Semantic Representations

School of Information Studies
Syracuse University

Outlines

- *Individual assignment #2* will be due on this incoming Sunday
- *Individual assignment #1*--- exemplary students' work can be found in Content
- Mini-talk
- Lecture
- No Lab ☺

Semantic Representation Approaches

- Some knowledge representation approaches:
  - First Order Logic
  - Frames
  - Semantic Networks
  - Case Grammar
  - Rule-Based
  - Conceptual Graphs
  - Conceptual Dependency

Approach #1: First Order Logic

- Also known as Predicate Calculus
- A symbolic language whose symbols have precisely stated meanings and uses
  - Typically express properties of entities in the world
- Example – if Joe Biden is a man, then Joe Biden is a mortal

  Man (Joe Biden) -> Mortal (Joe Biden)

FOL language

- FOL uses terms to represent objects in the real world
  - **Constants** are specific objects in the world - entities
    - Joe Biden, Blue Monkey
  - **Functions** represent concepts about objects
    - LocationOf (Blue Monkey)
      - Note the function will return some values
  - **Variables** are used to stand for any object
    - X

FOL language

- FOL uses **predicates** to state <u>relations between objects</u>
    - Note the value of a predicate is <u>True or False</u> representing facts in the world
  - "IsRestaurant" could be a predicate that when applied to an object returns True if it is a restaurant
    - *IsRestaurant (Blue Monkey)*
  - If "Serves" is a predicate taking a restaurant and a type of food as arguments, we can state that a restaurant serves a type of food
    - *Serves (Blue Monkey, JapaneseFood)*

FOL: operations and quantifiers

- FOL uses connectives *and (^)/or(v)* to combine statements
  - *Serves (Blue Monkey, JapaneseFood) ^ IsAffordable(Blue Monkey)*

- FOL uses the implication connection (=>) to mean if the first statement is true, then the second one is also true
  - *Serves(Blue Monkey, JapaneseFood) => IsRestaurant(Blue Monkey)*
    - Is this true?

FOL: operations and quantifiers

- FOL uses the existential quantifier(∃) to assert that an object with particular properties exists
  - *∃ X IsRestaurant(X) ^ Serves(X, JapaneseFood)*
    - ∃: "there exists"

and

- FOL uses the universal quantifier to assert that particular properties are true for all objects (using ∀ for the "for-all" symbol)

implies

  - *∀X IsRestaurant(X) => Serves(X, JapaneseFood)*
    - (this is definitely false because not all restaurants serve Japanese food)

Reasoning with FOL (1)

- FOL allows inference to make conclusions of new information
  - Inference rule is called "modus ponens", informally is *if-then* reasoning
    - if we know that A is true and we know that A => B is true,
    - we can conclude that B is true

Reasoning with FOL (2)

- This type of inference has efficient implementations to allow systems to reason from facts given in the semantic world or in text.
  - For example, reasoning could find answers for a QA system
    "Find me a restaurant serving Mexican food near the Hinds Hall"
  - Find the X such that

and

IsRestaurant(X) ^ Serves(X, MexicanFood) ^ Near(LocationOf(X), Hinds Hall)

