to appear in *Proc. IJCAI-03,* Poster Session, Acapulco, August 12-15, 2003, Copyright © IJCAI (www.ijcai.org)

Inducing criteria for lexicalization parts of speech using the Cyc KB

Tom O'Hara
Computer Science Department
New Mexico State University
Las Cruces, NM 88001
tomohara@cs.nmsu.edu

Michael Witbrock, Bjørn Aldag, Stefano Bertolo, Nancy Salay, Jon Curtis and **Kathy Panton**
Cycorp, Inc.
Austin, TX 78731
{witbrock,bertolo,aldag,nancy,jonc,panton}@cyc.com

Abstract

We present an approach for learning part-of-speech distinctions by induction over the lexicon of the Cyc knowledge base. This produces good results (74.6%) using a decision tree that incorporates both semantic features and syntactic features. Accurate results (90.5%) are achieved for the special case of deciding whether lexical mappings should use count noun or mass noun headwords. Comparable results are also obtained using OpenCyc, the publicly available version of Cyc.

1 Introduction

In semantic lexicons, a *lexical mapping* describes the relationship between a concept and phrases used to refer to it. This mapping includes syntactic information for deriving variant phrases, including the part of speech of the headword. Selecting the part of speech for the lexical mapping is required so that proper inflectional variations can be recognized and generated for the term. Although often a straightforward task, there are special cases that can pose problems, especially when fine-grained speech part categories are used.

To reduce the need for linguistic expertise in producing these *lexicalizations* for a large knowledge base like Cyc [Lenat, 1995], linguistic criteria can be inferred from decisions that have been made by lexical knowledge engineers in *lexicalizing* preexisting terms.

The Cyc knowledge base, containing 120,000 concepts and over one million axioms,[1] divides roughly into three layers. The upper ontology formalizes fundamental distinctions (e.g., tangibility versus intangibility). The lower ontology collects specific facts, often related to concrete applications, and the middle ontology encodes commonsense knowledge about the world. The KB also includes a broad-coverage English lexicon mapping words and phrases to terms throughout the KB.

Natural language lexicons are integrated directly into the Cyc KB [Burns and Davis, 1999]. Binary predicates, as in *(nameString HEBCompany "HEB")*, map names to terms. A denotational assertion maps a phrase, specified via a lexical concept with optional string modifiers, into a concept, usually a collection. The part of speech is specified by Cyc's *SpeechPart* constants. The simplest type of denotational mapping uses the *denotation* predicate. For example, *(denotation Device-Word CountNoun 0 PhysicalDevice)* indicates that sense 0 of the count noun 'device' refers to *PhysicalDevice* (via the wordforms "device" and "devices"). Three additional predicates account for phrasal mappings: *compoundString*, *headMedialString*, and *multiWordString* are used for phrases with the headword at the beginning, the middle, and the end, respectively.

These denotational assertions, excluding lexical mappings for technical, informal and slang terms, form the training data for a lexicalization speech part classifier.

2 Inference of lexicalization part of speech

Our method of inferring the part of speech for lexicalizations is to apply decision tree learning over existing lexical mappings. For each target denotatum term, the corresponding definitional information (e.g., *isa*), asserted or inferable via transitivity, is extracted from the ontology. For simplicity, these definitional types are referred to as *ancestor terms*. Associations between the lexicalization parts of speech and common ancestor terms underlie the lexicalization speech part classifier and its special case, the mass-count classifier (distinguishing, e.g., "much *sand*" from "many *books*"). To reduce the size of the training feature vector, only the most frequent 256 atomic terms from the thousands of possible ancestor terms are selected, after excluding certain internal bookkeeping constants.

Given a training instance, such as a denotation from a lexeme into a specific Cyc concept using a particular *SpeechPart* (e.g., *MassNoun* or a *CountNoun*), the feature specification is derived by determining all the ancestor terms of the denotatum term and converting this into a vector of occurrence indicators, one indicator per reference term. Then the headword is checked for the occurrence of a set of commonly used suffixes. If found, the suffix itself is added to the vector (in a position set aside for suffixes of the same length). The part of speech serves as the classification variable.

We use decision trees for this classification. Part of the motivation is that the result is readily interpretable and can be incorporated directly by knowledge-based applications. A

[1] These figures and the results discussed later are based on Cyc KB v576 and system v1.2577. See www.cyc.com for detailed documentation on the KB and [O'Hara *et al.*, 2003] for more technical details related to this work.

