BURC: Bootstrapping Using ResearchCyc

Kino Coursey
Daxtron Laboratories, Inc.
1001 W. Northwest Highway, Suite E
Grapevine, Texas 76051
kino@daxtron.com

Abstract

The goal of this research is to expand the knowledge in a large ontology called ResearchCyc by using existing knowledge in that ontology to extract knowledge for later use in that same system. In this way the system can be said to bootstrap, expanding its own knowledge through reading text, hopefully in a manner that will improve future knowledge acquisition. The system processes large quantities of texts using the CMU Link Grammar Parser, and using the individual links coupled with ResearchCyc, existing knowledge can generate new assertions about relationships implied to be possible, normal or commonplace in the world. This is possible by aggregating information from a large number of parse links over a large corpus

1 Introduction

The goal of this project is to extend ResearchCyc's (Cyc) knowledge base using *"relationships implied to be possible, normal or commonplace in the world"* found in processed text. General knowledge can be found based on the relationship between the concepts denoted by the words in sentences describing common occurrences. Cyc has a broad vocabulary for describing concepts, but only a fraction of the commonplace relationships. This project will investigate how the coverage of commonplace relationships could be improved by automated means.

Prior work on knowledge entry in Cyc and in general has been manually oriented. Such an approach is labor intensive and prone to lack of inclusion due to incomplete coverage by the ontological engineers. It is desirable to utilize available large-scale linguistic resources for creation of large-scale, broad-coverage ontologies.

Written language makes available to readers a valuable resource to gather knowledge about the world. For a very long time one common goal has been for machine to utilize this resource. One approach to learning from reading is to recognize what one can to frame the gathering of new knowledge. Once basic information is collected one can re-read if necessary to gather more knowledge. In this way as more is read, more is understood. Also, a typical reader will notice multiple references to a term or situation over time, and assume that if multiple texts in a topic mention the same thing, that it is either a common occurrence or common concern. Also a typical reader will notice references to information they already know, and seeing how the known relationship is expressed in text, will use that knowledge to infer new relationships over new texts.

The basic approach taken in this paper is inspired by some of the methods used in the Human Genome Project. Instead of careful, bit-by-bit parsing of the genetic sequence, the "shotgun approach" broke the genome into easy to process fragments, and then connected the common pieces. In this project text is parsed and the individual fragments are placed in a database where statistics on co-occurrences are collected. Then those fragments that occur often enough are used to hypothesize a common relationship, based on the fact that they have a wide occurrence in the processed text.

Like humans, a system that can gather information by reading texts can gain understanding beyond its direct experience. The result of an effective reader would be a system that pre-assembles the base knowledge used to represent more complex relationships.

2 Background

A large component of natural language processing is representing the information contained in linguistic communication. However, a large body of the work is focused on isolated sentences input into an application, not extracting meaning from a large body of text. The

KNEXT system (Knowledge EXtraction from Text) [1][2][3] is one system that explores extracting common sense relations from text. An example sentence and the extracted information is shown below:

```
(BLANCHE KNEW 0 SOMETHING MUST BE CAUSING
STANLEY 'S NEW , STRANGE BEHAVIOR BUT SHE
NEVER ONCE CONNECTED IT WITH KITTI WALKER .)
```

Output in English, followed by underlying logical forms:

```
A FEMALE-INDIVIDUAL MAY KNOW A PROPOSITION.
SOMETHING MAY CAUSE A BEHAVIOR.
A MALE-INDIVIDUAL MAY HAVE A BEHAVIOR.
A BEHAVIOR CAN BE NEW.
A BEHAVIOR CAN BE STRANGE.
A FEMALE-INDIVIDUAL MAY CONNECT A THING-
  REFERRED-TO WITH A FEMALE-INDIVIDUAL.

((:I (:Q DET FEMALE-INDIVIDUAL) KNOW[V] (:Q
  DET PROPOS))
 (:I (:F K SOMETHING[N]) CAUSE[V] (:Q THE
  BEHAVIOR[N]))
 (:I (:Q DET MALE-INDIVIDUAL) HAVE[V] (:Q
  DET BEHAVIOR[N]))
 (:I (:Q DET BEHAVIOR[N]) NEW[A]) (:I (:Q
  DET BEHAVIOR[N]) STRANGE[A])
 (:I (:Q DET FEMALE-INDIVIDUAL) CONNECT[V]
  (:Q DET THING-REFERRED-TO)
  (:P WITH[P] (:Q DET FEMALE-INDIVIDUAL))))
```

This process is fragment oriented. The relationship hypothesized is based on local relationships returned from the parser between terms. KNEXT generated 117,326 propositions (about two per sentence) from the Browns corpus; about 60% of them were judged "reasonable" by human judges. The work reported has been limited to the Penn Treebank corpora[1].

