tends to denote a relationship with an event (the event denoted by the originating verb), and a nonderived adjective tends to have a basic meaning. Therefore, the simplest classification strategy is to associate each derivational type with a semantic class: nonderived → basic, participial → event, deverbal → event, and denominal → object.

Table 1 reflects the accuracy results of this theoretically defined mapping between morphology and semantics, compared to our gold standard (cases corresponding to the predicted mapping in boldface).[3] For instance, the first line of this table shows that 39 of the 42 nonderived adjectives, predicted to be basic by the morphology-semantics mapping, are actually deemed basic by the human judges, while the remaining 3 are classified as object adjectives.

	basic	event	object	*Total*
nonderived (basic)	**39**	0	3	*42*
deverbal (event)	12	**11**	2	*25*
participial (event)	12	**35**	0	*47*
denominal (object)	24	0	**48**	*72*
Total	*87*	*46*	*53*	*186*
precision	.93	.64	.67	*.74*
recall	.45	1	.91	*.78*
f-score (α = 0.5)	.69	.82	.79	*.76*

Table 1: Morphology-semantics mapping: results

Note that the table correctly reflects the general tendencies just outlined: This simple classification achieves 0.76 f-score. However, there are obvious mismatches. Most of these mismatches are concentrated in the first column, namely many of the deverbal, participial and denominal adjectives (predicted to denote event or object meanings) actually have a basic meaning as their most frequent sense. This fact is reflected in the low recall score for basic adjectives (0.45), and in precision being much lower than recall for the other two classes (0.64 vs. 1 for event, 0.67 vs. 0.91 for object adjectives).

task was quite high.

[3] The morphological information was obtained from a manually constructed electronic database of adjectives, kindly provided by Roser Sanromà (2003).

The mismatches usually correspond to polysemy due to noncompositional senses of the adjectives, such as the denominal adjective *puntual* discussed above. Another case is the participial *abatut*, which compositionally means ‘shot-down’, but is most frequently used as a synonym to ‘depressed, downcast’, and therefore is classified as basic. Similarly, a deverbal adjective such as *radiant* most frequently means ‘happy’, but also has a compositional sense, ‘irradiating’.

Sometimes the compositional meaning is completely lost, as with most deverbal adjectives classified as basic. In some cases the underlying verb no longer exists in Catalan (*horrible*-**horrir*, *compatible*-**compatir*), and they are not perceived as derived.[4] In other cases, although the verb exists, it is a stative predicate (e.g. *inestable*, ‘unstable’, from *estar* ‘stand/be’; *pudent* ‘stinking’, from *pudir*, ‘stink’), and thus are much more similar to basic adjectives than deverbal adjectives deriving from dynamic predicates, such as *ofensiu* (‘offensive’). Aspectuality of the deriving verb is a factor that has to be examined more carefully in the future.

To summarise, the results for the morphology-semantics mapping indicate that there is a clear relationship between these two levels: Morphology does most of the job right, because each morphological rule has an associated semantic operation. However, this level of information has a clear performance ceiling. In case of noncompositional meanings the morphological class will systematically be misleading, which cannot be overcome unless other kinds of information are let into play.

4 Syntactic evidence

If we adhere to the hypothesis that semantics has a reflection in syntactic distribution (basis for most work in Lexical Acquisition), we can expect that syntax gives us a better clue to semantics than morphology, particularly in cases of noncompositional meanings. We expect that adjectives with a noncompositional meaning behave in the syntax as basic adjectives, not as event or object adjectives.

Before getting into the experiments using syntactic information, we briefly present the syntax of adjectives in Catalan and the predictions with respect to the syntactic behaviour of each class.

4.1 Adjective syntax in Catalan

The default function of the adjective in Catalan is that of modifying a noun; the default position is the postnominal one (about 66% of adjective tokens in the corpus modify nouns postnominally). Examples are *taula gran* (‘big table’), *arquitecte tècnic* (‘technical architect’), and *element constitutiu* (‘constitutive element’).

However, some adjectives can appear prenominally, mainly when used non-restrictively (so-called “epithets”; 26% of the tokens occur in prenominal position). In English, this epithetic use is not typically distinguished by position, but some adjectives can epithetically modify proper nouns (‘big John’ vs. ‘*technical John’). ‘Big’ in ‘big John’ does not restrict the reference of ‘John’, but highlights a property. In Catalan and other Romance languages, prenominal position is systematically associated to this use, with proper or common nouns.

The other main function of the adjective is that of predicate in a copular sentence (6% of the tokens), such as *aquesta taula és gran* (‘this table is big’). Other predicative contexts, such as adjunct predicates (as in *la vaig veure borratxa*, ‘I saw her drunk’), are much less frequent: approx. 1% of the adjectives in the corpus.

From empirical exploration and literature review, we gathered the following tentative predictions as to the syntactic behaviour of each class in Catalan:

Basic adjectives occur in predicative environments, have scope over other adjectives modifying the same head (most notably, object adjectives), and can have epithetic uses and therefore occur prenominally.

Event adjectives occur in predicative environments and after object adjectives.

Object adjectives occur in a rigid position, directly after their head noun; they do not allow pred-

[4] The question may arise of whether these adjectives are really deverbal. In the current version of the adjective database, all adjectives bearing a suffix that is active in the Catalan derivational system are classified as derived. The problem is that Catalan shares suffixes with Latin, so that fixed forms from Latin that have been incorporated into Catalan cannot be superficially distinguished from active derived forms.

icative constructions nor epithetic uses (therefore not prenominal position).

4.2 Setup

We modelled the syntactic behaviour of adjectives using three different representation strategies. The values in the three cases were frequency counts, that is, the percentage of occurrence of each adjective in that syntactic environment. The frequency of the adjectives from the gold standard in the corpus ranges from 27 to 7154 (median: 129.5). All in all, 56,692 out of the approx 600,000 sentences in the corpus were used as data for this experiment. We have not analysed the influence of frequency on the results, but each adjective is represented by a reasonable amount of data, so that the representation of the syntactic evidence in terms of frequency is adequate.

The simplest modelling strategy is unigram representation, taking the POS of the word to the left of the adjective and the POS of the word to the right as separate features. Adjectives have a limited syntactic distribution (much more restricted than e.g. verbs), so that even this simple representation should provide relevant evidence. The second one is bigram representation, with features consisting of the POS of the word to the left of the adjective and the POS of the word to the right as a single feature. This representation results in a much larger number of features (see Table 2), thus potentially leading to data sparsenes, but it should be more informative, because left and right context are taken into account at the same time.

The third one is the syntactic function, as given by CatCG. For adjectives, these functions are noun modifier (distinguishing between prenominal and postnominal position), predicate in a copular sentence, and predicative adjunct (more information in Section 4.4). CatCG does not yield completely disambiguated output, and the ambiguous functions were also taken into account, so as not to miss any potentially relevant source of evidence.

To perform the experiment, we used C5.0, a commercial decision tree and rule induction engine developed by Ross Quinlan (Quinlan, 1993). We tried several options, including the default, winnowing, and adaptive boosting. Although the results varied a bit within each representation strategy (boosting tended to perform better, winnowing did not have a homogeneous behaviour), the general picture remained the same as to the relative performance of each level of representation. Therefore, and for clarity of exposure and exploration reasons, we will only present and discuss results using the default options.

For comparison, we ran the tool on the 3 syntactic representation levels and on morphological information, using derivational type, a finer-grained derivational type, and the suffix.[5]

4.3 Results

The results of the experiment, obtained averaging ten 10-fold cross-validation runs, are depicted in Table 2. In this table, *#f* is the number of features for each representation strategy, *size* the size of the trees (number of leaves), *accuracy* the accuracy rate of the classifiers (in percentage), and *SE* the standard error of each parameter. We currently assume a majority baseline, that of assigning all adjectives to the most numerous class (basic). Given that there are 87 basic adjectives and 186 items in the gold standard (see Table 1), this baseline results in 46.8% accuracy.

		size		accuracy	
	#f	mean	SE	mean	SE
baseline	-	-	-	46.8	-
morphology	3	4.3	0.1	70.1	0.3
unigram	24	19.1	0.2	68.8	0.6
bigram	135	18.8	0.4	67.4	0.8
synt. funct.	14	3.5	0.1	73.8	0.3

Table 2: Decision Tree experiment

Note that all four classifiers are well above the majority baseline (46.8%). The best results are obtained with the lowests number of features (3 for morphology, 14 for syntactic function, vs. 24 and 135 for unigram and bigram), and correspondingly, with the smallest trees (average 4.3 and 3.5 leaves for morphology and function, 19.1 and 18.8 for n-grams). We interpret this result as indicating that the levels of description of morphology and syntactic function are more adequate than the n-gram representation, although this is only a tentative conclusion, because the differences in accuracy are not large. Function abstracts away from particular POS

[5] The finer-grained derivational type states whether the adjective is derived from a noun or verb that still exists in Catalan or not.

syntactic function	**basic**		**event**		**object**	
postnominal modifier	.69	+/-.16	.68	+/-.19	.94	+/-.06
prenominal modifier	.07	+/-.09	.02	+/-.04	.01	+/-.03
predicative adjunct	.09	+/-.08	.19	+/-.16	.02	+/-.03
predicate in a copular sentence	.10	+/-.10	.08	+/-.07	.01	+/-.02

Table 3: Average values for the syntactic functions in each adjective class.

environments, and summarises the most relevant information without the data sparseness problems inherent in n-gram representation.

Also noteworthy is that the accuracy rates for syntax are lower than we would have expected, according to the hypothesis that it better reflects synchronic meaning. For the first two syntactic representations, unigrams and bigrams, results are worse than using the simple morphological mapping explained above (respectively 68.8% and 67.4% accuracy, compared to 70.1% accuracy achieved with morphology).[6] Only syntactic function improves upon the morphological results, and only slightly (73.8% average accuracy). However, as will be explored in the rest of the Section, the mistakes of the morphological classifier are qualitatively different from those of the syntactic classifiers, which can be used to gain insight into the nature of the problem handled, and to build better classifiers.

4.4 Error analysis

For the analysis of the results, we will focus on the syntactic function features, because it is the best system and allows clearer exploration of the hypotheses stated so far than the n-gram representation.

Table 3 contains the data for the 4 main syntactic functions for adjectives. For each class (all adjectives classified as basic, event or object in the gold standard), it contains the average percentage of occurence with each syntactic function, along with the standard deviation. A set of 10 remaining syntactic features represented cases not disambiguated by CatCG, which had really low mean values and were rarely used in the DTs.

The values of the 4 syntactic functions confirm to a large extent the predictions made with respect to the syntactic behaviour of each adjective class, but also evidence an additional fact: basic and event adjectives, in the current definition of the classes, have only slight differences in their syntax.

Basic and event adjectives have similar mean values for the default adjective position in Catalan (postnominal modifier; 0.69 and 0.68 mean values), and also for the predicative function in a copular sentence (0.10 and 0.084 mean values). The two-sample t-test confirms that the differences in mean are not significant (p=0.73 and p=0.88 at the 95% confidence interval).[7]

Basic adjectives occur more frequently as prenominal modifiers (0.07 compared to 0.02), but note the large standard deviation (0.09 and 0.04)), which means that there is a large within-class variability. In addition, event adjectives have a larger mean value for the predicative adjunct function (0.19 vs. 0.09), but again, the standard deviation of both classes is very large (0.16 and 0.08). Nevertheless, a t-test returns significant p values ($<$ 0.001, 95% conf. int.) for the differences in mean of these two features, so that they can be used as a clue to the characterisation of the event class.[8] The bias of event adjectives towards predicative uses can be attributed to participials – the most frequent kind of adjectives in the event class (35 vs. 11).

Object adjectives do present a distinct syntactic behaviour: They act (as expected) as rigid postnominal modifiers (mean value 0.94), and cannot be used as prenominal modifiers (mean value 0.01) or as predicates (mean values 0.018 and 0.008 for predicative functions). Also note that the standard deviation for each feature is lower in the case of object adjectives than in the case of basic and event adjectives, which indicates a higher homogeneity of the object class. T-tests for the difference in means with

[6] When using morphological features, DTs used almost only the main derivational type, according to the hypothesis stated in Section 3.

