Annotating Semantic Structure of Web Text*

Abstract
Facing the challenges of annotating naturally occurring text into semantic structured form, current semantic role labeling (SRL) systems have been focusing on semantic predicate-argument structure. Based on a new Concept Description Language for Natural Language (CDL.nl) which aims to describe the entire concept structure of Web text using a set of semantic relations, we develop a new shallow semantic parser as an extension of SRL. With the assumption that all entity pairs between each there exits a relation have been recognized, we present a relation classification approach displaying two advantages over previous ones: (1) besides common used lexical information, we use a new lexical resource to provide semantic behavior features of words, and (2) in order to leverage weights among diverse features to improve the performance, we use kernel functions to model lexical features separately from syntactic and dependency features and combine them into a composition kernel. Preliminary evaluation on our manual dataset, using Support Vector Machine, shows that CDL.nl relations can be classified with good performance and both advantages of our method can improve the overall performance.

Introduction
Most of the information on the Internet is unstructured, generated in textual form. Automatically annotating naturally occurring text into structured format using NLP techniques can play a key role in Web applications such as Web Searching and Information Extraction by taking into account the meaning of the text. Recently, a lot of attention has been devoted to Semantic Role Labeling (SRL) of natural language text, as called shallow semantic parsing, which is becoming an important component in many kinds of NLP applications (Narayanan et al., 2004; Harabagiu et al., 2005). SRL is currently a well-defined task with a substantial body of work and comparative evaluation (Litkowski, 2004; Carreras et al. 2005). Within the task of semantic role labeling, high-performance systems have been developed using FrameNet (Baker et al., 1998) and PropBank (Palmer et al., 2005) corpora as training and testing material. However, FrameNet suffers from the limitation that roles are frame-specific, and only predicates from certain predetermined semantic frames can be annotated. Propbank has the limitation that it covers only verb predicate-argument roles. And both of PropBank and FrameNet corpora cover only predicate-argument roles, they don’t contain any other types of relationship between entities, such as conjunction, equivalent relation.

So, the challenges that lie ahead of semantic annotation of text are still those the field has faced: demanding a set of semantic relations which are competent to represent the semantic structure of text and corresponding effective relation recognition method.

Yokoi et al. (2005) presented a descriptive language named CDL.nl (Concept Description Language for Natural Language) which is part of the realization of spirits of the work “semantic information processing” (Minsky, 1968) for the annotating of Web text in plain form into semantic structure. The semantic relation set defined as the core element to form the structure are not frame-specific as FrameNet, and have better coverage than Propbank. They record semantic relations showing how each meaningful entity (can be Noun, Verb, Adjective, Adverb) semantically depends on another entity. It connects all meaningful entities into a united graph representation, not only predicate-argument related entities.

In this paper, based on CDL.nl relation set, with the assumption that positions of each entity pair have been identified, we describe an algorithm for relation classification with two advantages over previous ones: besides common used lexical information, we use a new lexical resource to provide semantic behavior features of words; and use kernel method to model all lexical features separately from syntactic and dependency features in order to leverage among them to improve the performance. Preliminary experiments are trained on a hand-annotated dataset and show that CDL.nl relations can be classified with good performance.

Our contributions can be summarized as follows:

- We develop a new shallow semantic parser over SRL to annotate the entire semantic structure of text using a new set of semantic relations. By annotating text with deeper and wider semantic structure, it can expand the extent to which shallow semantic information can be used in real NLP and Web applications such as Information Extraction and Text Summarization.
- We experiment to use a new lexicon which encodes knowledge of word semantic behaviors based on CDL.nl relation set for relation classification. It suggests more an effective way to collect semantic information.

Copyright © 2008, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

*This first version of this work has been submitted to ACL-08, and we improve the experiments and extend our work with more details in this paper.
By modeling and leveraging lexical information separately from syntactic and dependency knowledge, our study also suggests an example of the flexibility of using kernel method to leverage diverse knowledge.

The rest of this paper is organized as follows. Section 2 gives a comparison on relation/role sets of FrameNet, Propbank and CDL.nl. Section 3 reviews the related work on semantic relation extraction and Section 4 proposes our method of relation classification. Section 5 reports the experimental results and our observations. We conclude our work and indicate the future work in Section 6.

Semantic Relations/ Roles Comparison

Facing the first challenge of semantic text annotation, this section gives a comparison between semantic role sets of FrameNet & PropBank and CDL.nl semantic relation set.