As pointed out by Chklovski & Pantel [10] in their VerbOcean project, most work on extracting semantic information from large corpora has been focused on extracting "is-a" relationships between nouns, or finding a specific subset of relationships or attributes. Work has been done on using surface lexical patterns to extract information for question answering. Examples of surface pattern extraction methods can be found in DIRT–Discovery of Inference Rules from Text (Lin & Pantel, 2001)[13], Towards Terascale Knowledge Acquisition (Pantel, Ravichandran and Hovy, 2004)[11][12] and Learning Surface Text Patterns for a Question Answering System (Ravichandran & Hovy, 2002)[14][15]. One useful measure defined in this body of work is Pattern Precision.

```
Error!Pattern Precision P = Ca/Co
   Ca = total number of patterns with
    answer term present
   Co = Total number of patterns with any
    term present
```

These surface extraction rules can be seen as being backward-forward approaches. Starting with initial examples of relationships, they find how that information is expressed as a pattern (the backward direction), then they apply them in the corpora to find new relationships (the forward direction).

Other approaches to extracting useful information from large quantities of text exist. *VerbOcean* [10](Chklovski & Pantel) examines pairs of verbs and searches the Internet through Google to verify relationships by matching against a set of pre-defined patterns.

Lexical Acquisition via Constraint Solving (Pedersen & Chen)[9] acquires syntactic and semantic classification rules of unknown words for CMU Link Grammar Parser (LGP) using the selectional restrictions of the parts of the sentence the system can already parse.

Some research has been done on using the link grammar parser for information extraction. *Dependency-based semantic interpretation for answer extraction* (Molla-Aliod, et. al.)[6] describes a method for translation of link grammar parses into logical forms. Similar methods for translating semantic information from LGP output are used by others (*Event Information Extraction Using Link Grammar[7], Knowledge Representation in KRIS Using Link Grammar Parser[4], Learning to Generate CGs from Domain Specific Sentences[5]*)

The "Learning by Reading" project [17] has similar goals. It is focused on extracting more forms of knowledge over simplified texts. BURC is focused on extracting the broad range of Cyc relationships from significant occurrences in parser output.

3 What is Cyc?

Cyc[2] began in the mid-1980s as a logic-based inference engine designed to provide broad coverage of common topics for use in applications. According to Cycorp, Cyc is "the world's largest and most complete general knowledge base and commonsense reasoning engine." Cyc contains an inference engine, a large ontology, and both natural language processing tools and information, with the facility to link to WordNet. Today's ResearchCyc contains roughly 85,000 concepts, 2300 relationships and close to one million assertions.

A paraphrase of *Lenat's Bootstrap Hypothesis* is that once Cyc reaches a certain level/scale it can help in its own development and start using NLP to augment it

[1] Source: "Deriving General World Knowledge from Texts and Taxonomies" http://www.cs.rochester.edu/~schubert/projects/world-knowledge-mining.html

[2] Copies available from http://www.opencyc.org and http://researchcyc.cyc.com.

knowledge base. This project is one test of that hypothesis. Cyc has a vocabulary about objects in the world and their relationships. However, Cyc could use still more knowledge about common relationships. Just how much can Cyc help generate and aid in the use of extraction patterns?

An example of what Cyc currently knows about fingers (29 factoids):

Collection : Finger
GAF Arg : 1
Mt : UniversalVocabularyMt
isa : AnimalBodyPartType
quotedIsa : DensoOntologyConstant
genls : Digit-AnatomicalPart
comment : **"The collection of all digits of all Hands (q.v.). Fingers are (typically) flexibly jointed and are necessary to enabling the hand (and its owner) to perform grasping and manipulation actions."**

Mt : BaseKB
definingMt : AnimalPhysiologyVocabularyMt

Mt : AnimalPhysiologyMt
properPhysicalPartTypes : Fingernail

Mt : WordNetMappingMt
(synonymousExternalConcept Finger WordNet-Version2_0 **"N05247839")**
(synonymousExternalConcept Finger WordNet-1997Version **"N04312497")**

GAF Arg : 2
Mt : UniversalVocabularyMt
(genls LittleFinger Finger**)**
(genls IndexFinger Finger**)**
(genls Thumb Finger**)**
(genls RingFinger Finger**)**
(genls MiddleFinger Finger**)**