[7] Alternatives “not equal” and “basic smaller than event” respectively.

[8] Alternatives: “basic greater than event” for prenominal modification, “event greater than basic” for predicative adjunct.

respect to the basic and event class return significant p values (< 0.001) except for the difference in prenominal modification values between event and object adjectives (p=0.26).[9]

Decision trees built with this feature set use the information consistent with the observations just outlined. In general, they characterise object adjectives as postnominal modifiers (usual threshold: 0.9), basic adjectives as prenominal modifiers (usual threshold: 0.01), and event adjectives as not being prenominal modifiers. In some trees, information about predicativity is also included (event adjectives act as predicative adjuncts; usual threshold: 0.04).

From the discussion of the feature values, it is to be expected that most of the mistakes when using the syntactic function feature set are due to basic-event confusion, and this is indeed the case. For the error analysis, we divided the gold standard into three equal sets, and successively trained on two sets and classified the third. The classification of the gold standard that resulted is reflected in Table 4 (correctly classified items in boldface).

true class →	basic	event	object	*Total*
basic	**56**	7	5	68
event	18	**35**	4	57
object	13	4	**44**	61
Total	*87*	*46*	*53*	*186*
precision	.82	.61	.72	*.72*
recall	.64	.76	.83	*.69*
f-score	.73	.69	.78	*.73*

Table 4: Syntax-semantics mapping: results

Table 4 shows that the object class is best characterised (0.78 f-score), followed by the basic (0.73) and event (0.69) classes. Particularly low are precision for event (0.61) and recall for basic (0.64) adjectives. This distribution indicates that many adjectives are classified as event while belonging to other classes (18 to basic, 4 to object), and many basic adjectives are classified into other classes (18 as event, 13 as object).

The basic-event confusion mainly takes place with basic adjectives not used as epithets (in prenominal position; *curull* 'full', *dispers* 'scattered') and event adjectives used as epithets (*interminable* 'endless', *ofensiu* 'offensive'). Although more analysis is needed, in many of these cases (such as *interminable*) the underlying verb is stative, which makes the adjectives very similar to basic adjectives, as mentioned in Section 3. The judges reported difficulties particularly in distinguishing event from basic adjectives, which matches the results of the experiments. The classification is fuzzy in this point, and we intend to develop clearer criteria to distinguish adjectives with an "active" event in their lexical meaning from basic adjectives.

As for the basic-object confusion, it is due to two factors. The first one is basic being the default class: In the gold standard, if an adjective does not fit into the other 2 classes, it is considered basic, even if it does not denote a prototypical kind of attribute or property. Examples are *radioactiu* ('radioactive') and *recíproc* 'reciprocal'. These tend to be used less in predicative and epithetic functions.

The second one is polysemy. 4 adjectives classified in the gold standard as polysemous between a basic (primary) and an object (secondary) reading are classified by C5.0 as object because they almost only ($>$ 90% of the time) occur postnominally: *artesanal, mecànic, moral, ornamental* ('artesanal, mechanical, moral, ornamental'). All of these cases have a compositional meaning paraphrasable by 'related-to X', where X is the derived noun, and a noncompositional meaning such as 'automatic' for *mecànic*. The syntactic behaviour of the adjective is mixed according to the two classes, so that the values for environments typical of basic adjectives are too low to meet the thresholds.[10]

To sum up, event adjectives do not seem to have consistent syntactic characteristics that tell them apart from basic adjectives, while object adjectives have a consistent behaviour distinct from the other two classes. This result backs up previous experimentation with clustering (Boleda et al., 2004), where half of the event adjectives were systematically clustered together with basic adjectives.[11] Pol-

[9] Alternatives: all means of basic and event greater than those of object, except for postnominal modification, testing against a greater mean for object.

[10] Note, however, that in 6 other cases with the same polysemy, syntax does tell them apart from typical object adjectives, and are classified as basic (such as the *puntual* case discussed above; see discussion in next Section).

[11] The ones that were distinguished from basic adjectives

ysemy plays a tricky role, because depending on the uses of the adjective it leads to a continuum in the feature values which sometimes does not allow a clear identification of the most frequent sense.

5 Differences between morphology and syntax

A crucial point to understand the roles of morphology and syntax for our semantic classification is the differences in the kinds of mistakes that each of the information level carries with it. From the discussion up to this point, we would expect that the default morphological classification causes less mistakes with event vs. basic, because the deverbal morphological rules carry the associated "related-to-event" meaning. On the contrary, syntax should handle better the cases where the relationship between morphology and semantics is lost, what we have termed noncompositional meanings.

If we compare the mistakes made by each mapping, both morphology and syntax assign the expected class to 103 lemmata (55.4% of the gold standard), and both coincide in assigning a wrong class for 21 (11.3%). The cases where one mapping achieves the right classification and the other one makes a mistake are reflected in Tables 5 and 6.

true class →	basic	event	object	*Total*
basic		7	5	12
event	6		4	10
object	4	4		8
Total	10	11	9	

Table 5: Morphology right, syntax wrong

true class →	basic	event	object	*Total*
basic		2	3	3
event	10			12
object	17			17
Total	27	2	3	

Table 6: Syntax right, morphology wrong

Cases where morphology achieves the right class and syntax does not (Table 6) do not present a very clear pattern, although the basic-event confusion in syntax is indeed reflected as the most numerous in Table 5 (6+7 cases). In absence of a syntactic characterisation of the class, applying the default mapping will yield better results.

As for the cases where syntax classifies correctly and morphology does not (Table 6), they do present a clear pattern: They correspond, as expected, to deverbal (8), participial (2) and denominal (17) adjectives with a meaning that does not correspond to the morphological rule. Among denominals, examples are *elemental* and *horrorós* ('elementary' and 'horrifying'); among deverbals, *raonable* and *present* ('reasonable' and 'present'); among participials, *innat* and *inesperat* ('innate' and 'unexpected').

Note that syntax is most helpful in the identification of basic denominal adjectives (17 cases), providing support for the hypothesis that adjectives with a noncompositional meaning behave in the syntax as basic adjectives, which can be exploited in a lexical acquisition setting. In contrast, event and basic classes not having a clearly distinct syntactic distribution, the syntactic features do not help in telling these two classes apart. This problem accounts for the little overall accuracy improvement from morphology (70.1%) to syntax (73.8%): It improves the object vs. basic distinction, but it does not consistently improve the event vs. basic distinction.

5.1 Combining morphological and syntactic features

The next logical step in building a better classifier for adjectives is to use both morphological and syntactic function information. When doing that, a slightly better result is obtained, although no dramatic jump in improvement: 74.7% mean accuracy averaged across ten 10-fold cross-validation runs, with trees of average 8 leaves (mean accuracy being 70.1% with morphology and 73.8% with syntactic function; see Table 2).

In most of the partitions of the data when using this feature set, the first node uses syntactic evidence (high values for postnominal position for object adjectives vs. the rest), and the second level nodes use the derivational type. The remaining morphological features (suffix, fine-grained derivational type; see footnote 4.2) are seldom used.

In all the decision trees, nonderived adjectives are directly assigned to the basic class, and in 80% par-

were so due to their bearing complements, a parameter orthogonal to the targeted classification.

ticipial adjectives are classified as event. The last rule causes a large number of errors, because 12 out of 47 participles were classified as basic in the gold standard. For the other two derivational types, syntactic evidence is used again in almost all decision trees (99% for deverbal, 80% for denominal adjectives). Deverbal or denominal adjectives that occur prenominally are deemed basic, according to expectation. Contrary to expectation, however, deverbal adjectives that occur predicatively are classified as basic. This result confirms the suspicion that frequent predicative use is associated with participial, but not with other kinds of deverbal adjectives, as stated in Section 4.4.

6 Related work

In recent years much research (Merlo and Stevenson, 2001; Schulte im Walde and Brew, 2002; Korhonen et al., 2003) has aimed at exploiting the syntax-semantics interface for classification tasks, mostly based on verbs. In particular, Merlo and Stevenson (2001) present a classification experiment which bears similarities to ours. They use decision trees to classify intransitive English verbs into three semantic classes: unergatives, unaccusatives, and object-drop. As in our experiments, they define three classes, and use only 60 verbs for the experiments. Merlo and Stevenson identify linguistic features referring to verb argument structure (crucially involving thematic relations), and classify the verbs into the three classes with an accuracy of 69.8%. They compare their results with a random baseline of 33%.

There has been much less research in Lexical Acquisition for adjectives. Early efforts include Hatzivassiloglou and McKeown (1993), a cluster analysis directed to the automatic identification of adjectives belonging to the same scale (such as *cold-tempered-hot*). More recently, Bohnet et al. (2002) used bootstrapping to assign German adjectives to “functional” classes (of a more traditional sort, based on a German descriptive grammar). They relied on ordering restrictions and coordination data which can be adapted to Catalan.

As for Romance languages, the only related work we are aware of is Carvalho and Ranchod (2003), who developed a finite-state approach to disambiguating homograph adjectives and nouns in Portuguese. They manually classified the adjectival uses of the homographs into six syntactic classes with characteristics used in our classification (predicative uses, position with respect to the head noun, etc.). They used that information to build finite state transducers aimed at determinining the POS of the homographs in each context, with a high accuracy (99.3%) and coverage (94%). The research undergone in this paper leads to the automatic acquisition of the classes, defined however at a semantic rather than syntactic level.

7 Conclusion and future work

In this paper, we have presented and discussed the role of two sources of evidence for the automatic classification of adjectives into ontological semantic classes: morphology and syntax. Both levels provide relevant information, as indicated by their respective accuracy results (70.1% for morphology, 73.8% for syntax), both well above a majority baseline (46.8%). Morphology fails in cases of noncompositional meaning, when the relationship to the deriving word has been lost, cases that syntax tends to correctly classify. In contrast, syntax systematically confuses event and basic adjectives due to the lack of a sufficiently distinct syntactic profile of the event class. Therefore, the default morphology-semantics mapping handles these cases better.

Not suprisingly, the best classifier is obtained combining both kinds of information (74.7%), although it is not even 1% better than the syntactic classifier. More research is needed to achieve better ways of combining both levels of description.

We can summarise our results as indicating that morphology can give a reliable initial hypothesis with respect to the semantic class of an adjective, which syntax can refine in cases of noncompositional meaning, particularly for object adjectives. Therefore, morphology can be used as a baseline in future classification experiments.

The experiments presented in this paper also shed light on the characteristics of each class. In particular, we have shown that event adjectives do not have a homogeneous and distinct syntactic profile. One factor to take into account is that the morphological variability within the class (suffixes *-ble*, *iu*, *nt*, participles) is associated with a high semantic variability. This semantic variability is not found in the object class, where the several suffixes (*al*, *ic*, *à*, etc.) all have a similar semantic effect. Another factor which seems to play a role, and which has been identified in the error analysis, is the aspectuality of the deriving verb, particularly whether it is stative or dynamic. In the near future, we intend to use the best classifier to automatically classify more adjectives of our database, so as to allow further exploration of the data and a clearer definition of the class.

A major issue we leave for future research is polysemy detection. Up to now, we have only aimed at single-class classification, and not attempted to capture multiple uses of an adjective. E.g. the approach in Bohnet et al. (2002) could be adapted to Catalan: We can use data on coordination and ordering for polysemy detection, once the class of the most frequent sense is established with the methodology explained in this paper.

Finally, the results presented in this paper seem to point in a fruitful direction for the study of adjective semantics: Adjectives that are flexibly used, those that fully exploit the syntactic possibilities of the language (in Catalan, being used predicatively and as epithets), tend to correspond to adjectives with a basic meaning, that is, tend to be viewed as a compact attribute, as a prototypical adjective. In contrast, derived adjectives which retain much of the semantic link to the noun or verb from which they derive do not behave like prototypical adjectives, are tied to certain positions, and do not exhibit the full range of syntactic possibilities of adjectives as a class. We intend to explore the consequences of this hypothesis in more detail in the future.