FrameNet Semantic Roles

FrameNet is based on frame semantics and designed as an ontology of frames. Frame is a schematic representation of situations involving various participants, props, and other conceptual roles. Each frame has a set of predicates (nouns, verbs or adjectives), which introduce the frame. For each semantic frame, it defines a set of semantic roles called frame elements which are shared by all predicates of the frame.

Semantic role labeling process: When using FrameNet, the role labeling process goes through (1) identifies the target words; (2) disambiguates the frame for each target word; and (3) labels the roles of arguments that relate to the target word based on the frame definition.

Bill Gates is an American entrepreneur and the chairman of Microsoft, [created entity the software company] [created he founded] [co-participant with Paul Allen] [place in Albuquerque, New Mexico] [time on April 4, 1975].

Above is an example showing how to annotate a sentence using FrameNet roles. We can see that it annotates only predicate-argument roles and only for predicates “chairman” and “found”, not for “entrepreneur” which is not encoded in any frame. Since FrameNet lists only 10197 lexical units, comparing to 207016 word-sense pairs in WordNet 3.0, it also has the limitation that the roles are frame-specific, and only predicates from certain predetermined semantic frames can be annotated.

Propbank Semantic Roles

The Proposition Bank (PropBank) (Palmer et al., 2005) annotates the Penn TreeBank with verb argument structure. It does not have the concept of frame and predicates are not organized into a particular structure. The semantic roles covered by PropBank are the following:

- Numbered arguments (A0–A5, A4): Arguments defining verb-specific roles. The most frequent roles are AO and AI and, commonly, A0 stands for the agent and A1 corresponds to the patient or theme of the predicate. Their semantics depend on the verb sense and verb usage in a sentence.
- Adjuncts (AM–): General arguments that any verb may take optionally. There are 13 types of adjuncts such as AM-ADV (general-purpose), AM-TMP (temporal).

Semantic role labeling process: When using PropBank, the role labeling process goes through (1) identifies each verbal predicate and (2) labels its arguments.

Bill Gates is an American entrepreneur and the chairman of Microsoft, [arg the software company] [arg he] [in founded] [with Paul Allen] [in Albuquerque, New Mexico] [on April 4, 1975].

Above is an example showing how to annotate a sentence with PropBank roles. We can see that PropBank suffers the limitation that the coverage of semantic roles is limited to verb predicate-argument roles.

CDL.nl Semantic Relations

Yokoi et al. (2005) presented CDL.nl (Concept Description Language for Natural Language) which is used to describe the semantic/concept structure of text as a core member of W3C Common Web Language1. Different from existed dependency parsers which represent grammatical dependency structure of text, it is used to describe semantic dependency structure of plain text in graph form. The two basic elements for describing the structure are Entity and Relation, where the concept Entity is used to represent constitutes of sentences with head word. A set of relations2 are defined to represent the meaning of the relationships between pairs of entities. The entity which activates a relation is called entry entity and the other one is called participant entity. CDL.nl also contains a lexicon named UNKLB which is used to organize entities according to their semantic behaviors which is based on their participated relations.

Different from roles in PropBank, where role semantics depends on the verb and verb usage in a sentence, or verb sense, the semantics of CDL.nl semantic relations are predefined. And additional information for distinguishing from similar relations is also described. For example, the definition of a0j(nominal entity with attribute) contains two parts: Definition: a0j indicates a nominal thing that is in a state or has an attribute.

Differences between related relations: A thing with an attribute is different from mod in that mod gives some restriction of the concept in focus, while a0j indicates a thing of a state or characteristic.

CDL.nl relation set contain all 44 relation types which are organized into three groups:

- Intra-event relation: Relations defining case roles, which are divided into 6 abstract relations, QuasiAgent, QuasiObject, QuasiInstrument, QuasiPlace, QuasiState and QuasiTime. And each abstract relation contains several concrete relations which express concrete semantic information. Such as QuasiAgent contains five semantic relations, agent, a0j(agent), a0j(thing with attribute), a0j(co-agent), coa(co-thing with attribute), ptm(partner).
- Inter-entity relations: relations which are meaningful between entities, not limited on certain entity type. Both of the entities are treated relatively equally. There are 13 inter-entity relations, such as and(conjunction), seq(sequence), eq(equivalent).
- Qualification relations: relations representing qualification relationships between modified entity and modi-

---
1http://www.w3.org/2005/incubator/cwl/
2http://www.miv.t.u-tokyo.ac.jp/mem/yyan/CDLnl/
There exist inter-entity relations and (3) labels the relations way as FrameNet; (2) detects entity pairs between which there exist inter-entity relations and (3) labels the relationship of each entity pair from (1)&(2) to form the entire structure of the sentence.