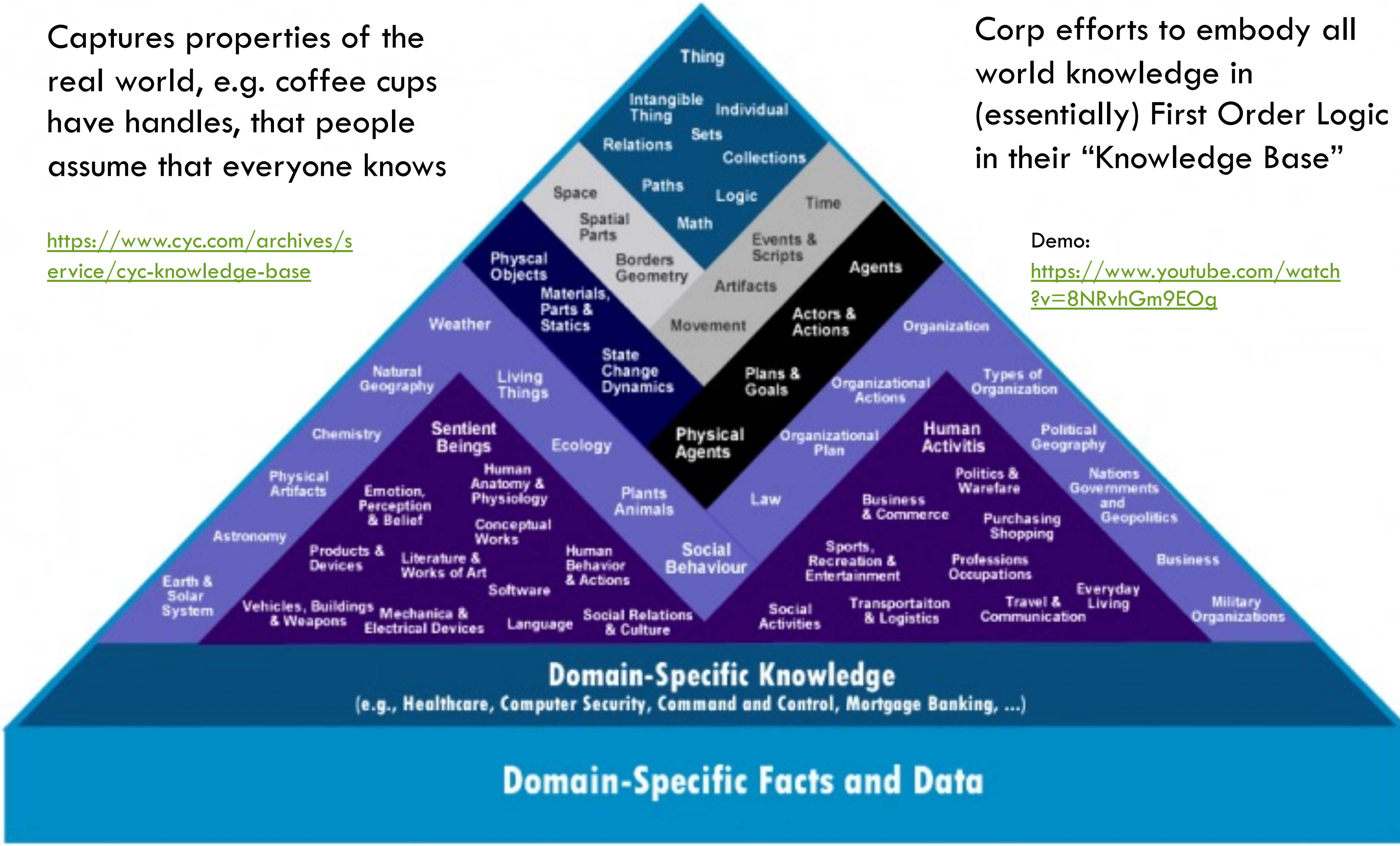
Difficulties with First Order Logic

- Problem for NLP:
  - Not everything is as clear cut as required by a formal logic
  - May not be enough “real world” predicates in the FOL system to capture semantics of text
    - This is a problem for all the semantic representations
  - Semantic systems better developed for objects and actions
    - Not as well developed to represent ideas and beliefs

Application of FOL: CYC Knowledge Base

Captures properties of the real world, e.g. coffee cups have handles, that people assume that everyone knows

https://www.cyc.com/archives/service/cyc-knowledge-base



Corp efforts to embody all world knowledge in (essentially) First Order Logic in their "Knowledge Base"

Demo:
https://www.youtube.com/watch?v=8NRvhGm9EOg

Approach #2: Frames

- Frames are a type of structured representation and are widely used for knowledge organization
- Frames group information about an entity or an event in terms of ‘slots’ and ‘fillers’
  - Each object has a frame with slots
  - One slot filled by the name of the object
  - Other slots filled with a property or relation and the value of the property or the entity that is related
- Graph structure of concepts (frames) and links (slot relations)

Example of Frames (1)

- Wikipedia Info Box is an example of a frame structure
  - Slot names are properties or relations
  - A property value is information such as a date or height
  - A relation value is another entity, which may have its own frame

Norwegian Lundehund

A Norwegian Lundehund

Other names	Norsk Lundehund Norwegian Puffin Dog	
Nicknames	Lundehund	
Country of origin	Norway	
Traits		[hide]
Weight	6–7 kilograms (13–15 lb)	
Height	30–40 centimetres (12–16 in)	
Classification / standards		[hide]
FCI	Group 5 Section 2 #265	standard
AKC	Non-Sporting	standard
UKC	Northern Breed	standard
Dog (*Canis lupus familiaris*)		

Example of Frames (2)