Acknowledgements

Many thanks to the people who have manually annotated the data: Àngel Gil, Martí Quixal, Roser Sanromà. Also thanks to Louise McNally, Maite Melero, Martí Quixal, and three anonymous reviewers for revision and criticism of previous versions of the paper. We thank Eric Joanis, Alexander Koller, and Oana Postolache for suggestions that lead to this paper. Special thanks are due to Roser Sanromà for kindly providing us with her manual morphological classification (Sanromà, 2003), and to the Institut d'Estudis Catalans for lending us the research corpus. This work is supported by the Fundación Caja Madrid.

References

À. Alsina, T. Badia, G. Boleda, S. Bott, À. Gil, M. Quixal, and O. Valentín. 2002. CATCG: a general purpose parsing tool applied. In *Proceedings of the 3rd LREC*, pages 1130-1135.

B. Bohnet, S. Klatt, and L. Wanner. 2002. An approach to automatic annotation of functional information to adjectives with an application to German. In *Proceedings of the Workshop on Linguistic Knowledge Acquisition and Representation at the 3rd LREC Conference*.

G. Boleda, T. Badia, and E. Batlle. 2004. Acquisition of semantic classes for adjectives from distributional evidence. In *Proceedings of the 20th COLING*, pages 1119–1125.

J. Carletta. 1996. Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254.

P. Carvalho and E. Ranchhod. 2003. Analysis and disambiguation of nouns and adjectives in Portuguese by FST. In *Proceedings of the Workshop on Finite-State Methods for Natural Language Processing at EACL2003*, pages 105–112.

DLC. 1993. *Diccionari de la Llengua Catalana*. Enciclopèdia Catalana, Barcelona, third edition.

V. Hatzivassiloglou and K. R. McKeown. 1993. Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. In *Proceedings of the 31st ACL*, pages 172–182.

A. Korhonen, Y. Krymolowski, and Z. Marx. 2003. Clustering polysemic subcategorization frame distributions semantically. In *Proceedings of the 41st ACL*, pages 64–71.

P. Merlo and S. Stevenson. 2001. Automatic verb classification based on statistical distributions of argument structure. *Computational Linguistics*, 27(3):373–408.

K. J. Miller. 1998. Modifiers in WordNet. In Christiane Fellbaum, editor, *WordNet: an Electronic Lexical Database*, pages 47–67. MIT, London.

C. Picallo. 2002. L'adjectiu i el sintagma adjectival. In Joan Solà, editor, *Gramàtica del català contemporani*, pages 1643–1688. Empúries, Barcelona.

R. Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Francisco.

J. Rafel. 1994. Un corpus general de referència de la llengua catalana. *Caplletra*, 17:219–250.

V. Raskin and S. Nirenburg. 1998. An applied ontological semantic microtheory of adjective meaning for natural language processing. *Machine Translation*, 13:135–227.

S. Schulte im Walde and C. Brew. 2002. Inducing German semantic verb classes from purely syntactic subcategorisation information. In *Proceedings of the 40th ACL*, pages 223–230.

Roser Sanromà. 2003. Aspectes morfològics i sintàctics dels adjectius en català. Master's thesis, Universitat Pompeu Fabra.

Automatic Acquisition of Bilingual Rules for Extraction of Bilingual Word Pairs from Parallel Corpora

Hiroshi Echizen-ya
Dept. of Electronics and Information
Hokkai-Gakuen University
S26-Jo W11-Chome, Chuo-ku
Sapporo, 064-0926 Japan
echi@eli.hokkai-s-u.ac.jp

Kenji Araki
Graduate School of Information Science
and Technology, Hokkaido University
N14-Jo W9-Chome, Kita-ku
Sapporo, 060-0814 Japan
araki@media.eng.hokudai.ac.jp

Yoshio Momouchi
Dept. of Electronics and Information
Hokkai-Gakuen University
S26-Jo W11-Chome, Chuo-ku
Sapporo, 064-0926 Japan
momouchi@eli.hokkai-s-u.ac.jp

Abstract

In this paper, we propose a new learning method to solve the sparse data problem in automatic extraction of bilingual word pairs from parallel corpora with various languages. Our learning method automatically acquires rules, which are effective to solve the sparse data problem, only from parallel corpora without any bilingual resource (e.g., a bilingual dictionary, machine translation systems) beforehand. We call this method **I**nductive **C**hain **L**earning (ICL). The ICL can limit the search scope for the decision of equivalents. Using ICL, the recall in three systems based on similarity measures improved respectively 8.0, 6.1 and 6.0 percentage points. In addition, the recall value of GIZA++ improved 6.6 percentage points using ICL.

1 Introduction

1.1 Sparse data problems in extraction of bilingual word pairs

Many studies of automatic extraction of bilingual word pairs have been reported. Most studies have used similarity measures (Manning and Schütze, 1999; Sadat et al., 2002) because they are language-independent. However, these studies are insufficient because of the sparse data problem. For example, we would like to obtain (book; 本 [*hon*[1]]) as the bilingual word pair from (Your book is on the table.; テーブル/[2]に/あなた/の/本/が/あり/ます. [*teburu ni anata no hon ga ari masu.*]) using the Dice coefficient (Smadja et al., 1996) automatically. The Dice coefficient is defined as

$$Dice(W_S, W_T) = \frac{2a}{(a+b)+(a+c)} \qquad (1)$$

In that equation, 'a' is the number of pieces in which both the **S**ource **L**anguage (SL) word W_S and **T**arget **L**anguage (TL) word W_T were found; 'b' is the number of pieces in which only W_S was found; and 'c' is the number of pieces in which only W_T was found.

In the case of using the Dice coefficient, the system cannot extract only (book; 本 [*hon*]) when the respective frequencies of "book", "本 [*hon*]" and "テーブル [*teburu*]" are 1. That is, the similarity value between "book" and "本 [*hon*]" becomes $1.0 (= \frac{2\times1}{1+1})$; the similarity value between "book" and "テーブル [*teburu*]" also be-

[1] Italics means Japanese pronunciation.

[2] '/' in Japanese sentences are inserted after each morpheme because Japanese is an agglutinative language.

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 87–96,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

comes 1.0($= \frac{2\times1}{1+1}$). This obstacle is common among methods based on similarity measures.

1.2 Basic idea for solution of the sparse data problem

We propose a new learning method to solve this sparse data problem. We call this method **I**nductive **C**hain **L**earning (ICL). For example, in (Your book is on the table.; テーブル/に/あなた/の/本/が/あり/ます. [*teburu ni anata no hon ga ari masu.*]), a system using ICL uses the information that "your" corresponds to "あなた/の [*anata no*]." Moreover, it uses the information that equivalents of words that adjoin the right side of "your" exist on the right side of "あなた/の [*anata no*]" in TL sentences. Using such bilingual rules, the system can extract only (book; 本 [*hon*]). This fact indicates that the system limits the search scope for the decision of equivalents in TL sentences. Consequently, ICL is effective to solve the sparse data problem. In this study, bilingual rules are acquired automatically only from parallel corpora by view of learning (Echizen-ya et al., 2002). The system using ICL extracts bilingual word pairs by applying the acquired bilingual rules to bilingual sentence pairs in parallel corpora. Therefore, the system using ICL causes a chain reaction in the acquisition of bilingual rules and the extraction of bilingual word pairs. The main advantages of ICL are the following three:

(1) The system using ICL requires no bilingual resource (e.g., a bilingual dictionary, machine translation systems) beforehand. All bilingual rules are acquired automatically solely from the parallel corpora. Moreover, the system using ICL extracts bilingual word pairs using only acquired bilingual rules to solve the sparse data problem.

(2) The system using ICL is effective for parallel corpora with various languages for which the grammatical structures of SL differ from the grammatical structures of TL (i.e., English – Japanese, not English – French, English – German) through the use of acquired bilingual rules. The bilingual rules can locate the information to cope with the difference word orders of SL and TL.

(3) The system using ICL can extract bilingual word pairs even when the frequencies of the pairs of the co-occurrence words and the bilingual word pairs are only 1 in a parallel corpus. For example, when the bilingual rule (your @; あなた/の/@[*anata no* @]) exists, the system using ICL can extract (book; 本 [*hon*]) even when the frequency of the pairs of "your" and "book" is only 1. This fact indicates that the system using ICL can extract not only high-frequency bilingual word pairs, but also low-frequency bilingual word pairs.

We applied this ICL to three systems based on the Dice coefficient, Yates' χ^2 (Hisamitsu and Niwa, 1996), and **A**kaike's **I**nformation **C**riterion (AIC) (Akaike, 1974). For evaluation experiments, five kinds of parallel corpora: English – Japanese, French – Japanese, German – Japanese, Shanghai-Chinese – Japanese and Ainu[3] – Japanese parallel corpora were used as evaluation data. Evaluation experiments indicated that, using ICL in the systems based on the Dice coefficient, Yates' χ^2 and AIC, the respective recall values improved 8.0, 6.1 and 6.0 percentage points. In addition, using ICL, the recall of the statistical word-alignment model GIZA++ (Och, 2003) improved 6.6 percentage points. Therefore, we confirmed that ICL is effective to solve the sparse data problem in the extraction of bilingual word pairs from parallel corpora with various languages.

1.3 Related works

Several methods based on the co-occurrence of words have been proposed. (Fung, 1995) proposed a method that specifically examines context heterogeneity, which indicates the number of kinds of words that adjoin SL words. (Rapp, 1999) proposed a method that uses co-occurrence vectors based on the two words that

[3] The Ainu language is spoken by some members of the Ainu ethnic group of northern Japan and Sakhalin. Ainu language is independent from, but similar to, Japanese and Korean.

adjoin SL words on the right side and left side. Moreover, (Fung, 1998; Kaji and Aizono, 1996) proposed methods that uses co-occurrence vectors based on all words that exist in the existing bilingual dictionary, among sentences. (Tanaka and Iwasaki, 1996) presented a translation matrix that provides co-occurring information translated from the source into the target, and obtains bilingual word pairs by determining the best translation matrix. Ultimately, these methods depend on the existing bilingual dictionary. Therefore, it is difficult to extract bilingual word pairs from parallel corpora with various languages when a sufficient bilingual dictionary does not exist. In contrast, the system using ICL automatically can extract bilingual word pairs without an existing bilingual dictionary as a bilingual resource.

Regarding methods for acquisition translation templates, (McTait, 1997; Güvenir and Cicekli, 1998) proposed methods that acquires bilingual templates using common parts and different parts. However, such methods require many similar bilingual sentence pairs to extract sufficient translation templates. Moreover, K-vec (Fung and Church, 1994) is unable to extract low-frequency bilingual word pairs. The algorithm is applicable only to bilingual word pairs that occur with a frequency greater than three.

In addition, statistical word-alignment methods (Brown et al., 1993; Melamed, 2000; Och and Ney, 2003; Nie*β*en and Ney, 2004) have been proposed, but they are also insufficient. That is, the statistical word-alignment methods cannot extract bilingual word pairs efficiently when the frequencies of many bilingual word pairs are low. (Watanabe and Sumita, 2003) proposed a method by which the decoder uses some translation examples whose source part is similar to the input. However, numerous translation examples are necessary as a bilingual resource. That is, it is difficult to deal with languages for which translation examples are not sufficiently obtainable. In contrast, ICL can extract bilingual rules and bilingual word pairs efficiently, even from a small parallel corpus.

2 Outline

Figure 1 shows an outline of a system using ICL. The ICL corresponds to three processes: a method based on bilingual rules, a method based on two bilingual sentence pairs, and the decision process of bilingual word pairs.