All these representations - FrameNet, Propbank and CDL.nl relation set - have their strengths. FrameNet focuses on how to describe word with its arguments in a related common scenario. Propbank describes how a verb relates to its arguments. CDL.nl describes how each meaningful entity is related to another entity and what the meaning of their relationship is. And we conclude that CDL.nl relation set has the following advantages:

- each relation in the set is pre-defined along with distinctive information from similar relations;
- it describes not only predicate-argument relations, but also those between each pair of entities there exists a meaningful relationship to form the entire structure of text. So, it has better coverage.

So, to this respect, facing the first challenge, we can say that CDL.nl relation set are more competent than FrameNet and Propbank role sets to represent the semantic structure of text. And annotating structure of text based on CDL.nl can be treated as deeper semantic parsing than semantic role labeling, although not deep enough to express all semantics of text, it is still a kind of shallow semantic parsing.

Related Work on Relation Extraction

Based on the CDL.nl relation set, the other challenge is that we need an effective relation extraction method.

Various lines of work on relation extraction have shown experimentally that the incorporation of diverse lexical, syntactic and semantic knowledge can lead to significant improvements in extraction accuracy. Miller, et al. (2000) augmented syntactic full parse trees with semantic information corresponding to entities and relations using rule-based method, and built generative models for the augmented trees. Kambhatla (2004) employed Maximum Entropy models for relation extraction with features derived from word, entity type, mention level, dependency tree and parse tree using feature-based method. Zhou et al. (2005) describes a relation detection approach that combines clues from different levels of syntactic processing: tokenization, sentence parsing and deep dependency analysis, and edge of three levels of language processing: syntactic analysis, dependency parsing and lexical construction.

This paper will further explore the kernel-based approach with a systematic study on the extensive incorporation of diverse lexical, syntactic and semantic information. In our model, based on the new semantic relation set, we present an relation classification method displaying two improvement over previous ones: (1) besides common used lexical information, we use a new lexical resource UNLK to provide semantic behavior features of words, and (2) we use kernel function to model lexical features separately from syntactic and dependency features and combine them into a composition kernel to improve the performance.

Kernel Method for Relation Classification

With the assumption that all entity pairs have been detected, this paper focus on the classification of CDL.nl relation labels. In this section, we describe a relation classification approach which uses kernel function to model diverse knowledge of three levels of language processing: syntactic analysis, dependency parsing and lexical construction.

Syntactic Kernel

Benefit from the Connexor Parser\(^3\), richful linguistic tags can be extracted as features to classify relations between entities. For each pair of entities of relation instances, in this section, we extract the following syntactic features and define a lexical kernel to match two relation instances.

**Morphology Features** Morphological information tells the details of word forms used in text. For example, for noun words, there are five tags: N(noun), SG(singular), PL(multiple), NOM(nominiative) and GEN(genitive).

We use a vector to represent the morphology feature space: \(X_{\text{Morp}} = (x_1, x_2, ..., x_70)\). Where \(x_i\) corresponds to a tag and receives "0" or "1" value, and Connexor Parser defines 70 morphology tags. For each entity \(E\), \(X_{\text{Morp}}(E) = (x_{11}, x_{12}, ..., x_{70})\). Where, \(x_{1i} = 1\) if the set of morphology tags of the headword of \(E\) contains the tag of the \(i\)th position, all other tags not contained will be set to \(x_{1i} = 0\).

**Syntax Features** Whereas morphology gives information on forms of words, syntax describes both surface syntactic and syntactic function information of words. For example, %NH (nominal head) and %>N (determiner or premodifier of a nominal) are surface syntactic tags, @SUB (Subject) and @F-SUBJ (Formal subject) are syntactic function tags.