More formal frame systems (such as those for information extraction) require uniformity of slot names and value syntax

Name	Joe Biden
Birthdate Birthplace Children	November 20, 1942 Scranton, PA Hunter Biden, Ashley Biden, Naomi Christina Biden, Beau Biden

Reasoning with Frames can use FOL:

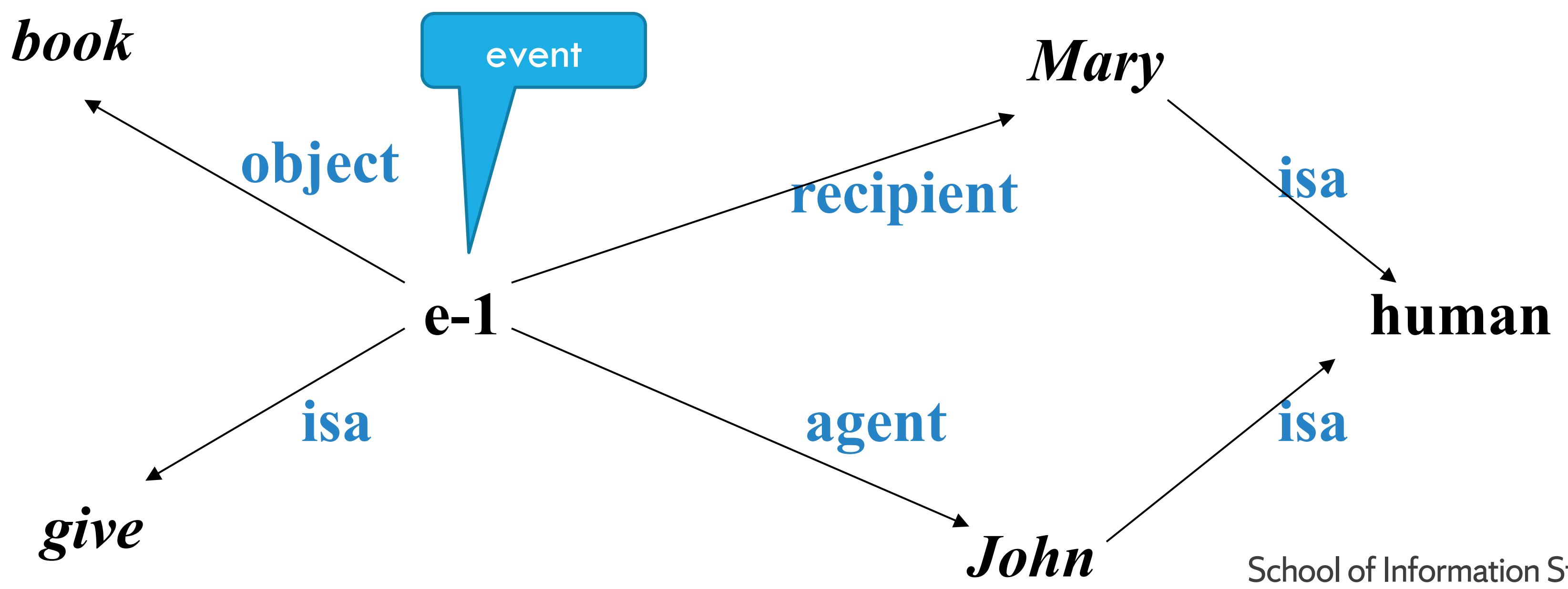
(∃X) (Name(X) = Joe Biden) ^ Birthplace(X) = Scranton, PA)

Approach #3: Semantic Networks

A network or graph of nodes joined by links where:

- nodes represent **concepts** (book, human) and **entities** (John, Mary)
- links (labelled, directed arcs) represent **relations** (e.g. isa)

John gives a book to Mary.



Applications of Semantic Representations

- **Paraphrase task**: two sentences map to the same semantic representation
- **Entailment task:** the semantics of the first sentence implies the semantics of the second under reasoning
- Used to represent entities with their properties and relations in **Information extraction and question answering systems**
- Used in **reasoning in AI systems** and **dialog system**

In-class activities:

- Find at least one application of the following semantic representations:
  - First order logic
  - Frame
  - Semantic network

Semantic Roles

School of Information Studies
Syracuse University

Semantics of events in sentences

- In a sentence, a verb and its semantic roles form a proposition; the verb can be called the **predicate** and the roles are known as **arguments**.