Mt : HumanActivitiesMt
(bodyPartsUsed-TypeType Typing Finger**)**

Mt : HumanSocialLifeMt
(bodyPartsUsed-TypeType PointingAFinger Finger**)**

Mt : AnimalPhysiologyMt
(conceptuallyRelated Fingernail Finger**)**
(properPhysicalPartTypes Hand Finger**)**
(relationAllInstance age Finger (YearsDuration **0 200))**
(relationAllInstance widthOfObject Finger (Meter **0.001 0.2))**
(relationAllInstance heightOfObject Finger (Meter **0.001 0.2))**
(relationAllInstance lengthOfObject Finger (Meter **0.01 0.5))**
(relationAllInstance massOfObject Finger (Kilogram **0.001 1))**

GAF Arg : 3
Mt : HumanPhysiologyMt
(relationAllExists anatomicalParts HomoSapiens Finger**)**

Mt : VertebratePhysiologyMt
(relationAllExistsCount physicalParts Hand Finger **5)**

Mt : UniversalVocabularyMt
(relationAllOnly wornOn Ring-Jewelry Finger**)**

Mt : AnimalPhysiologyMt
(relationExistsAll physicalParts Hand Finger**)**

GAF Arg : 4
Mt : GeneralEnglishMt
(denotation Finger-TheWord CountNoun 0 Finger)

4 Method

The basic method is to use what Cyc already knows, combined with a large collection of parser output from a large corpora, to generate new Cyc entries from common relationships implied in the text. The system works using two basic methods, forward from text to Cyc, and a combination of backward from existing Cyc knowledge to discover patterns then applying the patterns forward over text to derive new information.

1. Use the CMU Link Grammar Parser (LGP)[3] for bulk parsing of text, primarily narratives based in 'worlds like ours.' Other text styles could be included.
2. Load the link fragments into a database (1 and 2 link), and compute frequency of fragment occurrences. The database will be in a SQL format so multiple queries can be formed dynamically.
3. Extract knowledge for use in Cyc, using Cyc knowledge as a starting point (the "seeds"). Given a set of seed facts in Cyc, identify how those facts are represented as link fragments in the database, and generate conjectures as to new knowledge using the fragment patterns.
4. Use Cyc knowledge directly to conjecture new statements. Cyc has lexical knowledge, which can be used as templates against the DB to form new statements. For example, common adjectives applied to noun classes. Cyc knows "WhiteColor" and "Blouse" but does not know that white is a common blouse color, although it becomes apparent after reading some text.
5. Optionally, gather supporting background statistics for hypothesis generation using Google or some large search engine. [Or: perhaps Google desktop with a larger than fully parsed corpus]. Check against answer extraction engines?

[3] See http://bobo.link.cs.cmu.edu/link/index.html.

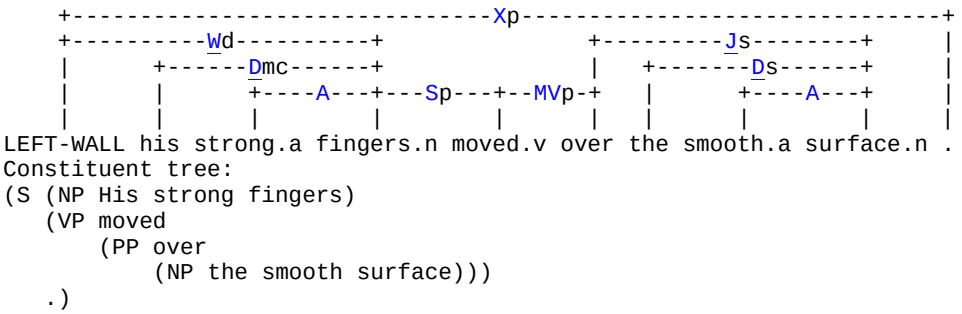


Figure 1: Example of a link grammar parse

Another way of stating the operation is application of n-gram concepts to parser path fragments. In the initial version of BURC the system examines the occurrence of bi-grams and tri-grams of parser path fragments.

4.1 The Process of Forward Mining Adjective Relations

There are 1941 basic assertions on adjSemTrans (the semantic translation template for adjectives), the primary lexical adjective predicate. We can directly apply these templates to a database of parser outputs. Each parse fragment record contains the number of links in a fragment (1 or 2), the link grammar label of the links and the terms involved with their tagging.

