GLARF and the 2nd Stage of Parsing
Combining Parsing, SRL, NE tagging, Temporal Tagging, ...

CUNY-NLP Seminar
Adam Meyers, NYU
March 25, 2011
Collaborators

- Michiko Kosaka (Formerly of Monmouth University)
- Nianwen Xue (Brandeis University)
- Shasha Liao (Google)
- Heng Ji (Rensselaer Polytechnic Institute)
- Ralph Grishman (NYU)
- Yusuke Shinyama (formerly of NYU)
- Others at NYU Proteus Group
Outline

• Approaches to Combining Annotation
• Introduction to GLARF
• Merging into one GLARF-based Theory
• Some GLARF results
• Future Work
Terminology

• Linguistic Annotation: A formal description of linguist properties of a text

• Automatic Annotation
  – Examples: parser output, NE tagging, other taggers

• Manual Annotation
  – Examples: Penn treebank, Manual NE labels, TimeBank and other “Banks”, etc.

• Transducer
  – System for automatically creating annotation
Approach 1: Annotating Multiple Linguistic Phenomena within a Single Theoretical Framework

- All annotation must share some assumptions: POS, segmentation, tokenization, headedness (if possible), constituents (if possible), etc.
- Possibly augment/modify theory over time
- Possibly revise previous annotation schemes
- Disadvantages:
  - Theory (unless revised) may overly constrain annotation of some phenomena
  - Limits input to those willing to work in that framework
- Examples:
  - Tübingen Treebank of Written German
  - Czech Dependency Treebank
  - Kyoto Corpus Treebank
  - Copenhagen Dependency Treebank
Approach 2: Merging Annotation A La Carte

- Given a set of annotations, convert each annotation into a common *physical form*, typically character offset-based XML
- Possible to incorporate the work of many different research projects (with no theory in common)
- Disadvantage: Glosses over incompatibilities between annotations (segmentation, tokenization, constituents, headedness, etc.) which may make cross-phenomenon generalizations difficult.
- Examples:
  - *Ontonotes: The 90% Solution* (Hovy, et. al. 2006)
  - *Combining Independent Syntactic and Semantic Annotation Schemes* (Verhagen, et. al. 2007)
Our Approach: Merging A la Carte Annotation while Changing it in Our Own Biased Way

• Advantages:
  – Resulting annotation fits together (like Approach 1), sharing segmentation, tokenization, headedness (if possible), constituents (if possible)
  – Incorporates annotation undertaken by several research projects which assume different theories (like Approach 2)
  – Corrects some annotation errors through conflict resolution

• Disadvantages
  – Our theory may be inappropriate for some phenomena (like Approach 1)
  – Some information may be lost/mishandled during translation

• Similar to CONLL 2008/2009 Shared Task
  – GLARF is more ambitious (a possibly more difficult shared task)
  – GLARF's Logic1 “level” that is easier to generate automatically and covers all words in the sentence.
Outline

• Approaches to Combining Annotation
• Introduction to GLARF
  – Grammatical and Logical Argument Representation Framework
• Merging into one GLARF-based Theory
• Some GLARF results
• Future Work
Why GLARF?

- Annotation of different linguistic phenomena are incompatible with each other.
- It is difficult to create NLP applications that use evidence from representations of several phenomena.
- It is difficult to correlate disparate types of linguistic evidence from corpora.
- GLARF provides one way of solving these problems.
GLARF is:

- Grammatical and Logical Argument Representation Framework
  - A framework for representing linguistic information
    - A GLARF-based theory
    - Any theory within that framework
      - In GLARF, we assume...
    - A GLARF representation of a sentence or phrase
      - In the GLARF of sentence X, ...
  - A system for automatically producing GLARF
    - Available for Download
    - Most Common Feedback from Users:
      - It is better than I expected it would be
      - This suggest that a Good tag line for GLARF might be:
        - GLARF, It's Better Than You Think it is
Introduction to GLARF

- Languages: English, Chinese, Japanese
- Typed Feature Structure: Maximal Information
- Multiple Dependency Tuples: Less info + headedness assumptions
- Produces a single-theory analysis
  - Not 100% Reversible
- GLARF System combines:
  - hand-annotation
  - automatically generated annotation
  - combination of manual/automatic annotation
- GLARF approach to merging annotation was part of the NSF-funded ULA (Unified Linguistic Annotation) project involving: The Penn Treebank, PropBank, NomBank, The Penn Discourse Treebank, TimeBank and the Pittsburg Opinion TreeBank
2 Main Purposes for GLARF