Using the same way of dealing with morphology features, syntax features for an entity \(E\) are represented in a vector: \(X_{\text{Syn}}(E) = (x_{11}, x_{12}, ..., x_{40})\). Connexor Parser defines 40 Syntax tags. For two entities of a relation instance \(R\), the syntactic feature vector \(X(R)\) is defined as the concatenation of morphology and syntax vector:

\[
X(R) = (X_{\text{Morp}}(E_1)X_{\text{Morp}}(E_2)X_{\text{Syn}}(E_1)X_{\text{Syn}}(E_2))
\]

Then we define a syntactic kernel to match syntactic features between two relation instances \(R_1, R_2\) by simply calculating the dot product of two vectors:

\[
K_S(R_1, R_2) = X(R_1) \cdot X(R_2)
\]

**Dependency Kernel**

A dependency relation specifies an asymmetric grammatical function relationship between a pair of words, where one

---

\(^3\)www.connexor.com
class of word senses under the synonym relation. Synsets are organized by semantic relations such as Synonymy, Antonymy and Hyponymy. In WordNet 3.0, the total of all unique noun, verb, adjective, and adverb strings is actually 155287 along with 206941 word-sense pairs, containing 11529 verbs with 25047 verb-sense pairs. In this paper, we use hypernymy and synonymy to represent word sense feature and also use synonymy to extend later resources.

A vector \( W \) is defined to capture sense features containing word sense and hypernymy senses of the headword of each entity: \( W = \{w_1, w_2, ..., w_n\} \). Since each word may have many hypernym senses, we select the top four senses.

Take the word “chairman” for example, it has only one sense \{‘chairman’ in noun: president, chairman, chairwoman, chair, chairperson\} which has eight levels of hierarchy, the top four are \{‘noun: living thing, animate thing’\}, \{‘noun: object, physical object’\}, \{‘noun: entity’\}, \{‘noun: causal agent, cause, causal agency’\}.

VerbNet VerbNet (Kipper et al, 2000) is a verb lexicon compatible with WordNet, with explicitly stated syntactic and semantic information based on Levin’s verb classification. VerbNet associates the semantics of a verb with its syntactic frames, and combines traditional lexical semantic information such as thematic roles and semantic predicates, with syntactic frames and selectional restrictions. 5269 total verbs are organized into 237 syntactic frames in this resource. However, it has a problem of coverage: it contains less than half of 11529 verbs of WordNet, and provides only verb predicate-related information, doesn’t state semantic behaviors of nouns, adjectives and adverbs.

For example, word give belongs to syntactic frame lend which contains a set of verbs \{lend loan pass peddle refund render volunteer give hock rent sell lease paw\}. A vector \( V \) is defined to capture frame features for verb entities: \( V = \{v_1, v_2, ..., v_n\} \).

UNLKB Since we use CDL.nl semantic relation set, for each usage of the word, we define semantic behavior as a series of CDL.nl semantic relations in which the word participates. Since many words have different senses and usages, they may have several semantic behaviors. The UNLKB⁴ is a lexicon which organizes words in a hierarchy structure form by their semantic behaviors. It covers nouns, verbs, adjectives and adverbs and also associates semantic relations in behavior representation with word type restrictions. The total of all word-behavior pairs is about 65000, containing 15000 verb-behavior pairs. It explicitly implements the close relationship between syntax and semantics for nouns, verbs, adjectives and adverbs. Here are some word-behavior pairs of word give in UNLKB:

\[
\begin{align*}
give(agt>thing, obj>thing) \\
give(agt>thing, gol>person, obj>thing) \\
give(agt>thing, gol>object, obj>thing) \\
give(agt>volitional thing, obj>action)
\end{align*}
\]

It shows that word give has at least these four kinds of semantic behaviors. And for the second behavior, it has agent relation with a thing-type word, goal relation with a person-type word and object relation with a thing-type word. Here type of a word is a hypernym word of the word.

⁴www.undl.org/ulnlsys/uw/unlkb.htm
A vector $U$ is defined to capture the hierarchy hypernym of semantic behaviors of word: $U = (u_1, u_2, \ldots, u_n)$.

**Lexical Kernel Development** All these resources - VerbNet, WordNet and UNLKB - encoding different kinds of knowledge and we plan to extract three kinds of features: word sense, verb frame if the entry word is a verb, semantic behavior of words. However, both of VerbNet and UNLKB suffer from the coverage problem. We use the synonymy set from WordNet to extend them based on the assumption: words with same senses tend to share the same behaviors.