	OpenCyc	Cyc
Instances	2607	30676
Classes	2	2
Entropy	0.76	0.90
Baseline	78.3	68.2
Accuracy	87.5	*90.5*

Table 1: Mass-count classification over Cyc lexical mappings, using Cyc reference terms and headword suffixes as features. *Instances* is size of the training data. *Classes* indicates number of choices. *Baseline* selects most frequent case. *Accuracy* is average in the 10-fold cross validation.

	OpenCyc	Cyc
Instances	3721	43089
Classes	16	33
Entropy	1.95	2.11
Baseline	54.9	49.0
Accuracy	71.9	*74.6*

Table 2: General speech part classification using Cyc.

simple fragment from the resulting decision tree shows how ontological features interact with morphological ones:

```
if (isa AbstractInformationalThing) then
  if (suffix = "-er") then
    if (not isa SomethingExisting) then AgentiveNoun
    if (isa SomethingExisting) then MassNoun
  if (suffix = "-ed") then MassNoun
  if (suffix = "-al") then Adjective
  if (suffix = "-or") then AgentiveNoun
```

Table 1 shows the results of 10-fold cross validation for the mass-count classification. This was produced using the J48 algorithm in the Weka machine learning package [Witten and Frank, 1999], which is an implementation of Quilan's C4.5 [Quinlan, 1993] decision tree learner. This shows that the system achieves an accuracy of 90.5%, an improvement of 22.3 percentage points over a baseline of always selecting the most frequent case. The OpenCyc version of the classifier also performs well. This suggests that sufficient data is already available in OpenCyc (available online at www.opencyc.org) to allow for good approximations for such classifications.

The mass/count noun distinction can be viewed as a special case of speech part classification. Running the same classifier using the full set of speech part classes yields the results shown in Table 2. Here the overall result is not as high, but there is a similar improvement over the baseline.

3 Discussion

Contextual part of speech tagging [Brill, 1995] has received substantial attention in the literature, but there has been relatively little written on automatically determining default lexicalization parts of speech. Woods [Woods, 2000] describes an approach to this problem using manually-constructed rules incorporating syntactic, morphological, and semantic tests (via an ontology). Our rules are induced from the knowledge base, which alleviates the need for rule maintenance as well as rule construction.

This paper shows that an accurate decision procedure (90.5%) for determining the mass-count distinction in lexicalizations can be induced from the lexical mappings in the Cyc KB. The performance (74.6%) in the general case is also promising, given that it is a much harder task with over 30 part of speech categories to choose from. The features incorporate semantic information, in particular Cyc's ontological types, in addition to syntactic information (e.g., headword morphology). Although the main approach incorporates Cyc's conceptual distinctions, it can be extended to non-Cyc applications via the WordNet mapping [O'Hara *et al.*, 2003].

This work is just a small initial step in applying machine learning techniques to the massive amount of data in Cyc. The recent release of OpenCyc enables wider investigation and exploitation of the information in the Cyc knowledge base for intelligent applications.

Acknowledgements

The lexicon work at Cycorp has been supported by many staff members, and, in part by grants from NIST, DARPA, and ARDA. The first author is currently supported by a GAANN fellowship from the Department of Education. The work utilized NMSU resources obtained through MII Grants EIA-9810732 and EIA-0220590.

References

[Brill, 1995] Eric Brill. Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. *Computational Linguistics*, 21(4):543–565, 1995.

[Burns and Davis, 1999] Kathy J. Burns and Anthony B. Davis. Building and maintaining a semantically adequate lexicon using Cyc. In Evelyn Viegas, editor, *The Breadth and Depth of Semantic Lexicons*, pages 121–143. Kluwer, 1999.

[Lenat, 1995] D. B. Lenat. Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11), 1995.

[O'Hara *et al.*, 2003] Tom O'Hara, Nancy Salay, Michael Witbrock, Dave Schneider, Bjoern Aldag, Stefano Bertolo, Kathy Panton, Fritz Lehmann, Matt Smith, David Baxter, Jon Curtis, and Peter Wagner. Inducing criteria for mass noun lexical mappings using the Cyc KB, and its extension to WordNet. In *Proc. Fifth International Workshop on Computational Semantics (IWCS-5)*, 2003.

[Quinlan, 1993] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, California, 1993.

[Witten and Frank, 1999] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, 1999.

[Woods, 2000] W. Woods. Aggressive morphology for robust lexical coverage. In *Proc. ANLP-00*, 2000.