• The Second Stage of a 2-stage (LFG-style) parser – the first stage is a standard tree-bank-based parse (PTB, Chinese PTB, Kyoto Corpus)
  – Other automatic output (at least NE) is incorporated
  – Before PTB-based parsers, 2 stage parsers (Hobbs and Grishman 1976) were popular.
    • 1st stage = Syntax
    • 2nd stage = Regularized (fill gaps, transform passives, etc.)
• A merging program for manual/automatic annotation (plus additional info derived via rules, dictionary information, etc.).
• These 2 functions are indistinguishable from each other.
Example Sentence

- **Afterwards, she decided to perform the operation.**
  - Current Sentence (Sentence Number 1) Offset of first character = 29
  - Previous Sentence (Sentence Number 0) is *The doctor ran some tests*

- **PDTB (and TimeML):**
  - *Afterwards*: ARG1 = previous S, ARG2 = current S

- **PropBank**
  - *decided*: ARG0 = she, ARG1= *to perform the operation*, ARGM-TMP = *Afterwards*
  - *perform*: ARG0 = she, ARG1= *the operation*

- **NomBank**
  - *operation*: ARG0 = she, Support = *perform*

- **Penn Treebank**
  - (S (ADVP (RB Afterwards)) (, ,)
    (NP (PRP she))
    (VP (VBN decided))
    (S (VP (TO to))
      (VP (VB perform))
      (NP (DT the) (NN operation))))))) (. .))
GLARF TFS

(S (ADV (ADVP (HEAD (ADVX (HEAD (RB Afterwards 0))
(P-ARG1 (S (EC-TYPE PB) (INDEX 0+0))
(P-ARG2 (S (EC-TYPE PB) (INDEX 0))
(RELATION-TYPE AFTER))
(INDEX 1) (POINTER 0:1))))
(PUNCTUATION (, , 1))
(SBJ (NP (HEAD (PRP she 2)) (INDEX 2) (POINTER 2:1))))
(PRD (VP (HEAD (VG (HEAD decided 3))
(P-ARG0 (NP (EC-TYPE PB) (INDEX 2)))
(P-ARG1 (S (EC-TYPE PB) (INDEX 5)))
(P-ARGM-TMP (ADVP (EC-TYPE PB) (INDEX 1)))
(SEM-TENSE PAST)))
(COMP (S (L-SBJ (NP (EC-TYPE INF) (INDEX 2)))
(PRD (VP (HEAD (VG (AUX (TO to 4))
(HEAD (VB perform 5))
(P-ARG0 (NP (EC-TYPE PB) (INDEX 2)))
(P-ARG1 (NP (EC-TYPE PB) (INDEX 4)))
(INDEX 3)))
(OBJ (NP (Q-POS (DT the 6))
(HEAD (NX (HEAD (NN operation 7))
(P-SUPPORT (VG (EC-TYPE PB) (INDEX 3)))
(P-ARG0 (NP (EC-TYPE PB) (INDEX 2))))
(INDEX 4) (POINTER 6:1))
(PB-POINTER 4:1))
(POINTER 4:2) (INDEX 5))
(POINTER 3:1))
(PUNCTUATION ( . 8)) (POINTER 0:2) (TREE-NUM 1) (INDEX 0))

GLARF and the 2nd Stage of Parsing
March 25, 2011
### GLARF Dependency Tuples (Abbreviated)