To explicitly capture these features, for the entity pair $E_1, E_2$, a new lexical feature vector $Y(R)$ is defined as the concatenation of all above lexical vectors:

$$Y(R) = (W(E_1)W(E_2)V(E_1)U(E_1)U(E_2))$$

Then we write the kernel to match lexical features between two relation instances $R_1, R_2$ by simply calculating the dot product of two vectors:

$$K_L(R_1, R_2) = Y(R_1) \cdot Y(R_2)$$

**Composition Kernel**

Having defined all the kernels representing syntactic, dependency and lexical processing results, we develop a composite kernel to combine and leverage individual kernels:

$$K = \alpha K_S + \beta K_D + \gamma K_L$$

This is the final kernel we used for this task. Trying with different $\alpha, \beta, \gamma$ values, we can observe performance of individual kernels and also of the composite kernel. Since all the individual kernels we defined can be seen directly/indirectly as matchings of features, it is clear that they are all valid kernels. And since the kernel function set is closed under linear combination, the composite kernels are also valid.

**Experiments**

**Experimental Setting**

The above kernels are experimented over a manual-annotated dataset$^3$ which contains about 1700 sentences from Wikipeida$^6$ documents. It was annotated with 13487 CDL.nl relations including 44 relation types. We evaluate the systems by using 10-fold cross validation.

We use one-vs-all scheme in which each binary classifier will be trained for each relation label over the composite kernel. The classifier evaluations are carried out using the SVM-light software (Joachims, 1998) with the default polynomial kernels. To process syntactic, dependency and lexical features, we implemented our own kernels and used them inside SVM-light.

**Preliminary Experimental Results**

The aim of our experiments is twofold: on the one hand, we study the performance of individual kernels and watch if adding kernels continuously improves the performance. On the other hand, we study how to leverage among syntactic, dependency and lexical features to get the best performance. Due to the insufficient dataset, we face with the data sparseness problem, so we choose to train on 36 of 44 relation types with large number of training instances.

---

$^3$http://www.miv.t.u-tokyo.ac.jp/mem/yyan/CDL.nl/

$^6$http://www.wikipedia.org/
Table 3: Evaluation on incremental lexical features.

<table>
<thead>
<tr>
<th>Lexicon</th>
<th>Precision</th>
<th>Recall</th>
<th>F-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A No-Lexicon</td>
<td>85.63</td>
<td>85.91</td>
<td>85.77</td>
</tr>
<tr>
<td>B A+WordNet</td>
<td>85.80</td>
<td>86.88</td>
<td>86.34</td>
</tr>
<tr>
<td>C B+VerbNet</td>
<td>85.77</td>
<td>87.13</td>
<td>86.45</td>
</tr>
<tr>
<td>D C+UNLKB</td>
<td>86.35</td>
<td>87.43</td>
<td>86.89</td>
</tr>
</tbody>
</table>

In order to compare the contribution of each lexicon, we also evaluate each kind of lexical features. As shown in table 3, the performance of using semantic behavior features from UNLKB lexicon is improved. And since VerbNet only works for event-related relation types, using it for all relation classifiers contributes less than WordNet and UNLKB.

Finally, figure 3 shows a graph describing the structure of a sentence where relations are classified and labeled by our algorithm. It shows that with CDL.nl relation set, unstructured sentence can be annotated into graph structure with not only predicate-argument relations, but also those between each pair of entities there exists a meaningful relationship, such as the \texttt{equ} (equivalent) relation between entities “\textit{Microsoft}” and “the software company” shows that both refer to the same object, and \texttt{arg} (thing with attribute) relation between “American entrepreneur” and “Gates” showing “Gates” has an attribute as “American entrepreneur”.

And through the preliminary experiments, we can see that CDL.nl relations can be classified with good performance (with Precision, Recall, F-value as 86.83%, 88.49%, 87.65%). We realize one of the important tasks of annotating occurring texts into structured forms, though we still have to deal with the entity pair boundary detection problem.

Conclusions and Future Work

In this paper, facing two challenges of semantic annotation of Web text, we have described a new shallow semantic parser of which (1) we used a new set of semantic relations of CDL.nl which are more competent than that of SRL to represent the semantic structure of text and (2) we proposed a relation classification approach displaying two advantages over previous ones by using a new lexicon to provide semantic behavior features of word, and using kernel method to model lexical features separately from syntactic and dependency features into a composition kernel. Experiments on our manual dataset of 1700 sentences showed that CDL.nl relations can be classified with good performance and two advantages of our method can improve the performance of relation classification.

The immediately extension of our work is to improve the performance of relation classification from two ways: firstly, to enrich the dataset, we plan to bootstrap the classifiers in larger amount of data; secondly, use word sense disambiguation techniques and existed well known kernels, such as tree kernels, to get best use of syntactic and lexical features.

References