1. Generate a SQL query of "Select * from LGPTable Where NumLinks=1 and Link1='a' and Term1 like '%.a' and Term2 like '%.n' " This will find all one link fragments with the adjective link "A" connecting words tagged .a and words tagged .n
2. Returns records of the form [Term1.a | a | Term2.n]
3. Potentially test using either an internal or search engine based relevancy metric
4. For each surviving record:
   a) Query Cyc for "(adjSemTrans <term1>-TheWord ?N RegularAdjFrame (?Pred :NOUN ?Val))"
   b) Generate **(plausiblePredValOFType <term2> <?Pred> <?Val>)**

4.2 Example of processing an adjective factoid

Given "strong fingers" as a fragment, the LGP generates a basic relationship record of [strong.a | a | fingers.n]. A quick check of Google shows that this fragment occurs 29,700 times, and thus may be a common relationship. Looking up the template for "Strong" we find:

```
(adjSemTrans Strong-TheWord 0 RegularAd-
jFrame (forceCapacity :NOUN Strong)).
```

So, the system generates a hypothesis:

```
(plausiblePredValueOfType Finger forceCapac-
ity Strong).
```

One additional step can be to use the adjSemTrans data to generate new translation rules for the parser. This would result in better parser precision.

4.3 Mined Finger Descriptions (26 unique factoids)

```
000010:(#$plausiblePredValueOfType #$Finger
       #$feelsSensation (#$PositiveAmountFn
       #$LevelOfSoreness))
000037:(#$plausiblePredValueOfType #$Finger
       #$forceCapacity #$Strong)
000025:(#$plausiblePredValueOfType #$Finger
       #$forceCapacity #$Strong)
000025:(#$plausiblePredValueOfType #$Finger
       #$hardnessOfObject #$Hard)
000037:(#$plausiblePredValueOfType #$Finger
       #$hardnessOfObject (#$MediumToVery-
       HighAmountFn #$Hardness))
000037:(#$plausiblePredValueOfType #$Finger
       #$hardnessOfObject (#$MediumToVery-
       HighAmountFn #$Hardness))
000002:(#$plausiblePredValueOfType #$Finger
       #$hasEvaluativeQuantity (#$MediumToV-
       eryHighAmountFn #$Goodness-Generic))
000002:(#$plausiblePredValueOfType #$Finger
       #$hasPhysicalAttractiveness #$GoodLook-
       ing)
000047:(#$plausiblePredValueOfType #$Finger
       #$isa (#$LeftObjectOfPairFn :REPLACE))
000015:(#$plausiblePredValueOfType #$Finger
       #$isa (#$RightObjectOfPairFn :REPLACE))
000155:(#$plausiblePredValueOfType #$Finger
       #$lengthOfObject (#$RelativeGenericVal-
       ueFn #$lengthOfObject :REPLACE
       #$highAmountOf))
000155:(#$plausiblePredValueOfType #$Finger
       #$lengthOfObject (#$RelativeGenericVal-
       ueFn #$lengthOfObject :REPLACE #$high-
       ToVeryHighAmountOf))
000003:(#$plausiblePredValueOfType #$Finger
       #$mainColorOfObject #$BlackColor)
000010:(#$plausiblePredValueOfType #$Finger
       #$mainColorOfObject #$LightYellow-
       ishBrown-Color)
000010:(#$plausiblePredValueOfType #$Finger
       #$mainColorOfObject #$ModerateYellow-
       ishBrown-Color)
000010:(#$plausiblePredValueOfType #$Finger
       #$mainColorOfObject #$SunTan-
       FleshColor)
000002:(#$plausiblePredValueOfType #$Finger
       #$possessiveRelation #$SuddenChange)
```

```
000006:(#$plausiblePredValueOfType #$Finger
       #$possessiveRelation (#$HighAmountFn
       #$Speed))
000094:(#$plausiblePredValueOfType #$Finger
       #$rigidityOfObject (#$HighAmountFn
       #$Rigidity))
000060:(#$plausiblePredValueOfType #$Finger
       #$sizeParameterOfObject (#$Rela-
       tiveGenericValueFn #$sizeParameterOfOb-
       ject :REPLACE #$highAmountOf))
000052:(#$plausiblePredValueOfType #$Finger
       #$sizeParameterOfObject (#$Rela-
       tiveGenericValueFn #$sizeParameterOfOb-
       ject :REPLACE
       #$highToVeryHighAmountOf))
000060:(#$plausiblePredValueOfType #$Finger
       #$sizeParameterOfObject (#$Rela-
       tiveGenericValueFn #$sizeParameterOfOb-
       ject :REPLACE
       #$highToVeryHighAmountOf))
000285:(#$plausiblePredValueOfType #$Finger
       #$sizeParameterOfObject (#$Rela-
       tiveGenericValueFn #$sizeParameterOfOb-
       ject :REPLACE #$veryLowToLowAmountOf))
000074:(#$plausiblePredValueOfType #$Finger
       #$sizeParameterOfObject (#$Rela-
       tiveGenericValueFn #$sizeParameterOfOb-
       ject :REPLACE #$veryLowToLowAmountOf))
000029:(#$plausiblePredValueOfType #$Finger
       #$speedOfObject-Underspecified (#$LowA-
       mountFn #$Speed))
000138:(#$plausiblePredValueOfType #$Finger
       #$surfaceFeatureOfObj #$Slippery)
000074:(#$plausiblePredValueOfType #$Finger
       #$temperatureOfObject #$Warm)
000004:(#$plausiblePredValueOfType #$Finger
       #$textureOfObject #$Rough)
000168:(#$plausiblePredValueOfType #$Finger
       #$thicknessOfObject (#$Rela-
       tiveGenericValueFn #$thicknessOfObject
       :REPLACE #$highAmountOf))
000168:(#$plausiblePredValueOfType #$Finger
       #$thicknessOfObject (#$Rela-
       tiveGenericValueFn #$thicknessOfObject
       :REPLACE #$highToVeryHighAmountOf))
000182:(#$plausiblePredValueOfType #$Finger
       #$wetnessOfObject #$Wet)
```