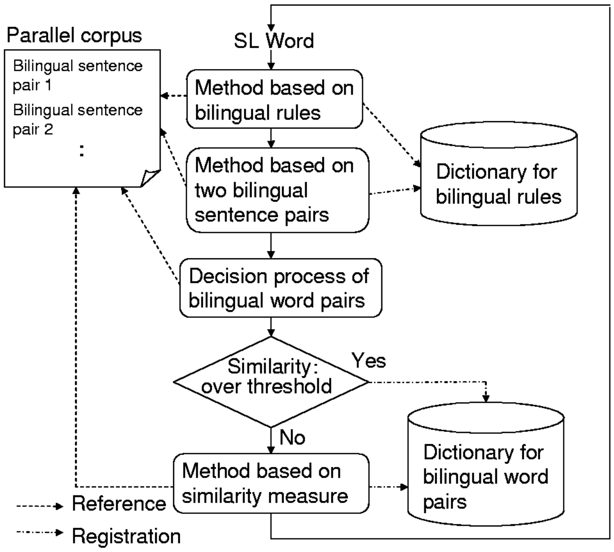
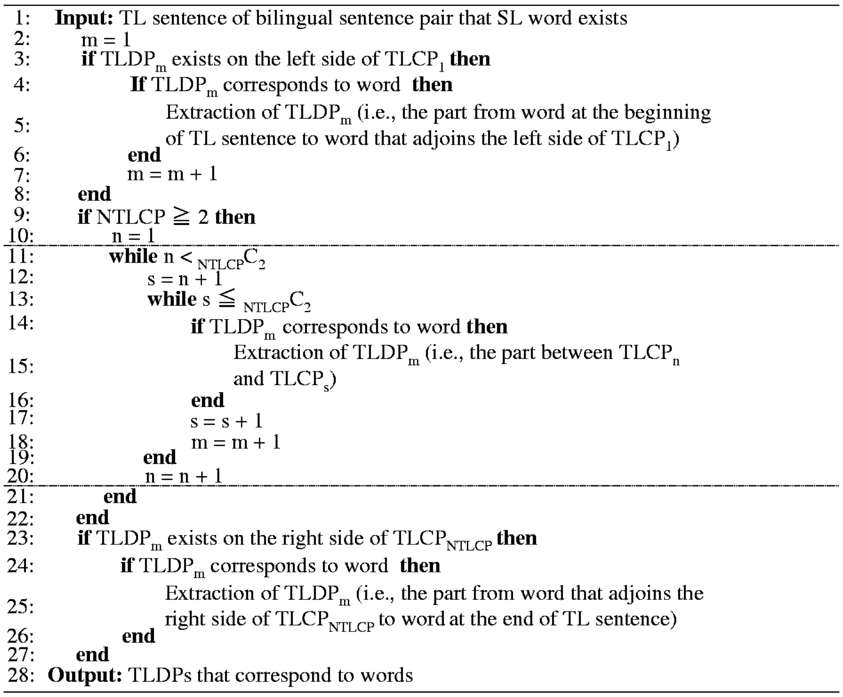

Parallel corpus
Bilingual sentence pair 1
Bilingual sentence pair 2
:
SL Word
Method based on bilingual rules
Method based on two bilingual sentence pairs
Decision process of bilingual word pairs
Similarity: over threshold
Yes
No
Method based on similarity measure
Dictionary for bilingual rules
Dictionary for bilingual word pairs
Reference
Registration

Figure 1: Process flow.

First, the user inputs the SL words of bilingual word pairs. In methods based on bilingual rules, the system extracts bilingual word pairs using the acquired bilingual rules in the dictionary for bilingual rules. In this paper, the bilingual rules are the rules for extracting new bilingual word pairs. In all extracted bilingual word pairs, similarity values between SL words and TL words are assigned using similarity measure. In the method based on two bilingual sentence pairs, the system obtains bilingual word pairs and new bilingual rules using the bilingual sentence pairs that SL words exist and other bilingual sentence pairs. Moreover, in the decision process of bilingual word pairs, the system chooses the most suitable bilingual word pairs using their similarity values when several bilingual word pairs candidates exist. The system compares the similarity values of chosen bilingual word pairs with a threshold value. Consequently, the system registers the chosen bilingual word pairs to the dictionary for bilingual word pairs when their re-



```
Input: TL sentence of bilingual sentence pair that SL word exists
  $m = 1$
  if $TLDP_m$ exists on the left side of $TLCP_1$ then
      If $TLDP_m$ corresponds to word then
          Extraction of $TLDP_m$ (i.e., the part from word at the beginning
          of TL sentence to word that adjoins the left side of $TLCP_1$)
      end
      $m = m + 1$
  end
  if $NTLCP \geqq 2$ then
      $n = 1$
      while $n < {}_{NTLCP}C_2$
          $s = n + 1$
          while $s \leqq {}_{NTLCP}C_2$
              if $TLDP_m$ corresponds to word then
                  Extraction of $TLDP_m$ (i.e., the part between $TLCP_n$
                  and $TLCP_s$)
              end
              $s = s + 1$
              $m = m + 1$
          end
          $n = n + 1$
      end
  end
  if $TLDP_m$ exists on the right side of $TLCP_{NTLCP}$ then
      if $TLDP_m$ corresponds to word then
          Extraction of $TLDP_m$ (i.e., the part from word that adjoins the
          right side of $TLCP_{NTLCP}$ to word at the end of TL sentence)
      end
  end
Output: TLDPs that correspond to words
```

Figure 2: The algorithm of method based on two bilingual sentence pairs.

spective similarity values are greater than the threshold value.

In the method based on similarity measure, the system extracts bilingual word pairs using only one similarity measure (*i.e.*, the Dice coefficient, Yates' χ^2, AIC) from bilingual sentence pairs that SL words exist without ICL. It does so when their similarity values are not greater than the threshold or when no bilingual word pairs are extracted in the ICL process.

3 Process

3.1 Method based on two bilingual sentence pairs

In the method based on two bilingual sentence pairs, the system acquires bilingual rules using the bilingual sentence pairs that SL words exist and other bilingual sentence pairs. The bilingual word pairs for SL words are also extracted. The system obtains bilingual rules using common parts between two bilingual sentence pairs. That is, the word strings for which the frequencies are very low are used as bilingual rules. Using such low-frequency word strings, the bilingual rules are acquired easily only from parallel corpus. In this paper, the respective common parts between SL sentences of two bilingual sentence pairs are called $\mathrm{SLCP}_{\mathrm{i}=1,\ldots,\mathrm{NSLCP}}$; the respective common parts between TL sentences of two bilingual sentence pairs are called $\mathrm{TLCP}_{\mathrm{i}=1,\ldots,\mathrm{NTLCP}}$; the respective different parts between TL sentences of two bilingual sentence pairs are called $\mathrm{TLDP}_{\mathrm{m}=1,2,3,\ldots}$. In addition, the number of SLCPs is called NSLCP; the number of TLCPs is called NTLCP. The details of the process based on two bilingual sentence pairs are the following:

P1-(1) The system selects bilingual sentence pairs for which SL words exist from a parallel corpus. Moreover, the system chooses the bilingual sentence pairs that have SLCPs and TLCPs as the bilingual sentence pairs with SL words. In that case, SLCPs must adjoin SL words in SL sentences.

P1-(2) The system extracts TLDPs that correspond to nouns, verbs, adjectives, adverbs, or conjunctions from TL sentences of bilingual sentence pairs for which SL words exist. Figure 2 shows the algorithm of this process. In lines 11 and 13 of Fig. 2, ${}_{\mathrm{NTLCP}}\mathrm{C}_2$ indicates $\frac{\mathrm{NTLCP}!}{2!(\mathrm{NTLCP}-2)!}$. That is, it means the number of combinations based on two TLCPs.

P1-(3) The system obtains bilingual word pairs by combining SL words and extracted TLCPs.

P1-(4) The system acquires bilingual rules using the extracted TLDPs. The details of this process are the following:

(i) The system replaces SL words and the extracted TLDPs with variables in the bilingual sentence pairs for which SL words exist.

(ii) The system extracts all pairs of each SLCP and variable, and all pairs of each TLCP and variable from bilingual sentence pairs with variables obtained by process (i) of P1-(4).

(iii) The system generates bilingual rules using all combinations of the pairs of SLCPs and variables, and the pairs of TLCPs and variables.

(iv) The system calculates the similarity values between SLCPs and TLCPs in the acquired bilingual rules using the Dice coefficient function (1); it registers the bilingual rules to the dictionary for bilingual rules.

Figure 3 shows an acquisition example of bilingual rules using two English – Japanese bilingual sentence pairs. The system selects bilingual sentence pair 1, for which "house" exists. Furthermore, the system chooses the bilingual sentence pair 2 that have SLCP and TLCPs as the bilingual sentence pairs with SL words by process P1-(1). In Fig. 3, "this" is SLCP in SL sentences of bilingual sentence pairs 1 and 2; it adjoins an SL word "house" in SL sentence of bilingual sentence pair 1. First, the system determines the TLDP that adjoins the left side of TLCP_1 by processes of lines 3 to 8

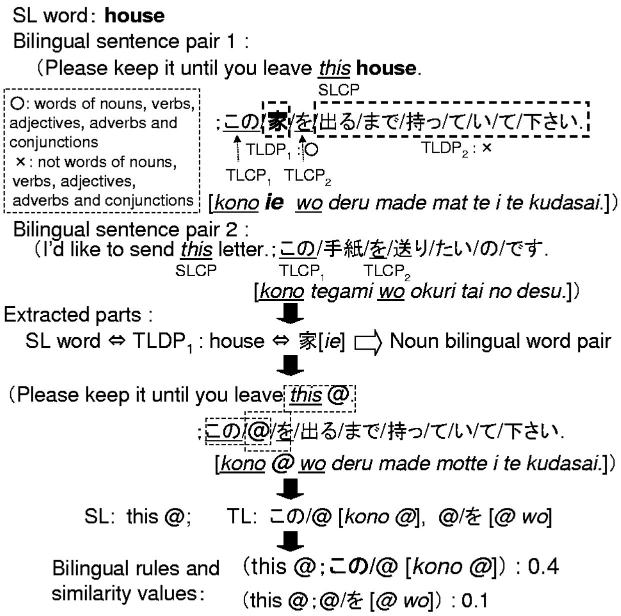

SL word: house
Bilingual sentence pair 1 :
(Please keep it until you leave this house.
SLCP
;この/家/を/出る/まで/持っ/て/い/て/下さい.
TLDP1 : ○
TLDP2 : ×
TLCP1 TLCP2
[kono ie wo deru made mat te i te kudasai.])
○: words of nouns, verbs, adjectives, adverbs and conjunctions
×: not words of nouns, verbs, adjectives, adverbs and conjunctions
Bilingual sentence pair 2 :
(I'd like to send this letter.;この/手紙/を/送り/たい/の/です.
SLCP TLCP1 TLCP2
[kono tegami wo okuri tai no desu.])
Extracted parts :
SL word ⇔ TLDP1 : house ⇔ 家[ie]
Noun bilingual word pair
(Please keep it until you leave this @.
;この/@/を/出る/まで/持っ/て/い/て/下さい.
[kono @ wo deru made motte i te kudasai.])
SL: this @; TL: この/@ [kono @], @/を [@ wo]
Bilingual rules and similarity values:
(this @;この/@ [kono @]) : 0.4
(this @;@/を [@ wo]) : 0.1

Figure 3: An acquisition example of bilingual rules using two bilingual sentence pairs.