<table>
<thead>
<tr>
<th>Logic1</th>
<th>Surface</th>
<th>Logic2</th>
<th>Functor</th>
<th>Off</th>
<th>POS +</th>
<th>Argument</th>
<th>Off</th>
<th>POS +</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADV</td>
<td>ADV</td>
<td>TMP</td>
<td>decided</td>
<td>45</td>
<td>VBD PAST</td>
<td>Afterwards</td>
<td>29</td>
<td>RB TMP</td>
</tr>
<tr>
<td>SBJ</td>
<td>SBJ</td>
<td>ARG0</td>
<td>decided</td>
<td>45</td>
<td>VBD PAST</td>
<td>she</td>
<td>41</td>
<td>PRP</td>
</tr>
<tr>
<td>COMP</td>
<td>COMP</td>
<td>ARG1</td>
<td>decided</td>
<td>45</td>
<td>VBD PAST</td>
<td>perform</td>
<td>56</td>
<td>VB</td>
</tr>
<tr>
<td>SBJ</td>
<td>NIL</td>
<td>ARG0</td>
<td>perform</td>
<td>56</td>
<td>VB</td>
<td>she</td>
<td>41</td>
<td>PRP</td>
</tr>
<tr>
<td>OBJ</td>
<td>OBJ</td>
<td>ARG1</td>
<td>perform</td>
<td>56</td>
<td>VB</td>
<td>operation</td>
<td>68</td>
<td>NN</td>
</tr>
<tr>
<td>AUX</td>
<td>AUX</td>
<td>NIL</td>
<td>perform</td>
<td>56</td>
<td>VB</td>
<td>to</td>
<td>53</td>
<td>TO</td>
</tr>
<tr>
<td>Q-POS</td>
<td>Q-POS</td>
<td>NIL</td>
<td>operation</td>
<td>68</td>
<td>NN</td>
<td>the</td>
<td>64</td>
<td>DT</td>
</tr>
<tr>
<td>NIL</td>
<td>NIL</td>
<td>ARG0</td>
<td>operation</td>
<td>68</td>
<td>NN</td>
<td>she</td>
<td>41</td>
<td>PRP</td>
</tr>
<tr>
<td>NIL</td>
<td>NIL</td>
<td>SUPP</td>
<td>operation</td>
<td>68</td>
<td>NN</td>
<td>perform</td>
<td>56</td>
<td>VB</td>
</tr>
<tr>
<td>NIL</td>
<td>NIL</td>
<td>ARG1</td>
<td>Afterwards</td>
<td>29</td>
<td>RB TMP</td>
<td>ran</td>
<td>11</td>
<td>VBD PAST</td>
</tr>
<tr>
<td>NIL</td>
<td>NIL</td>
<td>ARG2</td>
<td>Afterwards</td>
<td>29</td>
<td>RB TMP</td>
<td>decided</td>
<td>45</td>
<td>VBD PAST</td>
</tr>
</tbody>
</table>
Triple Dependency Graph

- **Solid Red Lines**
  - surface/L1
  - surface/L1/L2
- **Dashed Red Lines**
  - L1/L2
  - L1 only
- **Solid Blue Lines**
  - L2 only
- **Red Labels**
  - Surface/L1
- **Blue Labels**
  - L2
Logic1 and Surface Dependencies

- Surface dependencies form a tree
  - like S-Structure or C-Structure
- Logic1 dependencies form a directed acyclic graph
  - like F-Structure or D-Structure (if Empty Category = Antecedent)
- Many Logic1 and Surface Dependencies are the same
- Logic1/Surface Distinction represents syntactic regularization and gap filling
  - Passive, Relative Clause Gaps, Subjects of Infinitives, VP Deletion, etc.
- Apparent cycles are removed via Logic1/Surface distinction
  - PARENTHETICAL and RELATIVE are Surface dependencies, but their gaps represent logic1 dependencies
    - Mary, [John believed __], was a vampire
    - I want [the book [that John was reading __]]
Logic 1 Role Labels regularize relations between predicate/argument pairs

- **Red**: Predicate
- **Blue**: Logic1 OBJ
- **Yellow**: Logic1 SBJ

*They were eaten by the giant clam*

≈

*The giant clam ate them*
Logic 2 Dependencies

- Logic2 dependencies form a directed graph with cycles.
- Logic2 includes argument relations that do not fit neatly into the Surface vs Logic1 dichotomy.
- Includes semantics-based argument relations that are in complementary distribution, because the functors belong to distinct parts of speech:
  - Arguments of Verbs: PropBank
  - Arguments of (subset of) nouns: NomBank
  - Arguments of (subset of) adverbs, prepositions, coord/subord conjunctions: overt PDTB. TimeML TLINK and analogous relations with NP arguments.
Full GLARF TFS and Tuples

- More detail: morphology, semantic classes, senses from PropBank/NomBank, etc.
- Regularizes across productive syntactic regularities, distinguishing logical and surface SBJ/OBJ, e.g., passive, relative clause, etc.
- Regularizes Conjunctions, distinguishing the functor (conjunction) from the conjuncts, the latter acting like heads for purposes of tuples.
- Incorporates recognition of Named Entities, Time Expressions, Numbers, and similar phenomena.
- Handles non-headed constructions (multi-word expressions, range phrases, the-more-the-merrier constructions, etc.)
- Handles degree/comparative/superlative complements.
- Current Research in MT, time sequencing/causation relations among events, times and other elements.
- Current tuples are 25-tuples including base forms, senses, etc.
Outline

- Approaches to Combining Annotation
- Introduction to GLARF
- Merging into one GLARF-based Theory
- Some GLARF results
- Future Work
Annotation Merging with GLARF

TEXT → Combination of Human Annotators and automatic processors, e.g., segmenter, sentence-splitter parser, NE-tagger, PropBanker, etc. → Parse, NE tags, Propositions, etc.