4.4 Verb Semantic Filtering

A process similar to that used for adjectives can be used for finding information based on Cyc's verb semantic parsing frames. For each potential <NOUNWORD>-<VERB> pair query Cyc to find basic relationships using the verb semantic templates. The CycL query used would be:

```
(#$and
  (#$denotation <NOUNWORD> ?NOUNTYPE ?N
?CYCTERM)
  (#$wordForms ?WORD ?PRED ""<VERB>"")
  (#$speechPartPreds ?POS ?PRED
  (#$semTransPredForPOS ?POS ?SEMTRANSPRED)
  (?SEMTRANSPRED ?WORD ?NUM ?FRAME
?TEMPLATE))
```

One would then verify for each potential relationship (<SPRED> <VERTERM> <CYCTERM>) derivable from ?TEMPLATE that it makes sense in the ontology using selectional constraints.

```
 (#$and
  (#$arg1Isa <SPRED> ?VTYP)
  (#$arg2Isa <SPRED> ?CTYP)
  (#$genls <CYCTERM> ?CTYP)
  (#$genls <VERBTERM> ?VTYP) )
```

Translation: *The VERBTERM is a specialization of something that can be the first argument of SPRED and CYCTERM is a specialization of something that can be the second argument of SPRED.*

4.5 Cyc's Verb Semantic Translation Template for 'Move-TheWord'

```
(verbSemTrans Move-TheWord 0
IntransitiveVerbFrame
       (and
           (isa :ACTION MovementEvent)
           (primaryObjectMoving :ACTION
:SUBJECT)))

(verbSemTrans Move-TheWord 1
IntransitiveVerbFrame
       (and
           (isa :ACTION ChangeOfResidence)
           (performedBy :ACTION :SUBJECT)))

(verbSemTrans Move-TheWord 2 TransitiveNPFrame
       (and
           (isa :ACTION
CausingAnotherObjectsTranslationalMotion)
           (objectActedOn :ACTION :OBJECT)
           (doneBy :ACTION :SUBJECT)))
(arg1Isa performedBy  Action)

(arg2Isa performedBy  Agent-Generic)
```

Using Cyc's knowledge of various selectional restrictions on actions and attributes the system can filter out implausible relationships.

4.6 Mined Finger Related Actions (8 additional factoids)

```
(#$behaviorCapableOf #$Finger #$CausingAn-
otherObjectsTranslationalMotion #$doneBy)
~~(#$behaviorCapableOf #$Finger #$ChangeOfResi-~~
~~dence #$performedBy)~~
~~(#$behaviorCapableOf #$Finger #$Inspecting~~
~~#$performedBy)~~
(#$behaviorCapableOf #$Finger #$Movement-
TranslationEvent #$primaryObjectMoving)
(#$behaviorCapableOf #$Finger #$MovementEvent
#$primaryObjectMoving)
(#$behaviorCapableOf #$Finger #$PushingAnObject
#$providerOfMotiveForce)
(#$behaviorCapableOf #$Finger #$Sliding-Generic
#$objectMoving)
(#$behaviorCapableOf #$Finger #$Sliding-Generic
#$primaryObjectMoving)
(#$behaviorCapableOf #$Finger #$Slipping #$ob-
jectMoving)
(#$behaviorCapableOf #$Finger #$Slipping #$pri-
maryObjectMoving)
```

This is an example of how Cyc can help in its own knowledge entry process. In it, the system was able to eliminate two possible actions, ChangeOfResidence and Inspection, from using semantic selectional restrictions. Each of these actions are performed by a full agent, not just an agent part. 62% of verb-based generated hypotheses were filtered out using semantic role filtering.