in Fig. 2. However, in TL sentences of bilingual sentence pair 1, the word that adjoins the left side of TLCP_1 ("この [*kono*]") does not exist. Therefore, TLDP is not extracted by this process. The system then determines TLDPs using the parts exist between two TLCPs by the processes of lines 9 to 22 in Fig. 2. In TL sentences of bilingual sentence pair 1, one TLDP exists because the number of combinations based on two TLCPs is 1 by ${}_{\mathrm{NTLCP:2}}\mathrm{C}_2 = \frac{2!}{2!(2-2)!} = 1$. That is, "家 [*ie*]" that exists between TLCP_1 ("この [*kono*]") and TLCP_2 ("を [*wo*]") is determined as TLDP_1. Moreover, the system determines the TLDP that adjoins the right side of $\mathrm{TLCP}_{\mathrm{NTLCP:2}}$ by the processes of lines 23 to 27 in Fig. 2. In TL sentences of bilingual sentence pair 1, "出る/まで/持っ/て/い/て/下さい [*deru made mot te i te kudasai*]" is determined as TLDP_2 because it is the part from the word that adjoins the right side of TLCP_2 ("を [*wo*]") to the word at the end of TL sentence. Among two extracted TLDPs, the TLDP that corresponds to word of noun, verb, adjective, adverb, or conjunction is TLDP_1 ("家 [*ie*]") that is noun word. TLDP_2 ("出る/まで/持っ/て/い/て/下さい [*deru made mot te i te kudasai*]") is

```
Input: SL word
    while Selection of bilingual sentence pair that SL word exist, and selection of ICL
          rule that has SLCP and TLCP to the selected bilingual sentence pair
        if Variable exists on the right side of TLCP in TL part of ICL rule then
            i = 0
            while i < NTLCP
                Extraction of TL word (i.e., word of noun, verb, adjective, adverb
                and conjunction) that adjoins the right side of TLCP$_i$ in TL sentence
                i = i + 1
            end
        end
        if Variable exists on the left side of TLCP in TL part of ICL rule then
            i = 0
            while i < NTLCP
                Extraction of TL word (i.e., word of noun, verb, adjective, adverb
                and conjunction) that adjoins the left side of TLCP$_i$ in TL sentence
                i = i + 1
            end
        end
    end
    Calculation of similarity value between SL word and each extracted TL word using
    the cosine function (1)
    Extraction of bilingual word pair by combining SL word and each TL word
Output: Bilingual word pairs
```

Figure 4: The extraction algorithm of bilingual word pairs based on bilingual rules.

verb phrase, not word. Therefore, only (house; 家 [*ie*]) is obtained by combining the SL word ("house") and the extracted TLDP ("家 [*ie*]") by process P1-(3). In addition, the system replaces "house" and "家 [*ie*]" with variable "@" by process (i) of P1-(4). As a result, (this @; この/@[*kono* @]), (this @;@/を [@ *wo*]) are acquired as bilingual rules by process (ii) and (iii) of P1-(4). Similarity values in the acquired bilingual rules (this @; この/@[*kono* @]) and (this @;@/を [@ *wo*]) are calculated using Dice coefficient function (1) by process (iv) of P1-(4). The similarity value of (this @; この/@[*kono* @]) is higher than that of (this @;@/を [@ *wo*]) because (this @; この/@[*kono* @]) is the correct bilingual rule; and (this @;@/を [@ *wo*]) is the erroneous bilingual rule. That is, "this" corresponds to "この [*kono*]", not "を [*wo*]" in Japanese. In this paper, the parts extracted from SL sentences are called SL parts; the parts extracted from TL sentences are called TL parts.

3.2 Method based on bilingual rules

In the method based on bilingual rules, the system extracts bilingual word pairs using the bilingual rules acquired by the method based on two bilingual sentence pairs. The system can limit the search scope for the decision of equivalents in the TL sentences by the use of bilingual rules. Figure 4 gives the extraction algorithm of bilingual word pairs based on bilingual rules.

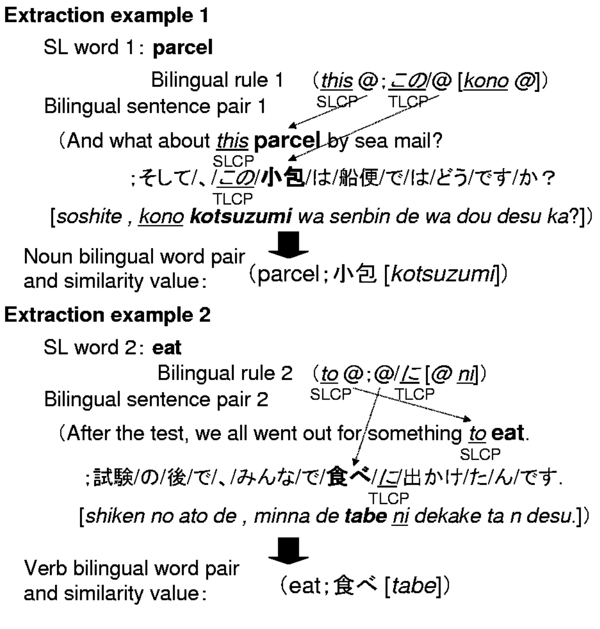

Extraction example 1
SL word 1 : parcel
Bilingual rule 1 （this @ ; この/@ [kono @]）
Bilingual sentence pair 1
SLCP
TLCP
（And what about this parcel by sea mail?
SLCP
；そして/、/この/小包/は/船便/で/は/どう/です/か？
TLCP
[soshite , kono kotsuzumi wa senbin de wa dou desu ka?]）
Noun bilingual word pair and similarity value :
（parcel ; 小包 [kotsuzumi]）
Extraction example 2
SL word 2 : eat
Bilingual rule 2 （to @ ; @/に [@ ni]）
Bilingual sentence pair 2
SLCP
TLCP
（After the test, we all went out for something to eat.
SLCP
；試験/の/後/で/、/みんな/で/食べ/に/出かけ/た/ん/です.
TLCP
[shiken no ato de , minna de tabe ni dekake ta n desu.]）
Verb bilingual word pair and similarity value :
（eat ; 食べ [tabe]）

Figure 5: Examples of extraction of bilingual word pairs based on bilingual rules.

Figure 5 shows examples of extraction of bilingual word pairs from English-Japanese bilingual sentence pairs in the method based on bilingual rules. In example 1 of Fig. 5, (parcel; 小包 [*kotsuzumi*]) is extracted as the noun bilingual word pair using (this @; この/@[*kono @*]) acquired in Fig. 3. First, the system selects bilingual sentence pair 1 that SL word 1 "parcel" exists from a parallel corpus. Moreover, the system selects bilingual rule 1 (this @; この/@[*kono @*]) from the dictionary for bilingual rules because the variable "@" exists on the right side of SLCP ("this") in the SL part of bilingual rule 1, and SL word 1 "parcel" also exists on the right side of SLCP ("this") in the SL sentence of bilingual sentence pair 1. The system then extracts TL words that adjoin the right side of TLCP because the variable "@" exists on the right side of TLCP ("この *kono*") in the TL part of bilingual rule 1. Using bilingual rule 1, noun word "小包 [*kotsuzumi*]", which exists on the right side of TLCP ("この *kono*") is extracted from TL sentence of bilingual sentence pair 1. As a result, the system can obtain (parcel; 小包 [*kotsuzumi*]) as the noun bilingual word pair.

In example 2 of Fig. 5, (eat; 食べ [*tabe*]) is extracted as the verb bilingual word pair using bilingual rule 2 (to @;@/に [*@ ni*]). The system selects bilingual sentence pair 2, in which SL word 2 "eat" exists from a parallel corpus. Moreover, the system selects bilingual rule 2 (to @;@/に [*@ ni*]) from the dictionary for bilingual rules because the variable "@" exists on the right side of SLCP ("to") in the SL part of bilingual rule 2, and SL word 2 "eat" also exists on the right side of SLCP ("to") in SL sentence of bilingual sentence pair 2. The system then extracts TL words that adjoin the left side of TLCP because the variable "@" exists on the left side of TLCP ("に [*ni*]") in the TL part of bilingual rule 2. Using bilingual rule 2, verb word "食べ [*tabe*]", which adjoins the left side of TLCP ("に [*ni*]") is extracted from the TL sentence of bilingual sentence pair 2. The system calculates the similarity value between "eat" and "食べ [*tabe*]" using the Dice coefficient function (1), and registered (eat; 食べ [*tabe*]) into the dictionary of bilingual word pairs. The system determines the most suitable bilingual word pairs according to their similarity values when several bilingual word pairs have been extracted as described in section 3.3.

Using the bilingual rules, the system can decrease the number of candidates of equivalents for SL words. In example 2 of Fig. 5, the system could decrease the number of candidates of equivalents for "eat" using the bilingual rule (to @;@/に [*@ ni*]). All words of nouns, or verbs "試験 [*shiken*]", "後 [*ato*]", "みんな [*minna*]", "食べ [*tabe*]", "出かけ [*dekake*]", and "ん [*n*]" become candidates of equivalents for "eat" when ICL is not used. In contrast, only "食べ [*tabe*]" becomes candidates of equivalents for "eat" using ICL. This fact indicates that ICL is effective to solve the sparse data problem. Moreover, the system can extract bilingual word pairs from parallel corpora of various languages for which the grammatical structure of SL differs from the structure of TL. For example, in the bilingual rule 2 (to @;@/に [*@ ni*]), the variable "@" exists on the right side of "to." In contrast, in the TL part, the variable "@" exists on the left side of "に [*ni*]." Therefore, bilingual rules have the knowledge to cope with the different word order between SL and TL.

3.3 Decision process of bilingual word pair

The system determines the most suitable bilingual word pairs according to their similarity values when several bilingual word pairs have been extracted. The details of this process are the following:

P2-(1) The system selects the bilingual word pairs that have the highest similarity values.

P2-(2) When several bilingual word pairs with identical similarity values exist, the system selects the bilingual word pairs that used bilingual rules with the highest similarity values.

P2-(3) The system selects the bilingual word pairs that appear in a parallel corpus for the first time when it cannot choose only one bilingual word pair by processes P2-(1) and P2-(2).

Table 1: Results of evaluation experiments.

SL	Dice coefficient	Dice +ICL	Yates' χ^2	Yates +ICL	AIC	AIC +ICL	Number of bilingual word pairs
English	49.7%	58.0%	53.8%	59.8%	53.3%	58.6%	169
French	47.9%	56.7%	55.4%	60.4%	55.4%	60.4%	240
German	53.3%	61.0%	53.3%	58.5%	53.8%	59.0%	195
Sh.-Chinese	54.9%	62.9%	57.6%	62.5%	58.3%	62.9%	264
Ainu	54.0%	61.5%	52.1%	62.0%	52.6%	62.4%	213
Total	52.1%	60.1%	54.7%	60.8%	54.9%	60.9%	1,081

3.4 Method based on similarity measure

In the method based on similarity measures, the system extracts bilingual word pairs using only one similarity measure (*i.e.*, the Dice coefficient, Yates' χ^2, AIC) without using ICL when the similarity values are not greater than the threshold value or when no bilingual word pairs are extracted. Moreover, the system chooses the bilingual word pairs that appear in the parallel corpus at the first time when several candidates of bilingual word pairs are obtained.

4 Performance Evaluation

4.1 Experimental Procedure and Evaluation Standard

Five kinds of parallel corpora were used in this paper as experimental data. These parallel corpora are for English – Japanese, French – Japanese, German – Japanese, Shanghai-Chinese – Japanese and Ainu – Japanese. They were taken from textbooks(Harukawa and Snelling, 1998; Chikushi, 2001; Oshio, 2004; Emoto and Han, 2004; Nakagawa and Nakamoto, 2004). The number of bilingual sentence pairs was 1,794; the average numbers of words in SL and TL sentences were 6.8 and 8.8, respectively. We inputted all 1,081 SL words of nouns, verbs, adjectives, adverbs, and conjunctions in five parallel corpora to six systems: a system based on the Dice coefficient; a system based on the Dice coefficient in which AIL is applied (herein, we call it the system based on Dice+ICL); a system based on Yates' χ^2; a system based on Yates' χ^2 in which ICL is applied (herein, the system based on Yates+ICL); a system based on AIC; and a system based on AIC in which ICL is applied (herein, the system based on AIC+ICL). Initially, the dictionary for bilingual word pairs and the bilingual rule dictionary are empty. Moreover, the system uses 0.5 as its best threshold[4]. We repeated the experiments for each parallel corpus using respective systems.

We evaluated whether or not correct bilingual word pairs exist in the dictionary. Moreover, we calculated the recall. The recall is the rate for the number of correct bilingual word pairs to the number of all bilingual word pairs in the parallel corpora (*i.e.*, 1,081).