FS to Dependency Converter ↓ Feature Structure

Dependencies as 25-Tuples

GLARFer for Language X

GLARF and the 2nd Stage of Parsing
March 25, 2011
Current (English) System

- Expected Input:
  - Sentence-split input with offsets
  - Named entity input for ACE classes (GPE, PER, FAC, LOC, ORG)
    - Automatic annotation (JET) or BBN's hand annotation of Penn Treebank
    - Syntactic Tree: PTB-parse (Charniak) or manual treebank (PTB)
- Manually created rules add additional information to all input
  - Syntactic Regularizations (Logic1)
  - Error Correction (Part of Speech, Constituent Structure)
  - Special constructions: Time/Number Expressions, Legal Cases, etc.
  - TimeML information
- Can incorporate input annotation or produce as part of GLARF system:
  - PropBank, NomBank, Overt PDTB relations
- Cascade of filters starting with parse tree and assuming strict rule ordering
Merging Considerations

• When are 2 relations part of the same “level”?
• When can 2 annotations be assumed to represent different parts of a single relation?
• What if 2 annotations assume Different Constituent Structures?
• What if merging causes undesirable dependency structures, e.g., loops?
Justification for Levels

• Surface and Logic1 levels
  – Sentence-internal regularizations based on consensus of popular theories
  – Passive, standard gap filling constructions (WH, relative, parenthetical, control, raising, VP-deletion), and a other phenomena compatible with surface/logic1 distinction
  – Distinction prevents loops in Logic1
    • Modifier relations are SURFACE if they contain gaps that modify the containing structure (parentheticals, relatives)

• The Level Logic2
  – Phenomena that don't neatly fit into Logic1
  – Phenomena are compatible with each other
    • Mergeable or in Complementary Distribution
  – Do not prevent loops or constrain to sentence-internal phenomena, etc.
Logic2: PropBank/NomBank

– PropBank includes verb alternations that may not be syntactic
  • *The pilot*/ARG0 *flew the plane*/ARG1
  • *The plane*/ARG1 *flew*.

– NomBank clusters arguments due to related verbs and other factors
  • *Rome's*/ARG1 *destruction by tourists*/ARG0
  • *John's*/ARG0 *capacity for understanding*/ARG1

– PropBank/NomBank argument relations are finer grained than Logic1
Logic2: Overt PDTB, TimeML/TLINK, Extensions

- PDTB relations and TimeML TLINK relations are not compatible with sentence-internal Surface/Logic1 relations
- When a TLINK signal and a PDTB predicate are the same, they also have the same arguments and their information is mergeable
  - *The doctor ran some tests*/ARG1 *Afterwards, she decided to perform the operation*/ARG2
  - ARG1 = EventInstance, ARG2 = relatedToEventInstance
- Other times, they are in complementary distribution
  - *The test was performed*/ARG2, *but I don't know the results*/ARG1. (ONLY PDTB)
  - *She left*/ARG1 *on Tuesday*/ARG2 (Only TimeML)
- Natural Extentions of analysis, e.g., to non-time-related PPs
  - *She left*/ARG1 *because of the problem*/ARG2
Segment/Token/Constituent Compatibility Assumptions

1. Non parse-tree units are compatible with parse-tree units if there are no “crossing boundaries”. There are 2 subcases:
   a) they correspond to parse-tree units OR
   b) they can be analyzed as evidence sub-constituents

2. When 1 is not possible, the difference is predictable by rules or heuristics
Adding Subconstituents

• Nombank
  – (NP cotton and acetate fibers) →
    (NP (NP cotton and acetate) fibers)
  – (NP a Thursday night practice) →
    (NP a (NP Thursday night) practice)