4.7 The General Backwards Model

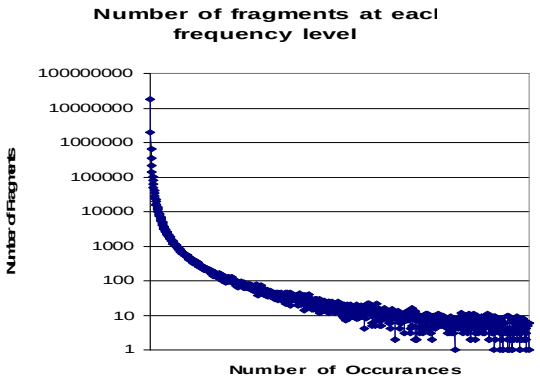
The primary idea behind the Backwards model is to find how existing Cyc knowledge is expressed for various known relationships in texts, and then use that text pattern to find new instances of that relationship. Given some Cyc relation Pred(?X,?Y):

1. Create SQL search query:
   a) Lookup in Cyc lexical entries for X & Y ➔ LX, LY
   b) Select * from LGPTable where Term1="<LX>" and Term3="<LY>“
   c) System returns records [LX | Link1 | Term2 | Link2 | LY] (Freq)
2. Generate new hypothetical extraction patterns|:
   a) Select * from LGPTable where Link1="<L1>" and Link2="<L2>" and Term2="<T2>“
   b) [* L1 T2 L2 *] ➔ generate hypothetical record (Pred | ?S1 | ?S3)
   c) Frequency information is propagated forward

One option is to search Cyc for ?PRED (X,Y) and use the set to form a local ambiguity class to reduce search labor and identify ambiguity. In this way one rule would hypothesize multiple ambiguous relationships. This would relate one template to multiple predicates. Other options include generating instance-specific extraction patterns (for Pred(X,_) and Pred(_,Y)) and updating the LGParser to CycL generation rules.

4.8 Current Status

LGP parsing was performed using ‘only best’ parse settings for filtered British National Corpus. Sentences that were over 256 characters or caused a parser panic were ignored. Five 3 Ghz Pentium 4 Linux computers were used to process the text in parallel over approx 70 hours, and their combined output was merged. The parsers generated 1.951 Gigs of data, which reduced down to 1.001Gig of unique countable elements. This consisted of 264 Megs (2,232,683 instances) of sentence data and 783 Megs (21,528,980 instances) of link fragments. There were approx. 35.3 Meg (996810) adjective link fragments, and 35.4 Meg (934029) of subject-verb-object link fragments. There were roughly the same numbers of unique adjective fragments (4.63%) as S-V-O fragments (4.34%).



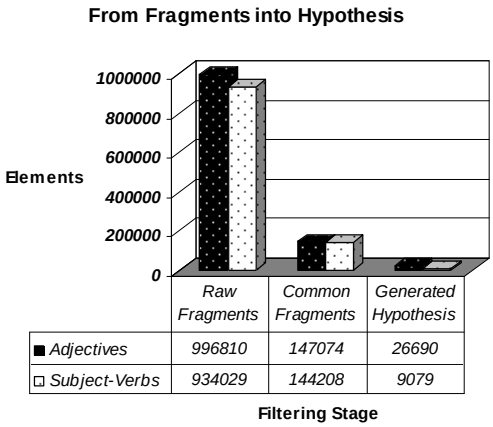
The fragment distribution was plotted for first 1000 bins to see the distribution of fragment occurrences.

Most work has been done in the forward direction. This provides familiarization with the tools and generation of output. Work on the backwards miner has not been completed.

5 System Output

The forward adjective mining process was run on adjective fragments with more than one instance. This was 147074 records or the top 14.75% adjective fragments. Of this 26690 relations were hypothesized, or 18.15% of those selected generated a hypothesis.

The verb extraction process selected fragments with more than 2 fragments. This resulted in 144208 records or the top 15.44% of subject-verb fragments. This generated 9079 hypothesis or 6.3% of fragments generated hypothesis.