When Disney offered to pay Mr. Steinberg a premium for his shares, the New York investor didn't demand the company also pay a premium to other shareholders.

Example semantic roles for the verb "pay" (using verb-specific roles)
Shows which noun and prepositional phrases are related to the verb.

When [$_{\text{payer}}$ Disney] offered to [$_{\text{V}}$ pay] [$_{\text{recipient}}$ Mr. Steinberg] [$_{\text{money}}$ a premium] for [$_{\text{commodity}}$ his shares], the New York investor …

CASE Grammar

- Focuses on conceptual events
  - for each event or situation, there is a limited number of roles/cases which people or objects play in the situation
  - roles reflect ordinary human judgments about:
    - Who did the action?
    - Who / what was it done to?
    - What was it done with?
    - Where was it done?
    - What was the result?
    - When was it done?
- Case roles show **semantic structures**, not syntactic structures

Syntactic structure is **not** the same as semantic structure

- Syntactic similarities hide semantic dissimilarities
    - ***We** baked every Saturday morning.*
    - *The **pie** baked to a golden brown.*
    - *This **oven** bakes evenly.*
  - 3 subject NPs perform very different roles in regard to *bake*

- Syntactic dissimilarities hide semantic similarities
  - John$_{\text{agent}}$ broke the window$_{\text{theme}}$.
  - John$_{\text{agent}}$ broke the window$_{\text{theme}}$ with a rock$_{\text{instrument}}$.
  - The rock$_{\text{instrument}}$ broke the window$_{\text{theme}}$.
  - The window$_{\text{theme}}$ broke.
  - The window$_{\text{theme}}$ was broken by John$_{\text{agent}}$.

Cases (aka Thematic/Theta) Roles

- Some of Fillmore's original set of roles still in use as general descriptors of roles
  - **Agentive (A)**
    - the instigator of the action, an animate being
      - *John opened the door.*
      - *The door was opened by John.*
  - **Instrumental (I)**
    - the thing used to perform the action, an inanimate object
      - *The key opened the door.*
      - *John opened the door with the key.*
  - **Locative (L)**
    - the location or spatial orientation of the state or action of the verb
      - *It's windy in Chicago.*
- Other original roles not typically used

Verb-specific Roles

- **General thematic roles don't work** for many verbs and roles
  - Many general sets are proposed; not uniform agreement
  - Generalized semantic roles now often called
    - Proto roles: Proto-agent, proto-patient, etc.
- **Verb-specific roles are proposed in treebanks**
  - PropBank annotates the verbs of Penn Treebank
  - FrameNet annotates the British National Corpus
    - Uses domains of semantically similar verbs called frames.

Semantic Role Labeling (SRL) Treebanks

School of Information Studies
Syracuse University

Propbank

- **Propbank** is a corpus with annotation of semantic roles, capturing the semantic role structure of each verb sense

- Each verb sense has a frameset, listing its possible semantic roles
  - Argument notation uses numbers for the annotation
  - First sense of *accept* (accept.01)
    - Arg0: acceptor
    - Arg1: thing accepted
    - Arg2: accepted-from
    - Arg3: attribute

- The frameset roles are standard across all syntactic realizations in the corpus of that verb sense
  - Each verb has a frameset file describing the args as above
    - Example texts are also given

Example entry from Propbank

```
- <roleset vncls="58.2 58.1" name="beg" id="beg.01">
  - <roles>
    - <role n="0" descr="begger, appealer">
        <vnrole vncls="58.2" vntheta="Agent"/>
        <vnrole vncls="58.1" vntheta="Agent"/>
      </role>
    - <role n="1" descr="appealed to, begged from">
        <vnrole vncls="58.2" vntheta="Recipient"/>
        <vnrole vncls="58.1" vntheta="Patient"/>
      </role>
    - <role n="2" descr="begged/appealed for">
        <vnrole vncls="58.2" vntheta="Proposition"/>
        <vnrole vncls="58.1" vntheta="Proposition"/>
      </role>
    </roles>
  - <example name="all args">
      <text> ``Just a blind fear of the unknown is causing
        them to beg the regulators for protection.'' </text>
      <arg n="0">them</arg>
      <rel>beg</rel>
      <arg n="1">the regulators</arg>
      <arg n="2">for protection</arg>
    </example>
    <note> </note>
  </roleset>
```