• BBN NEs
  – (NP New York-based Loews Corp.) →
    (NP (ADJP (NP New York) - based) (NP Loews Corp.))
  – (NP Republican Rudolph Guliani 's) →
    (NP (NP Republican) (NP Rudolph Guliani) 's)
Resolving Token Level Conflicts

- BBN Named Entities
  - (ADJP (NNP New) (JJ York-based)) →
    (ADJP (STEM (NP New York))
     (PUNCTUATION (HYPH -))
     (HEAD (VBN based)))
  - (NP (JJ U.S.-Japanese) (NNS relations)) →
    (NP (N-POS (NP (CONJ1 (NP U.S.))
     (CONJUNCTION (HYPH -))
     (CONJ2 (NP Japanese))))
     (HEAD (NNS relations)))

- NomBank
  - higher/ARG1 student/ARG0-test/ARG2 scores
  - (NP (A-POS (ADJP (HEAD (JJR higher 9)) (INDEX 1)))
    (N-POS (NP (N-POS (NX (HEAD (NN student 10.1)) (INDEX 2))
     (PUNCTUATION (HYPH - 10.2))
     (HEAD (NX (HEAD (NN test 10.3)) (INDEX 3))))))
    (HEAD (NX (HEAD (NN scores))
     (P-ARG1 (NP (EC-TYPE PB) (INDEX 1))))
     (P-ARG0 (NP (EC-TYPE PB) (INDEX 2))))
     (P-ARG2 (NP (EC-TYPE PB) (INDEX 3))))

- PTB ↔ Text alignment problems (I found 15 cases)
  - Predictable misalignments between text and hand-coded trees – cannot → can + not, tis → -t + is
  - Rare Errors deletions, unpredictable textual changes
Logic2: Formal Difficulties

• Cycles resulting from interactions between predicate types
  – NomBank Support Verbs and PropBank Arguments
    • NomBank: Mary/ARG0 took/Support a walk
    • PropBank: Mary/ARG0 took a walk/ARG1
  – PDTB predicates are PropBank modifiers
    • PDTB: Afterwards, she slept/ARG2.
    • PropBank: Afterwards/ARGM-TMP, she/ARG0 slept
  – These cycles seem unavoidable without a Surface2/Logic2 distinction

• Predicates embedded inside their arguments
  – The cow/ARG1, John/ARG0 said, jumped over the moon/ARG1.
  – The ARG1 is a discontinuous constituent (perhaps irrelevant for dependency representation)
  – This is unavoidable – Further discussion on next few slides
Discontinuous Arguments are interrupted by a Self-Phrase Containing the Predicate

- **Self-Phrase P’**
  - Given: Predicate P, Argument A, Adverbial P’
    - P’ is a PP, ADVP, parenthetical, etc.
    - Where P’ is a child of A, P’ is an ancestor of P
    - Convention: Listing A as the argument will be understood to mean A minus P’

- **PropBank Parentheticals**
  - (S-1 (NP The cow)
    (PRN (S (NP John)
      (VP (VBD said)
        (SBAR (-NONE- 0) (S –NONE- *T*-1))))))
    (VP jumped over the moon))
  - ARG1 of said = The cow + jumped over the moon

- **NomBank:**
  - (S (NP The legislation) (PP at their request) (VP was (VP introduced (ADVP (RB early))))))
  - ARG1 of request = The legislation + was + introduced early

- **PDTB**
  - (S (NP the company) (ADVP also) (VP disclosed what it did))
  - ARG2 of also = the company + disclosed what it did
Adjunction Rule Unites Contiguous Clausal arguments

- (S (SBAR although
  (S preliminary findings suggest X))
  (NP the latest results)
  (VP suggest Y)) →
  (S (SBAR although
  (S preliminary findings suggest X)) )
  (S (NP the latest results)
  (VP suggest Y)))
- Rule allows *although* to select whole Ss as arguments
- Preserves dependency structure
- Defensible Constituent Structure
Eliminating apparent SBJ + VP discontinuities using PTB empty categories

• If an S contains an EC bound to an external NP, an argument need not include that NP
  – John$_i$ seems (S $e_i$ to leave)
    • If ARG1 of seem is John + to leave
    • John can be deleted for purposes of LOGIC2
• If PDTB argument of a coordinate conjunction is a VP plus its SBJ, add an EC to the VP and delete the SBJ
  – (S They try to (VP (VP watch the other ropes)
    and thus
    (VP time their pulls))
  – Args of thus: They + time their pulls and They + watch the other ropes
Outline