This shows the percentage of mined relations that knowledge can be extracted from. One possible flaw with this measure is how to count when duplicate n-grams generate the same hypothesis. In such cases the target hypothesis should have more support.

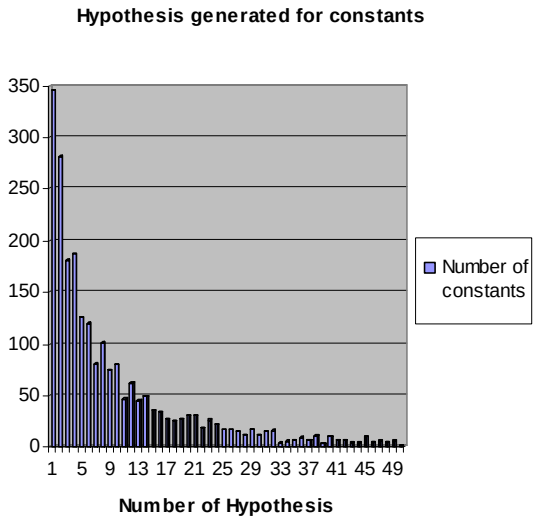
One measure would be the coverage of Cyc to relations extracted. What fraction of adjective fragments could Cyc explain or produce a parse, and then what

fraction are reasonable? The subject-verb data showed the percentage reduction in hypothesis generated using selectional restrictions. Future measurement will determine how well Cyc existing knowledge prunes the hypothesis space.

6 Discussion

What BURC is able to do is to quantify how common the proposed relationships are in corpora processed. As the corpora match the distribution of information and events in the real world, so should the distribution of assertions generated by BURC. In the finger example, Cyc had 29 original statements. Using forward adjective and verb mining BURC generated 34 additional facts. The various versions of Cyc average between 10 to 20 assertions per constant while BURC generated an average of 14.29 hypotheses per constant.

A long term goal of Cyc is to increase the useful content. With further development and once the quality of the hypothesis is quantified, a BURC-like approach could be one way to significantly augment the manually constructed KB.



6.1 Future directions

Systems like ConceptNet have a much coarser defined relationship hierarchy. One possible output is to modify the LGP to generate "underspecified" relationships based just on the links exist. This would allow ConceptNet compatible information to be generated directly from LGP processing.

Examples:

```
[<obj1>|ss|<act>.v|os|<obj2>] ➔ capableOf(<obj1>, "<act> <obj2>")
[<act>.v |os|<obj>] ➔ CapableOfReveivingAction(<obj>,<act>)
[<obj>|s*|<act>.v] ➔ capableOf(<obj>,<act>)
```

Given the set of hypothesized information, Cyc rules need to be added to the KB to exploit the new information. For instance Cyc can engage in knowledge extraction dialogs with user to learn about new topics. The new knowledge extracted by BURC can improve the question answer process, by proposing plausible defaults, and by querying about possible relationships it saw in text. The template-based parser was recently made available in ResearchCyc. One output could be additional information to aid the template parser.

Additional areas of research include:

- Add NL rules to ResearchCyc to allow querying of acquired and lexical knowledge.
- Add rules to allow knowledge entry tools to use BURC data for asking user questions.
- Use information from the forward runs of BURC to identify Cyc predicates for backward runs.
- Identify a graceful way to integrate BURC plausible hypothesis with Cyc's relationAllExists and relationAllInstance predicates that contain similar information in a similar format.
- Provide additional filtration of extracted relations. A fragment in isolation lacks the context to identify a particular sense of a polysemous word. Hence the use of selectional restrictions. However, given the desire to generate a hypothesis, the system could return to the original texts in a focused way to find the most probable sense.
- Add rules to notice either typical or unique attributes of new instances based on plausible expectations and given a novel value generate plausible connections to other classes.
- Given applications that use semantic similarity measures, determine if adding BURC extracted knowledge improves those applications performance.
- Utilize lexical resources like WordNet to explore alternative possibilities via synonyms and antonyms. If something is described as 'hot' is there any evidence for 'cold'? If something is 'yellow' how often is it 'blue'?
- Analyze the effect of mentioning the unusual in text. For instance for the most part fingers are dry, and it is mentioned when they are wet, because it is counter to the normal state, which is rarely noted. In that case the antonym of the commonly reported situation about instances that is in fact the more common state.
- Explore definition of semantic coverage metrics for unmapped domains. The space of 2.4K of binary predicates applied to 85K constants provides a 16 trillion combination search space, only a fraction of would be considered part of 'common knowledge'.