4.2 Experiments and Discussion

Table 1 shows the results of the experiments. The respective recall values of the systems based on Dice+ICL, Yates+ICL, and AIC+ICL were more than 8.0, 6.1, and 6.0 percentage points higher than those of the systems based on the Dice coefficient, Yates' χ^2, and AIC. These results indicate that ICL is effective for various similarity measures. Particularly, the recall values of the bilingual word pairs for which the frequencies are 1 improved to 11.0, 9.7 and 9.9 percentage points using ICL. In systems without ICL, many bilingual word pairs for which the frequencies are 1 were erroneous bilingual

[4] This value was obtained through preliminary experiments. Some correct bilingual word pairs are evaluated as erroneous bilingual word pairs when the system using ICL uses a high value as a threshold. In contrast, some erroneous bilingual word pairs are evaluated as correct bilingual word pairs when the system using ICL uses a low value as threshold. Therefore, 0.5, the middle value, became a most suitable threshold.

Table 2: Examples of bilingual word pairs extracted by ICL.

SL	Correct bilingual word pairs	Erroneous bilingual word pairs	
		Bilingual word pairs	Equivalents
English	(cereal; シリアル） 1.0 (boarding house; 下宿） 1.0	(curtains; 新しい [new]） 0.67 (interesting; 外 [outside]） 0.67	curtains interesting
French	(monuments; 記念/建造/物） 1.0 (cherche; 探し [search]） 0.67	(surtout; 関係 [relation]） 1.0 (petit; 所 [place]） 0.67	specially small
German	(nämlich; つまり [after all]） 0.67 (das Foto; 写真 [photograph]） 1.0	(Wege; 橋 [bridge]） 1.0 (Neues; 新聞 [newspaper]） 0.67	lane new event
Sh.-Chinese	(下班；退勤/し [leave office]） 1.0 (大閘蟹; 上海/ガニ [Shanghai crab]） 1.0	(中飯; ご馳走/し [treat]） 1.0 (夜飯; サービス [service]） 0.67	lunch dinner
Ainu	(ekupa; くわえ [take something in one's mouth]） 1.0 (set；寝床 [bed]） 1.0	(apto; 降っ[fall]） 1.0 (tunasno; 起き [get up]） 0.67	rain early

word pairs created by data sparseness problems, as described in section 1.1. Therefore, improvement of the recall values of bilingual word pairs for which the frequencies are 1 indicates that ICL is effective to solve the sparse data problem. On the other hand, the precision values – the rates of the number of correct bilingual word pairs to the number of all extracted bilingual word pairs – are all equal to the recall values. Among all 1,081 SL words, the correct bilingual word pairs or erroneous bilingual word pairs were obtained by the method based on similarity measure when the ICL process extracted no bilingual word pairs. Consequently, the numbers of all bilingual word pairs in the parallel corpora and of all extracted bilingual word pairs became 1,081. That is, the precision values are identical to the recall values. Table 2 shows examples of bilingual word pairs extracted by ICL and their similarity values. Table 2 indicates that ICL can extract not only bilingual word pairs that the number of words is 1, but also bilingual word pairs that the number of words is over 1.

Furthermore, we applied ICL to GIZA++. Table 3 shows those experimental results. The total recall of GIZA++ +ICL was more than 6.6 percentage points higher than that of GIZA++. Table 3 indicates that ICL is very effective for parallel corpora between languages for which the grammatical structure of SL differs from the grammar structure of TL. Grammatical structures of English, French, and German are SVO, whereas the Japanese grammatical structure is SOV. Using ICL, the recall improved 15.3 percentage points on average in English – Japanese, French – Japanese, and German – Japanese parallel corpora.

Table 3: Experimental results in GIZA++.

SL	GIZA++	GIZA++ +ICL
English	47.3%	54.4%
French	39.6%	54.2%
German	37.4%	61.5%
Sh.-Chinese	62.5%	60.6%
Ainu	66.6%	58.2%
Total	51.3%	57.9%

5 Conclusion

This paper presented **I**nductive **C**hain **L**earning (ICL) as a new learning method to solve the sparse data problem in extraction of bilingual word pairs among various languages. From experimental results, we confirmed that ICL is effective to solve the sparse data problem in extraction of bilingual word pairs from parallel corpora with various languages.

Future studies will solve the problem of word-ambiguity. Moreover, we apply our method to a multilingual machine translation system and an cross-language information retrieval system.

6 Acknowledgements

This work was partially supported by Grants from the High-Tech Research Center of Hokkai-Gakuen University, and an academic research grant of Hokkai-Gakuen University.

References

Manning, C. D. and Schütze, H. 1999. Foundations of Statistical Natural Language Processing. The MIT Press.

Sadat, F., Déjean, H. and Gaussier, É. 2002. A combination of models for bilingual lexicon extraction from comparable corpora. In *Proceedings of Papilion'02*, pp.16–21.

Smadja, F., McKeown, K. R. and Hatzivassiloglou, V. 1996. Translating Collocations for Bilingual Lexicons: A Statistical Approach. *Computational Linguistics*, vol.22, no.1, pp.1–38.

Echizen-ya, H., Araki, K. Momouchi, Y., and Tochinai, K. 2002. Study of Practical Effectiveness for Machine Translation Using Recursive Chain-link-type Learning. In *Proceedings of COLING '02*, pp.246–252.

Hisamitsu, T. and Niwa, Y. 2001. Topic-Word Selection Based on Combinatorial Probability. In *NLPRS'01*, pp.289–296.

Akaike, H. 1974. A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, AC-19(6), pp.716–723.

Och, F. J. 2003. GIZA++: Training of statistical translation models. Available at http://www-i6.informatik.rwth-aachen.de/Colleagues/och/software/GIZA++.html

Fung, P. 1995. Compiling bilingual lexicon entries from a non-parallel English-Chinese corpus. Workshop on very large corpora, pp.173–183.

Rapp, R. 1999. Automatic identification of word translations from unrelated English and German corpora. In *Proceedings of ACL'99*, pp.519–526.

Fung, P. 1998. A statistical view on bilingual lexicon extraction: from parallel corpora to non-parallel corpora, *LNAI*, Springer Publishing, vol.1529, pp.1–17.

Kaji, H. and Aizono, T. 1996. Extracting Word Correspondences from Bilingual Corpora Based on Word Co-occurrence Information. In *Proc. Coling'96*, pp.23–28.

Tanaka, K. and Iwasaki, H. 1996. Extraction of Lexical Translation from Non-Aligned Corpora. In *Proc. Coling'96*, pp.580–585.

McTait, K. 1997. Linguistic knowledge and complexity in an EBMT system based on translation patterns. In *Proceedings Workshop on EBMT, MT Summit VIII.*

Güvenir, H. A. and Cicekli, I. 1998. Learning translation templates from examples. *Information Systems*, vol. 23, no.6, pp.353–363.

Fung, P. and Church, K. 1994. K-vec: A new approach for alignment parallel texts. In *Proc. Coling'94*, pp.1096–1102.

Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., and Mercer, R. L. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, vol.19, no.2, pp.263–311.

Melamed, I. D. 2000. Models of translation equivalence among words. *Computational Linguistics*, vol.26, no.2, pp.221–249.

Och, F. J. and Ney, H. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, vol.29, no.1, pp.19–51.

Nie*ß*en, S. and Ney, H. 2004. Statistical machine translation with scarce resources using morpho-syntactic information. *Computational Linguistics*, vol.30, no.2, pp.181–204.

Watanabe, Y. and Sumita, E. 2003. Example-based decoding for statistical machine translation. In *Proceedings of MT summit IX*, pp.410–417.

Harukawa, Y. and Snelling, J. 1998. Express: English. Hakusui-sha (in Japanese).

Chikushi, F. 2001. Express: French. Hakusui-sha (in Japanese).

Oshio, T. 2004. Express: German. Hakusui-sha (in Japanese).

Emoto, H. and Han, G. 2004. Express: Shanghai. Hakusui-sha (in Japanese).

Nakagawa, H. and Nakamoto, M. 2004. Express: Ainu. Hakusui-sha (in Japanese).

Approximate Searching for Distributional Similarity

James Gorman and **James R. Curran**
School of Information Technologies
University of Sydney
NSW 2006, Australia
{jgorman2,james}@it.usyd.edu.au

Abstract

Distributional similarity requires large volumes of data to accurately represent infrequent words. However, the nearest-neighbour approach to finding synonyms suffers from poor scalability. The Spatial Approximation Sample Hierarchy (SASH), proposed by Houle (2003b), is a data structure for approximate nearest-neighbour queries that balances the efficiency/approximation trade-off. We have intergrated this into an existing distributional similarity system, tripling efficiency with a minor accuracy penalty.

1 Introduction

With the development of WordNet (Fellbaum, 1998) and large electronic thesauri, information from lexical semantic resources is regularly used to solve NLP problems. These problems include collocation discovery (Pearce, 2001), smoothing and estimation (Brown et al., 1992; Clark and Weir, 2001) and question answering (Pasca and Harabagiu, 2001).

Unfortunately, these resources are expensive and time-consuming to create manually, and tend to suffer from problems of bias, inconsistency, and limited coverage. In addition, lexicographers cannot keep up with constantly evolving language use and cannot afford to build new resources for the many subdomains that NLP techniques are being applied to. There is a clear need for methods to extract lexical semantic resources automatically or tools that assist in their manual creation and maintenance.

Much of the existing work on automatically extracting resources is based on the *distributional hypothesis* that *similar words appear in similar contexts*. Existing approaches differ primarily in their definition of "context", e.g. the surrounding words or the entire document, and their choice of distance metric for calculating similarity between the vector of contexts representing each term. Finding synonyms using distributional similarity involves performing a nearest-neighbour search over the context vectors for each term. This is very computationally intensive and scales according to the vocabulary size and the number of contexts for each term. Curran and Moens (2002b) have demonstrated that dramatically increasing the quantity of text used to extract contexts significantly improves synonym quality. Unfortunately, this also increases the vocabulary size and the number of contexts for each term, making the use of huge datasets infeasible.

There have been many data structures and approximation algorithms proposed to reduce the computational complexity of nearest-neighbour search (Chávez et al., 2001). Many of these approaches reduce the search space by using clustering techniques to generate an index of near-neighbours. We use the Spacial Approximation Sample Hierarchy (SASH) data structure developed by Houle (2003b) as it allows more control over the efficiency-approximation trade-off than other approximation methods.

This paper describes integrating the SASH into an existing distributional similarity system (Curran, 2004). We show that replacing the nearest-neighbour search improves efficiency by a factor of three with only a minor accuracy penalty.

Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition, pages 97–104,
Ann Arbor, June 2005. ©2005 Association for Computational Linguistics

2 Distributional Similarity

Distributional similarity systems can be separated into two components. The first component extracts the contexts from raw text and compiles them into a statistical description of the contexts each term appears in. The second component performs nearest-neighbour search or clustering to determine which terms are similar, based on distance calculations between their context vectors. The approach used in this paper follows Curran (2004).

2.1 Extraction Method

A *context relation* is defined as a tuple (w, r, w') where w is a term, which occurs in some grammatical relation r with another word w' in some sentence. We refer to the tuple (r, w') as an *attribute* of w. For example, (dog, diect-obj, walk) indicates that dog was the direct object of walk in a sentence.

Context extraction begins with a Maximum Entropy POS tagger and chunker (Ratnaparkhi, 1996). The Grefenstette (1994) relation extractor produces context relations that are then lemmatised using the Minnen et al. (2000) morphological analyser. The relations for each term are collected together and counted, producing a context vector of attributes and their frequencies in the corpus.

2.2 Measures and Weights

Both nearest-neighbour and cluster analysis methods require a distance measure that calculates the similarity between context vectors. Curran (2004) decomposes this measure into *measure* and *weight* functions. The *measure* function calculates the similarity between two weighted context vectors and the *weight* function calculates a weight from the raw frequency information for each context relation.