• Approaches to Combining Annotation
• Introduction to GLARF
• Merging into one GLARF-based Theory
• Some GLARF results
• Future Work
GLARF Evaluation Description

• Evaluate exact match of 4 out of 25-tuples
  – Logic1 Role Label, +/-Transparent, Functor, Argument
  – 4 English: 46-100 sentences or 450 to 1500 tuples
    • 2 Written (WSJ and LET) and 2 Spoken (TEL and NAR)
  – CTB & KYO: 20 sentences or 400 to 600 tuples

• +/- Transparent refers to whether or not a functor is semantically empty
  – Conjunctions (and/or), Transparent Nouns (variety of birds), copulas, etc.

• F-score (F-T) also calculated ignoring transparency
  – Reduces Number Correct
    • precision = correct/output-length
    • recall = correct/answerkey-length

• Answer Keys: different for tb/parse system output
  – Genuine Ambiguity
  – Features measured not always specified as correct/incorrect
  – Sometimes more than one way to represent same concept in framework
# Evaluation on Test Corpora

- **English**: WSJ = Wall Street Journal, LET = correspondence, TEL = telephone transcripts, NAR = transcripts of narratives (LET, TEL, NAR are from OANC)
- **Chinese**: CTB = Chinese Treebank
- **Japanese**: KYO = Kyoto Corpus
- The Chinese/Japanese systems less developed compared to English
- English Evaluations: 46-100 sentences, Chinese/Japanese: 20 sentences (number of relations = 400-1500)

<table>
<thead>
<tr>
<th>Treebank</th>
<th>Prec</th>
<th>Rec</th>
<th>F</th>
<th>F-T</th>
<th>Prec</th>
<th>Rec</th>
<th>F</th>
<th>F-T</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSJ</td>
<td>83.0%</td>
<td>84.2%</td>
<td>83.6%</td>
<td>87.1%</td>
<td>80.2%</td>
<td>78.9%</td>
<td>79.5%</td>
<td>81.8%</td>
</tr>
<tr>
<td>LET</td>
<td>92.9%</td>
<td>92.3%</td>
<td>92.6%</td>
<td>93.3%</td>
<td>89.9%</td>
<td>85.9%</td>
<td>87.8%</td>
<td>87.8%</td>
</tr>
<tr>
<td>TEL</td>
<td>76.2%</td>
<td>81.2%</td>
<td>78.6%</td>
<td>82.2%</td>
<td>74.8%</td>
<td>74.5%</td>
<td>74.7%</td>
<td>77.4%</td>
</tr>
<tr>
<td>NAR</td>
<td>89.7%</td>
<td>84.0%</td>
<td>82.3%</td>
<td>84.1%</td>
<td>75.7%</td>
<td>74.7%</td>
<td>75.2%</td>
<td>76.1%</td>
</tr>
<tr>
<td>CTB</td>
<td>87.8%</td>
<td>89.1%</td>
<td>88.4%</td>
<td>88.7%</td>
<td>87.3%</td>
<td>80.4%</td>
<td>83.7%</td>
<td>83.7%</td>
</tr>
<tr>
<td>KYO</td>
<td>91.3%</td>
<td>91.0%</td>
<td>91.1%</td>
<td>91.1%</td>
<td>84.9%</td>
<td>86.2%</td>
<td>85.5%</td>
<td>87.8%</td>
</tr>
</tbody>
</table>

GLARF and the 2nd Stage of Parsing
March 25, 2011
More Evaluations

- Number of sentences: 50 En News, 46 En Blog, 53 Ch News, 40 Ja News
- Parser Output Only

<table>
<thead>
<tr>
<th>Corpus</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>En NEWS (JENAAD)</td>
<td>731/815=89.7%</td>
<td>715/812=90.0%</td>
<td>89.9%</td>
</tr>
<tr>
<td>En BLOG</td>
<td>704/844=83.4%</td>
<td>704/899=78.3%</td>
<td>80.8%</td>
</tr>
<tr>
<td>Ch Nwire</td>
<td>1031/1415=72.9%</td>
<td>1031/1352=76.3%</td>
<td>74.5%</td>
</tr>
<tr>
<td>Ja NEWS (JENAAD)</td>
<td>764/843=91.0%</td>
<td>764/840=90.6%</td>
<td>90.8%</td>
</tr>
</tbody>
</table>
2 Indirect Evaluations of Automatically Generated GLARF