Roles consistent with VerbNet

- Propbank builds on VerbNet to assign more specific roles.
- VerbNet: giving semantic roles from about 20 possible roles
  - Agent, Patient, Theme, Experiencer, etc., similar to theta roles
- Whenever possible, the Propbank argument numbering is made consistent for all verbs in a VerbNet class.
  - There is only 50% overlap between Propbank and VerbNet verbs.
- Example from frameset file for “explore”, which has a VN class:

```
<roleset id="explore.01" name="explore, discover new places or things" vncls="35.4">
<roles> <role descr="explorer" n="0">
                    <vnrole vncls="35.4" vntheta="Agent"/></role>
          <role descr="thing (place, stuff) explored" n="1">
                    <vnrole vncls="35.4" vntheta="Location"/></role>
</roles>
```

Semantic Role Notation for Propbank

- The first two numbered arguments correspond, approximately, to the core case roles:
  - Arg0 – Prototypical Agent
  - Arg1 – Prototypical Patient or Theme
  - Remaining numbered args are verb specific case roles

- Another large groups of roles are the adjunctive roles (which can be applied to any verb) and are annotated as ArgM with a suffix:

  - ArgM-LOC – location
  - ArgM-EXT – extent
  - ArgM-DIR – direction
  - ArgM-ADV – general adverbial
  - ArgM-DIS – discourse connective
  - ArgM-MOD – modal verb
  - ArgM-CAU - cause
  - ArgM-TMP - time
  - ArgM-PNC – purpose
  - ArgM-MNR - manner
  - ArgM- NEG – negation

Example for word "*Demand*"

When Disney offered to pay Mr. Steinberg a premium for his shares, the New York investor didn't demand the company also pay a premium to other shareholders.

ArgM-TMP - time

Example of Propbank annotation (demand):

[$_{\text{ArgM-TMP}}$ When Disney offered to pay Mr. Steinberg a premium for his shares], [$_{\text{Arg0}}$the New York investor] did [$_{\text{ArgM-NEG}}$ n't] [$_{\text{V}}$ demand] [$_{\text{Arg1}}$ the company also pay a premium to other shareholders].

ArgM-NEG - negation

Propbank Annotations

- **Framesets** were created by looking at sample sentences containing each verb sense.
  - ~ 4500 frames
- Corpus is primarily newswire text from Penn Treebank
  - Annotated the Wall Street Journal section, and, more recently, the "Brown" corpus
  - Verbs and semantic role annotations added to the parse trees