The SASH requires a distance measure that preserves metric space (see Section 4.1). For these experiments we use the JACCARD (1) measure and the TTEST (2) weight, as Curran and Moens (2002a) found them to have the best performance in their comparison of many distance measures.

$$\frac{\sum_{(r,w')} \min(\mathrm{wgt}(w_m, *_r, *_{w'}), \mathrm{wgt}(w_n, *_r, *_{w'}))}{\sum_{(r,w')} \max(\mathrm{wgt}(w_m, *_r, *_{w'}), \mathrm{wgt}(w_n, *_r, *_{w'}))} \quad (1)$$

$$\frac{p(w, r, w') - p(*, r, w')p(w, *, *)}{\sqrt{p(*, r, w')p(w, *, *)}} \quad (2)$$

3 Nearest-neighbour search

The simplest algorithm for finding synonyms is nearest-neighbour search, which involves pairwise vector comparison of the target term with every term in the vocabulary. Given an n term vocabulary and up to m attributes for each term, the asymptotic time complexity of nearest-neighbour search is $O(n^2m)$. This is very expensive with even a moderate vocabulary and small attribute vectors making the use of huge datasets infeasible.

3.1 Heuristic

Using cutoff to remove low frequency terms can significantly reduce the value of n. In these experiments, we used a cutoff of 5. However, a solution is still needed to reduce the factor m. Unfortunately, reducing m by eliminating low frequency contexts has a significant impact on the quality of the results.

Curran and Moens (2002a) propose an initial heuristic comparison to reduce the number of full $O(m)$ vector comparisons. They introduce a bounded vector (length k) of *canonical* attributes, selected from the full vector, to represent the term. The selected attributes are the most strongly weighted verb attributes: Curran and Moens chose these relations as they generally constrain the semantics of the term more and partake in fewer idiomatic collocations.

If a pair of terms share at least one canonical attribute then a full similarity comparison is performed, otherwise the terms are not considered similar. If a maximum of p positive results are returned, our complexity becomes $O(n^2k+npm)$, which, since k is constant, is $O(n^2 + npm)$.

4 The SASH

The SASH approximates a nearest-neighbour search by pre-computing some of the near-neighbours of each node (terms in our case). It is arranged as a multi-leveled pyramid, where each node is linked to its (approximate) near-neighbours on the levels above and below. This produces multiple paths between nodes, allowing the SASH to shape itself to the data set (Houle, 2003a). This graph is searched by finding the near-neighbours of the target node at each level. The following description is adapted from Houle (2003b).

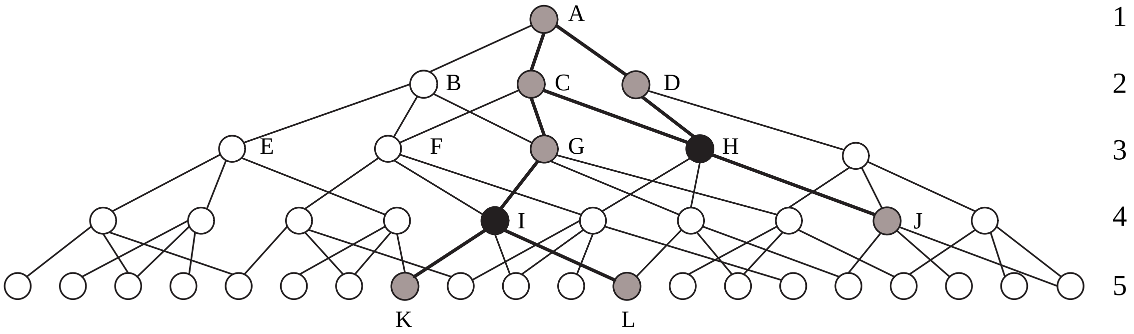


Figure 1: A SASH, where $p = 2$, $c = 3$ and $k = 2$

4.1 Metric Spaces

The SASH organises nodes that can be measured in *metric space*. Although it is not necessary for the SASH to work, only in this space can performance be guaranteed. Our meaures produce a *metric-like* space for the terms derived from large datasets.

A domain D is a *metric space* if there exists a function $dist : D \times D \rightarrow R^{\geq 0}$ such that:

1. $dist(p, q) \geq 0 \; \forall \; p, q \in D$ (*non-negativity*)
2. $dist(p, q) = 0$ iff $p = q \; \forall \; p, q \in D$ (*equality*)
3. $dist(p, q) = dist(q, p) \; \forall \; p, q \in D$ (*symmetry*)
4. $dist(p, q) + dist(q, r) \geq dist(p, r)$ $\forall \; p, q, r \in D$ (*triangle inequality*)

We invert the similarity measure to produce a distance, resulting in condition 2 not being satisfied since $dist(p, p) = x$, $x > 0$. For most measures x is constant, so $dist(p, q) > dist(p, p)$ if $p \neq q$ and p and q do not occur in exactly the same contexts. For some measures, e.g. DICE, $dist(p, p) > dist(p, q)$, that is, p is closer to q than it is to itself. These do not preserve metric space in any way, so cannot be used with the SASH.

Chávez et al. (2001) divides condition 2 into:

5. $dist(p, p) = 0 \; \forall \; p \in D$ (*reflexivity*)
6. $dist(p, q) > 0$ iff $p \neq q \; \forall \; p, q \in D$ (*strict positiveness*)

If strict positiveness is not satisfied the space is called *pseudometric*. In theory, our measures do not satisfy this condition, however in practice most large datasets will satisfy this condition.

4.2 Structure

The SASH is a directed, edge-weighted graph with the following properties:

- Each term corresponds to a unique node.
- The nodes are arranged into a hierarchy of levels, with the bottom level containing $\frac{n}{2}$ nodes and the top containing a single root node. Each level, except the top, will contain half as many nodes as the level below. These are numbered from 1 (top) to h.
- Edges between nodes are linked from consecutive levels. Each node will have at most p *parent* nodes in the level above, and c *child* nodes in the level below.
- Every node must have at least one parent so that all nodes are reachable from the root.

Figure 1 shows a SASH which will be used below.

4.3 Construction

The SASH is constructed iteratively by finding the nearest parents in the level above. The nodes are first randomly distributed to reduce any clustering effects. They are then split into the levels described above, with level h having $\frac{n}{2}$ nodes, level 2 at most c nodes and level 1 having a single root node.

The root node has all nodes at level 2 as children and each node at level 2 has the root as its sole parent. Then for each node in each level i from 3 to h, we find the set of p nearest parent nodes in level $(i - 1)$. The node then asks that parent if it can be a child. As only the closest c nodes can be children of a node, it may be the case that a requested parent rejects a child.

Dist	c	Load Time
RANDOM	16	21.0hr
RANDOM	64	15.6hr
RANDOM	128	21.1hr
FOLD1500	16	50.2hr
FOLD1500	64	33.4hr
FOLD1500	128	25.7hr
SORT	16	75.5hr
SORT	64	23.8hr
SORT	128	33.8hr

Table 1: Load time distributions and values of c

If a child is left without any parents it is said to be *orphaned*. Any orphaned nodes must now find the closest node in the above level that has fewer than c children. Once all nodes have at least one parent, we move to the next level. This proceeds iteratively through the levels.

4.4 Search

Searching the SASH is also performed iteratively. To find the k nearest neighbours of a node q, we first find the k nearest neighbours at each level. At level 1 we take the single root node to be nearest. Then, for each level after, we find the k nearest unique children of the nodes found in the level above. When the last level has been searched, we return the closest k nodes from all the sets of near neighbours returned.

In Figure 1, the filled nodes demonstrate a search for the near-neighbours of some node q, using $k = 2$. Our search begins with the root node A. As we are using $k = 2$, we must find the two nearest children of A using our similarity measure. In this case, C and D are closer than B. We now find the closest two children of C and D. E is not checked as it is only a child of B. All other nodes are checked, including F and G, which are shared as children by B and C. From this level we chose G and H. We then consider the fourth and fifth levels similarly.

At this point we now have the list of near nodes A, C, D, G, H, I, J, K and L. From this we chose the two nodes closest to q: H and I marked in black. These are returned as the near-neighbours of q.

k can also be varied at each level to force a larger number of elements to be tested at the base of the SASH using, for instance, the equation:

$$k_i = \max\{\, k^{1-\frac{h-i}{\log_2 n}}, \frac{1}{2}pc \,\} \qquad (3)$$

We use this geometric function in our experiments.

4.5 Complexity

When measuring the time complexity, we consider the number of distance measurements as these dominate the computation. If we do not consider the problem of assigning parents to orphans, for n nodes, p parents per child, at most c children per parent and a search returning k elements, the loose upper bounds are:

SASH construction

$$pcn \log_2 n \qquad (4)$$

Approx. k-NN query (uniform)

$$ck \log_2 n \qquad (5)$$

Approx. k-NN query (geometric)

$$\frac{k^{1+\frac{1}{\log_2 n}}}{k^{\frac{1}{\log_2 n}}-1} + \frac{pc^2}{2}\log_2 n \qquad (6)$$

Since the average number of children per node is approximately $2p$, practical complexities can be derived using $c = 2p$.

In Houle's experiments, typically less than 5% of computation time was spent assigning parents to orphans, even for relatively small c. In some of our experiments we found that low values of c produced significantly worse load times that for higher values, but this was highly dependant on the distribution of nodes. Table 1 shows this with respect to several distributions and values of c.

5 Evaluation

The simplest method of evaluation is direct comparison of the extracted synonyms with a manually-created gold standard (Grefenstette, 1994). However, on small corpora, rare direct matches provide limited information for evaluation, and thesaurus coverage is a problem. Our evaluation uses a combination of three electronic thesauri: the Macquarie (Bernard, 1990), Roget's (Roget, 1911) and Moby (Ward, 1996) thesauri.

With this gold standard in place, it is possible to use precision and recall measures to evaluate the quality of the extracted thesaurus. To help overcome the problems of direct comparisons we use several measures of system performance: direct matches (DIRECT), inverse rank (INVR), and precision of the top n synonyms (P(n)), for n = 1, 5 and 10.

INVR is the sum of the inverse rank of each matching synonym, e.g. matching synonyms at ranks 3, 5 and 28 give an inverse rank score of $\frac{1}{3} + \frac{1}{5} + \frac{1}{28}$, and with at most 100 synonyms, the maximum INVR score is 5.187. Precision of the top n is the percentage of matching synonyms in the top n extracted synonyms.

The same 70 single-word nouns were used for the evaluation as in Curran and Moens (2002a). These were chosen randomly from WordNet such that they covered a range over the following properties:

frequency Penn Treebank and BNC frequencies;

number of senses WordNet and Macquarie senses;

specificity depth in the WordNet hierarchy;

concreteness distribution across WordNet subtrees.

For each of these terms, the closest 100 terms and their similarity score were extracted.

6 Experiments

The contexts were extracted from the non-speech portion of the British National Corpus (Burnard, 1995). All experiments used the JACCARD measure function, the TTEST weight function and a cutoff frequency of 5. The SASH was constructed using the geometric equation for k_i described in Section 4.4. When the heuristic was applied, the TTESTLOG weight function was used with a canonical set size of 100 and a maximum frequency cutoff of 10,000.

The values 4–16, 8–32, 16–64, and 32–128 were chosen for p and c. This gives a range of branching factors to test the balance between *sparseness*, where there is potential for erroneous fragmentation of large clusters, and *bushiness*, where more tests must be made to find near children. The $c = 4p$ relationship is derived from the simple hashing rule of thumb that says that a hash table should have roughly twice the size required to store all its elements (Houle, 2003b).

DIST	FREQUENCY		# RELATIONS	
	Mean	Median	Mean	Median
RANDOM	342	18	126	13
FOLD500	915	865.5	500	500
FOLD1000	2155	1970.5	1001	1001.5
FOLD1500	3656	3444	1506	1510.5
SORT	44753	37937.5	8290	7583.5

Table 2: Top 3 SASH level averages with c = 128

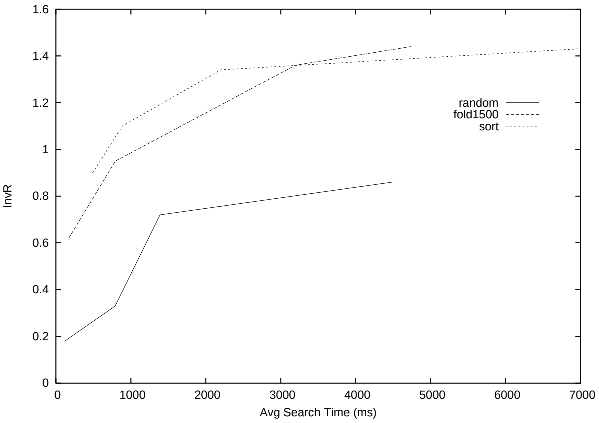


Figure 2: INVR against average search time

Our initial experiments showed that the random distribution of nodes (RANDOM) in SASH construction caused the nearest-neighbour approximation to be very inaccurate for distributional similarity. Although the speed was improved by two orders of magnitude when c = 16, it achieved only 13% of the INVR of the naïve implementation. The best RANDOM result was less than three times faster then the naïve solution and only 60% INVR.