- Define admissibility criteria. How much evidence is necessary to consider a fact worthy of addition to the KB as commonplace?
- Determine performance relative to and in conjunction with volunteer commonsense knowledge entry projects.
- Create an interface for quick review of hypothesis by humans.

6.2 Generated Outputs

We are currently running a modified version of link grammar for bulk reading of text and generating files of parse fragments for later analysis. These fragments are processed by a database control program to queue texts, monitor their processing, and merge the fragment results. This database of fragments with fragment counts for some corpora can be one output in addition to the programs. Of course, the hypothesis sets generated in various formats (CycL assertions, ConceptNet factoids) would be made available for use and analysis by others.

7 References

[1] Deriving General World Knowledge from Texts and Taxonomies [http://www.cs.rochester.edu/~schubert/projects/world-knowledge-mining.html]

[2] Lenhart K. Schubert and Matthew Tong, *Proc. of the HLT-NAACL Workshop on Text Meaning*, May 31, 2003, Edmonton, Alberta, pp. 7-13.

[3] Lenhart K. Schubert, "Can we derive general world knowledge from texts?", *Human Language Technology Conference (HLT 2002)*, San Diego, CA, March 24-27, 2002, pp. 94-97.

[4] D. Rajesh Duthie & Rajendra Akerkar. 2002. Knowledge Representation in KRIS Using Link Grammar Parser. [PDF]

[5] Lei Zhang and Yong Yu. 2001. Learning to Generate CGs from Domain Specific Sentences. In *The Proceedings of the 9th International Conference on Conceptual Structures*, LNAI 2120, July 30-August 3, 2001, Stanford, CA, USA. [Abstract] [Full text (PDF)]

[6] Molla-Aliod,D and Hutchinson,B. 2002, Dependency-based semantic interpretation for answer extraction, In 2002 Australasian Natural Language Processing Workshop [PDF]

[7] Harsha V. Madhyastha, N. Balakrishnan, & K. R. Ramakrishnan. 2003. Event Information Extraction Using Link Grammar. *13th International WorkShop on Research Issues in Data Engineering: Multilingual Information Management (RIDE'03)*. [Abstract]

[8] Chirag Shah and Pushpak Bhattacharyya. 2003. Improving Document Vectors Representation using Semantic Links and Attributes. *International Conference on Natural Language Processing (ICON)*, Mysore, India, December 2003. [PDF]

[9] Ted Pedersen, Weidong Chen. 1995. Lexical Acquisition via Constraint Solving. In *Working* Notes of the AAAI Spring Symposium on Representation and Acquisition of Lexical Knowledge [Citeseer]

[10] Timothy Chklovski and Patrick Pantel. 2004. VERBOCEAN: Mining the Web for Fine-Grained Semantic Verb Relations. In Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-04). pp. 33-40. Barcelona, Spain. [PDF][PS]

[11] Patrick Pantel, Deepak Ravichandran and Eduard Hovy. 2004. Towards Terascale Knowledge Acquisition. In Proceedings of Conference on Computational Linguistics (COLING-04), pp. 771-777. Geneva, Switzerland. [PDF][PS]

[12] Deepak Ravichandran, Patrick Pantel, and Eduard Hovy. 2004. The Terascale Challenge. In Proceedings of KDD Workshop on Mining for and from the Semantic Web (MSW-04), pp. 1-11. Seattle, WA. [PDF][PS]

[13] Dekang Lin and Patrick Pantel. 2001. DIRT-Discovery of Inference Rules from Text. In Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD-01), pp. 323-328. San Francisco, CA. [PDF][PS]

[14] Deepak Ravichandran and Eduard Hovy. 2002. Learning Surface Text Patterns for a Question Answering system. In Proceedings of the 40th ACL conference, Philadelphia, PA. [ps, pdf]

[15] Eduard Hovy, Ulf Hermjakob, and Deepak Ravichandran. 2002. A Question/Answer Typology with Surface Text Patterns. In Proceedings of the DARPA Human Language Technology Conference (HLT). San Diego, CA. [ps, pdf]

[16] Daniel Gildea and Daniel Jurafsky, *ACL 2000*. Automatic labeling of semantic roles. [pdf]

[17] Ken Forbus, Chris Riesbeck, Lary Birnbaum, *DARPA IPTO funded research 2004*. Learning By Reading. [http://www.qrg.northwestern.edu/projects/LearningReader/index.htm][http://www.qrg.cs.northwestern.edu/projects/LearningReader/LR-Proposal-Technical.htm