- Improved Giza++ score for 2010 MT research at NYU
  - Automatically aligned English/Chinese Logic1 GLARF graphs
  - Derived mappings to reordered English text to be like Chinese
  - Lowered Alignment Error Rate from 51.9% (raw text) to 50.6% on Test Corpus (1505 sentences hand-aligned by LDC released GALE Y1 Q4)

<table>
<thead>
<tr>
<th>NYU ACE Event 12/2005 on DEV-TEST</th>
<th>VDR</th>
<th>VMD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chunker + SBJ/OBJ Heuristics</td>
<td>18.3</td>
<td>20.9</td>
</tr>
<tr>
<td>Parser + SBJ/OBJ/Passive Heur.</td>
<td>21.3</td>
<td>24.8</td>
</tr>
<tr>
<td>GLARF Logic1</td>
<td>25.8</td>
<td>31.2</td>
</tr>
<tr>
<td>GLARF Logic1 + Logic2</td>
<td>27.6</td>
<td>32.8</td>
</tr>
</tbody>
</table>
Other Work Using GLARF

- Part of 5-W System for 2009 SRA team GALE
  - NYU's part of an ensemble system for answering: Who, What, Why, Where & How questions
  - Parton, et. al. (2009), Yaman, et. al. (2009)

- Several IE systems used GLARF-based patterns
  - NYU dissertations: Zhao (2005), Shinyama (2007)

- Creating Data: CONLL 2008 & 2009 English Shared Task
  - Automatic NP-internal relations from GLARF (Prec: 83.9%-88%)
  - Automatic Split Tokenization due to hyphens (Prec: 85.5%-92.2%)
  - NomBank dependencies (filtered through GLARF)
  - Surdeanu, et. al. (2008) and Hajič, et. al. (2009)
Outline

• Approaches to Combining Annotation
• Introduction to GLARF
• Merging into one GLARF-based Theory
• Some GLARF results
• Current and Future Work
Alpha Version of GLARF for Download

• English only
• Packaged with version of Charniak parser and JET NE tagger/sentence splitter
• Intended for automatically created annotation
• Open Source, except for encrypted version of Comlex Syntax (due to LDC license)
• Free for Non-Profit and Research Use
• Commercial inquiries are welcome
• http://nlp.cs.nyu.edu/meyers/GLARF.html
What's Next?

• Work on Causation and Temporal Relations
  – Automatic system influenced by TimeML and PDTB specifications
  – Incorporates more PPs and NPs

• Further work on Machine Translation
  – Expanding using Chinese/Modified-English with MOSES translation system

• Online System
  – Functionality, Feedback and Collaboration
Summary

• GLARF is a framework for cutting out a theory from a merger of several different annotation schemata.

• Our transducer derives the analysis from:
  – Manual and/or automatic annotation
  – An ordered series of filters

• A Theoretically-biased merger provides consistent structures for use in applications
  – We believe its consistency outweighs negative impact of our biases

• GLARF has been used successfully as part of many systems at NYU.

• English GLARF has been 10 years in the making and is now available for download.
Abstract Meaning Representation (AMR)

- Annotation project that includes aspects of Parsing, PropBank, Nombank, quantifiers, etc.
- A one framework approach
- Some of GLARF is not covered by AMR
- Some of AMR is not covered by GLARF
- A mapping from GLARF to AMR probably possible
  - Hypothesis:
    - Not enough annotated data to generate good AMR
    - Requires mapping components and combining
Extra Slides

• Some Explanations
  – Multiple Correct Answers
  – Transparency Clarification
• Chinese and Japanese Examples
• Slides about contribution to CONLL
# Multiple Correct Answers