In accordance with Zipf's law the majority of terms have very low frequencies. Similarity measurements made against these low frequency terms are less reliable, as accuracy increases with the number of relations and their frequencies (Curran and Moens, 2002b). This led to the idea that ordering the nodes by frequency before generating the SASH would improve accuracy.

The SASH was then generated with the highest frequency terms were near the root so that the initial search paths would be more accurate. This has the unfortunate side-effect of slowing search by up to four times because comparisons with high frequency terms take longer than with low frequency terms as they have a larger number of relations.

DIST	c	DIRECT	P(1)	P(5)	P(10)	INVR		SEARCH TIME
NAIVE		2.83	49%	41%	32%	1.43		12217ms
RANDOM	16	0.17	9%	6%	3%	0.18	13%	120ms
RANDOM	64	1.09	30%	21%	15%	0.72	50%	1388ms
RANDOM	128	1.53	31%	24%	20%	0.86	60%	4488ms
SORT	16	1.51	33%	25%	20%	0.90	63%	490ms
SORT	64	2.55	47%	38%	31%	1.34	94%	2197ms
SORT	128	2.81	49%	41%	33%	1.43	100%	6960ms

Table 3: Evaluation of different random and fully sorted distributions

This led to updating our original frequency ordering idea by recognising that we did not need the *most* accurately comparable terms at the top of the SASH, only *more* accurately comparable terms than those randomly selected.

As a first attempt, we constructed SASHs with frequency orderings that were *folded* about a chosen number of relations $\mathcal{M}$. For each term, if its number of relations m_i was greater than $\mathcal{M}$, it was given a new ranking based on the score $\frac{\mathcal{M}^2}{m_i}$. In this way, very high and very low frequency terms were pushed away from the root. The folding points this was tested for were 500, 1000 and 1500. There are many other node organising schemes we are yet to explore.

The frequency distributions over the top three levels for each ordering scheme are shown in Table 2. Zipf's law results in a large difference between the mean and median frequency values in the RANDOM results: most of the nodes have low frequency, but some high frequency results push the mean up. The four-fold reduction in efficiency for SORT (see Table 3) is a result of the mean number of relations being over 65 times that of RANDOM.

Experiments covering the full set of permutations of these parameters were run, with and without the heuristic applied. In the cases where the heuristic rejected pairs of terms, the SASH treated the rejected pairs as being as infinitely far apart. In addition, the brute force solutions were generated with (NAIVE HEURISTIC) and without (NAIVE) the heuristic.

We have assumed that all weights and measures introduce similar distribution properties into the SASH, so that the best weight and measure when performing a brute-force search will also produce the best results when combined with the SASH. Future experiments will explore SASH behaviour with other similarity measures.

7 Results

Table 3 presents the results for the initial experiments. SORT was consistently more accurate than RANDOM, and when $c = 128$, performed as well as NAIVE for all evaluation measures except for direct matches. Both SASH solutions outperformed NAIVE in efficiency.

The trade-off between efficiency and approximation accuracy is evident in these results. The most efficient result is 100 times faster than NAIVE, but only 13% accurate on INVR, with 6% of direct matches. The most accurate result is 100% accurate on INVR, with 99% of direct matches, but is less than twice as fast.

Table 4 shows the trade-off for folded distributions. The least accurate FOLD500 result is 30% accurate but 50 times faster than NAIVE, while the most accurate is 87% but less than two times faster. The least accurate FOLD1500 result is 43% accurate but 71 times faster than NAIVE, while the most accurate is 101% and two and half times faster. These results show the impact of moving high frequency terms away from the root.

Figure 2 plots the trade-off using search time and INVR at $c = 16$, 32, 64 and 128. For $c = 16$ every SASH has very poor accuracy. By $c = 64$ their accuracy has improved dramatically, but their search time also increased somewhat. At $c = 128$, there is only a small improvement in accuracy, coinciding with a large increase in search time. The best trade-off between efficiency and approximation accuracy occurs at the knee of the curve where $c = 64$.

When $c = 128$ both SORT and FOLD1500 perform as well as, or slightly outperform NAIVE on some evaluation measures. These evaluation measures involve the rank of correct synonyms, so if the SASH

Dist	*c*	Direct	P(1)	P(5)	P(10)	InvR		Search Time
fold500	16	0.53	23%	11%	8%	0.43	30%	243ms
fold500	64	1.69	49%	29%	23%	1.09	76%	2880ms
fold500	128	2.29	50%	35%	27%	1.25	87%	6848ms
fold1000	16	0.61	29%	14%	9%	0.51	35%	228ms
fold1000	64	2.07	49%	36%	26%	1.21	84%	3192ms
fold1000	128	2.57	50%	39%	31%	1.40	98%	4330ms
fold1500	16	0.90	30%	17%	13%	0.62	43%	171ms
fold1500	64	2.36	57%	39%	30%	1.36	95%	3193ms
fold1500	**128**	**2.67**	**53%**	**42%**	**32%**	**1.44**	**101%**	**4739ms**

Table 4: Evaluation of folded distributions

approximation was to fail to find some incorrectly proposed synonyms ranked above some other correct synonyms, those correct synonyms would have their ranking pushed up. In this way, the approximation can potentially outperform the original nearest-neighbour algorithm.

From Tables 3 and 4 we also see that as the value of *c* increases, so does the accuracy across all of the experiments. This is because as *c* increases the number of paths between nodes increases and we have a solution closer to a true nearest-neighbour search, that is, there are more ways of finding the true nearest-neighbour nodes.

Table 5 presents the results of combining the canonical attributes heuristic (see Section 3.1) with the SASH approximation. This NAIVE HEURISTIC is 14 times faster than NAIVE and 97% accurate, with 96% of direct matches. The combination has comparable accuracy and is much more efficient than the best of the SASH solutions. The best heuristic SASH results used the SORT ordering with $c = 16$, which was 37 times faster than NAIVE and 2.5 times faster than NAIVE HEURISTIC. Its performance was statistically indistinguishable from NAIVE HEURISTIC.

Using the heuristic changes the impact of the number of children *c* on the SASH performance characteristics. It seems that beyond $c = 16$ the only significant effect is to *reduce* the efficiency (often to slower than NAIVE HEURISTIC).

The heuristic interacts in an interesting way with the ordering of the nodes in the SASH. This is most obvious with the RANDOM results. The RANDOM heuristic InvR results are eight times better than the full RANDOM results. Similar, though less dramatic, results are seen with other orderings. It appears that using the heuristic changes the clustering of nearest-neighbours within the SASH so that better matching paths are chosen and more noisy matches are eliminated entirely by the heuristic.

It may seem that there are no major advantages to using the SASH with the already efficient heuristic matching method. However, our experiments have used small canonical attribute vectors (maximum length 100). Increasing the canonical vector size allows us to increase the accuracy of heuristic solutions at the cost of efficiency. Using a SASH solution would offset some of this efficiency penalty. This has the potential for a solution that is more than an order of magnitude faster than NAIVE and is almost as accurate.

8 Conclusion

We have integrated a nearest-neighbour approximation data structure, the Spacial Approximation Sample Hierarchy (SASH), with a state-of-the-art distributional similarity system. In the process we have extended the original SASH construction algorithms (Houle, 2003b) to deal with the non-uniform distribution of words within semantic space.

We intend to test other similarity measures and node ordering strategies, including a more linguistic analysis using WordNet, and further explore the interaction between the canonical vector heuristic and the SASH. The larger 300 word evaluation set used by Curran (2004) will be used, and combined with a more detailed analyis. Finally, we plan to optimise our SASH implementation so that it is comparable with the highly optimised nearest-neighbour code.

DIST	*c*	DIRECT	P(1)	P(5)	P(10)	INVR		SEARCH TIME
NAIVE HEURISTIC		2.72	49%	40%	32%	1.40		827ms
RANDOM	16	2.61	50%	40%	31%	1.39	99%	388ms
RANDOM	64	2.72	49%	40%	32%	1.40	100%	1254ms
RANDOM	128	2.71	49%	40%	32%	1.40	100%	1231ms
FOLD1500	16	2.53	49%	40%	31%	1.36	97%	363ms
FOLD1500	64	2.72	49%	40%	32%	1.40	100%	900ms
FOLD1500	128	2.72	49%	40%	32%	1.40	100%	974ms
SORT	**16**	**2.78**	**49%**	**40%**	**32%**	**1.41**	**100%**	**323ms**
SORT	64	2.73	49%	40%	32%	1.40	100%	1238ms
SORT	128	2.73	49%	40%	32%	1.40	100%	1049ms

Table 5: Evaluation of different distributions using the approximation

The result is distributional similarity calculated three times faster than existing systems with only a minor accuracy penalty.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful feedback and corrections. This work has been supported by the Australian Research Council under Discovery Project DP0453131.

References

John R. L. Bernard, editor. 1990. *The Macquarie Encyclopedic Thesaurus*. The Macquarie Library, Sydney, Australia.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jennifer C. Lai, and Robert L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479, December.

Lou Burnard, editor. 1995. *Users Reference Guide British National Corpus Version 1.0*. Oxford University Computing Services.

Edgar Chávez, Gonzalo Navarro, Ricardo Baeza-Yates, and José L. Marroquín. 2001. Searching in metric spaces. *ACM Computing Surveys*, 33(3):273–321, September.

Stephen Clark and David Weir. 2001. Class-based probability estimation using a semantic hierarchy. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 95–102, Pittsburgh, PA USA, 2–7 June.

James R. Curran and Marc Moens. 2002a. Improvements in automatic thesaurus extraction. In *Proceedings of the Workshop of the ACL Special Interest Group on the Lexicon (SIGLEX)*, pages 59–66, Philadelphia, USA, 12 July.

James R. Curran and Marc Moens. 2002b. Scaling context space. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 231–238, Philadelphia, USA, 7–12 July.

James R. Curran. 2004. *From Distributional to Semantic Similarity*. Ph.D. thesis, University of Edinburgh.

Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. The MIT Press, Cambridge, MA USA.

Gregory Grefenstette. 1994. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, Boston, USA.

Michael E. Houle. 2003a. Navigating massive data sets via local clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 547–552, Washington, DC, USA, 24–27 August.

Michael E. Houle. 2003b. SASH: a saptial approximation sample hierarchy for similarity search. Technical Report RT0517, IBM Reasearch, Tokyo Research Laboratory, Yamato Kanagawa, Japan, March.

Guido Minnen, John Carroll, and Darren Pearce. 2000. Robust applied morphological generation. In *Proceedings of the First International Natural Language Generation Conference*, pages 201–208, Mitzpe Ramon, Israel, 12–16 June.

Marius Pasca and Sanda Harabagiu. 2001. The informative role of wordnet in open-domain question answering. In *Proceedings of the Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 138–143, Pittsburgh, PA USA, 2–7 June.

Darren Pearce. 2001. Synonymy in collocation extraction. In *Proceedings of the Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 41–46, Pittsburgh, PA USA, 2–7 June.

Adwait Ratnaparkhi. 1996. A maximum entropy part-of-speech tagger. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 133–142, 17–18 May.

Peter Roget. 1911. *Thesaurus of English words and phrases*. Longmans, Green and Co., London, UK.

Grady Ward. 1996. *Moby Thesaurus*. Moby Project.

Author Index