<table>
<thead>
<tr>
<th>Ambiguity</th>
<th>Corp</th>
<th>Treebank</th>
<th>Parser</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Tokenization</td>
<td>NAR</td>
<td>2 - hour, 2 - cent</td>
<td>2-hour, 2-cent</td>
</tr>
<tr>
<td>2. Prefix?</td>
<td>KYO</td>
<td>大/big + 枠/framework</td>
<td>大枠/the big picture</td>
</tr>
<tr>
<td>4. Attachment (relative)</td>
<td>LET</td>
<td>thousands [of people]</td>
<td>thousands of [people]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>who face obstacles</td>
<td>[who face obstacles]</td>
</tr>
<tr>
<td>5. Conj Scope</td>
<td>TEL</td>
<td>[pearls or [beads of some sort of necklace]]</td>
<td>[[pearls or beads] of some sort of necklace]</td>
</tr>
<tr>
<td>6. Mod ambiguity</td>
<td>KYO</td>
<td>Relative Clause</td>
<td>Adjectival Modifier</td>
</tr>
<tr>
<td></td>
<td></td>
<td>businesses that are varied</td>
<td>various businesses</td>
</tr>
<tr>
<td>7. POS ambiguity</td>
<td>CTB</td>
<td>进口五十亿/Exportation of 5 billion</td>
<td>进口 五十亿/Exported 5 billion</td>
</tr>
</tbody>
</table>
Transparency Explanation

- Transparency: conjunctions, partitives, light Vs, copulas
  - Arguments act like semantic head(s)
  - \([John\ and\ Mary]\ \text{ate} \ [a\ bag\ of\ sandwiches]\)
  - \text{red} = \text{functor of NP, } \text{yellow} = \text{semantic heads}
Chinese: 汉语中，关联词和被动句也有很明显的特点。

In Chinese, conjunctions and passive sentences also have very obvious features.

<table>
<thead>
<tr>
<th>Surf</th>
<th>L1</th>
<th>L2</th>
<th>Func</th>
<th>Arg</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADV</td>
<td>ADV</td>
<td></td>
<td>有 /have</td>
<td>中 /in</td>
</tr>
<tr>
<td>SBJ</td>
<td>SBJ</td>
<td>A0</td>
<td>有 /have</td>
<td>和 /and</td>
</tr>
<tr>
<td>ADV</td>
<td>ADV</td>
<td></td>
<td>有 /have</td>
<td>也 /also</td>
</tr>
<tr>
<td>OBJ</td>
<td>OBJ</td>
<td>A1</td>
<td>有 /have</td>
<td>特点 /features</td>
</tr>
<tr>
<td>OBJ</td>
<td>OBJ</td>
<td></td>
<td>中 /in</td>
<td>汉语 /Chinese</td>
</tr>
<tr>
<td>CONJ</td>
<td>*CONJ</td>
<td>和 /and</td>
<td>关联词 /conjunctions</td>
<td></td>
</tr>
<tr>
<td>CONJ</td>
<td>*CONJ</td>
<td>和 /and</td>
<td>被动句 /passive sentences</td>
<td></td>
</tr>
<tr>
<td>A-POS</td>
<td>A-POS</td>
<td>特点 /features</td>
<td>的 /DE</td>
<td></td>
</tr>
<tr>
<td>COMP</td>
<td>*COMP</td>
<td>的 /DE</td>
<td>明显 /obvious</td>
<td></td>
</tr>
<tr>
<td>ADV</td>
<td>ADV</td>
<td>明显 /obvious</td>
<td>很 /very</td>
<td></td>
</tr>
</tbody>
</table>
生命・財産を守ることは国家の責務だ。

It is the state's duty to protect lives and assets.
CONLL Splitting at Hyphens/Slashes 1

• Split tokens:
  – Assign POS tags
    • Automatic results for sample of 179 tokens
      – 153 correct (85.5%), 14 incorrect (7.8%), 12 unclear (6.7%)
    – Decimal token numbers
  – (VP (NP (NNP New 6)
    – (NNP York 7.1)))
  – (HYPH – 7.2)
  – (VBN based 7.3))
NP-internal Relations

• NP internal relations used for CONLL
  – Title: Mr. John Smith
  – Post-Hon: John Smith Jr. III, Inc., Ph.D., etc.
  – APPosite: John Smith, president of the U.S.
  – SUFFIX: John 's
  – Near 100% accuracy for small sample
    • 45 correct, 2 unclear

• All NP GLARF Roles
  – RELATIVE, COMP, A-POS, T-POS, Q-POS, etc.
  – 224 correct (83.9%), 32 wrong (12%), 11 unclear (4.1%)
CONLL Splitting at Hyphens/Slashes 2

- Split Segments iff:
  - COMLEX words, numbers, prefixes (from a list)
  - Required by BBN NE tags (we made a gazatteer)

- Relations from GLARF
  - Conjunction cases: Japan-U.S. agreement
  - Everything else (distinguish HMOD/HEAD)
    - GLARF distinguishes them further