Propbank Annotations

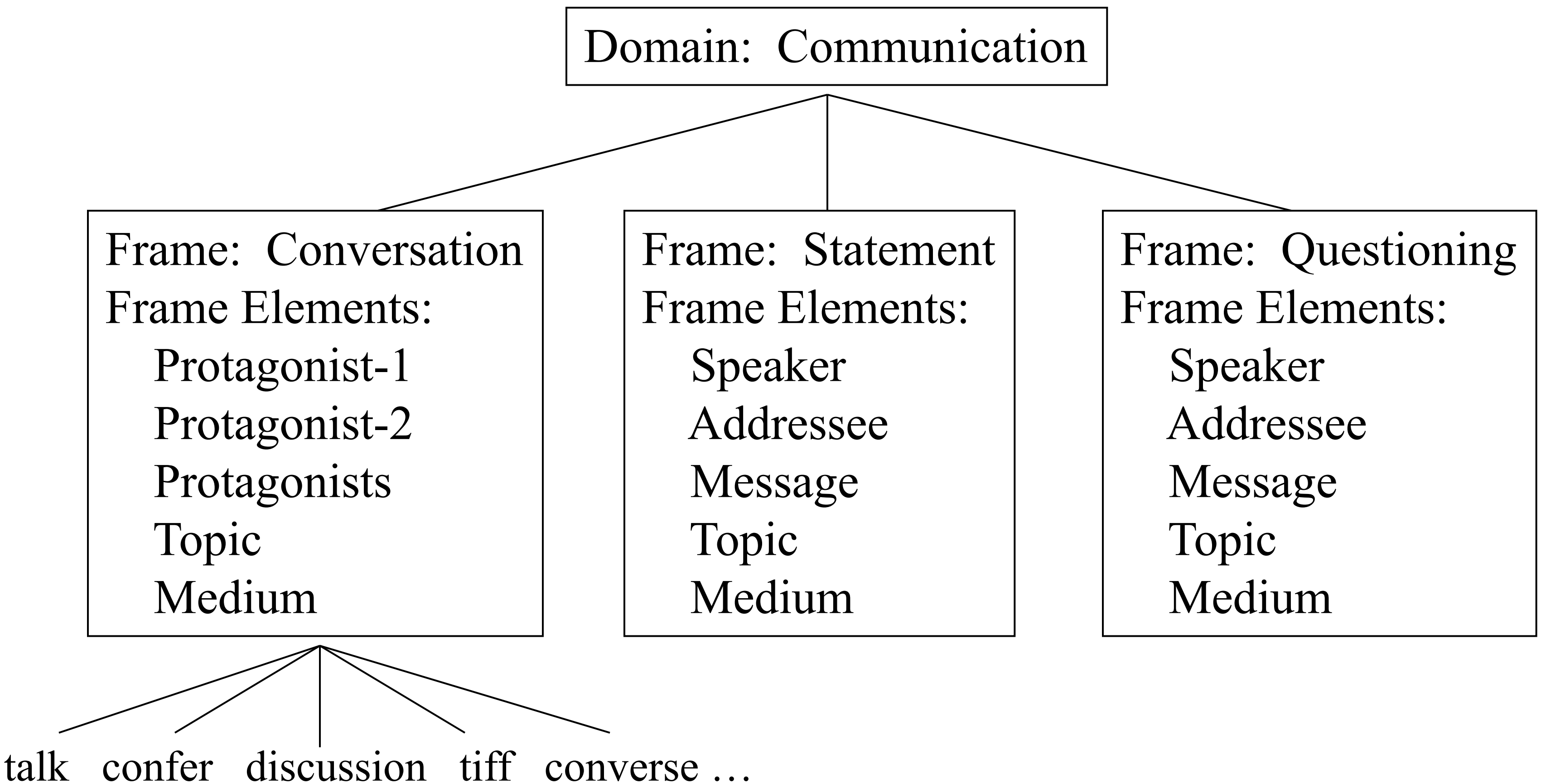
- Annotators are presented with **roleset descriptions** of a verb and the (gold) **syntactic parses** of a sentence in Treebank, and they annotate the roles of the verb.
  - Annotated on a verb-by-verb basis.
  - ~40,000 sentences were annotated
- Inter-annotator agreement
  - Identifying argument and classifying role: 99%

FrameNet

- Project at International Computer Science Institute with Charles Fillmore
  - http://framenet.icsi.berkeley.edu/
- Similar goal to document the syntactic realization of arguments of predicates in the English language
- Starts from semantic frames (e.g. Commerce) and defines frame elements (e.g. Buyer, Goods, Seller, Money)
- Annotates example sentences chosen to illustrate all possibilities
  - Recent release includes 1224 framesets
  - British National Corpus

Example of FrameNet frames

Semantic frames are related by topic domain



Comparison of FrameNet and Propbank

- FrameNet semantic roles are consistent for semantically related verbs (not just synonyms as in PropBank)
  - Commerce examples:

FrameNet annotation:
[$_{\text{Buyer}}$ Chuck] *bought* [$_{\text{Goods}}$ a car] [$_{\text{Seller}}$ from Jerry][$_{\text{Payment}}$ for $1000].
[$_{\text{Seller}}$ Jerry] *sold* [$_{\text{Goods}}$ a car] [$_{\text{Buyer}}$ to Chuck] [$_{\text{Payment}}$ for $1000].

Propbank annotation:
[$_{\text{Arg0}}$ Chuck] *bought* [$_{\text{Arg1}}$ a car] [$_{\text{Arg2}}$ from Jerry][$_{\text{Arg3}}$ for $1000].
[$_{\text{Arg0}}$ Jerry] *sold* [$_{\text{Arg1}}$ a car] [$_{\text{Arg2}}$ to Chuck] [$_{\text{Arg3}}$ for $1000].

Frame for buy:
Arg0: buyer
Arg1: thing bought
Arg2: seller
Arg3: price paid
Arg4: benefactive

Frame for sell:
Arg0: seller
Arg1: thing sold
Arg2: buyer
Arg3: price paid
Arg4: benefactive

In-class activity

Identify the semantic roles of the verbs in the following sentences based on the verb frames from PropBank

1. That settlement *represented* the first time shareholders were *granted* a major payment in a greenmail case.
2. Tensions are rising in Japan over radioactive water *leaking* into the Pacific Ocean from Japan's crippled Fukushima Daiichi nuclear plant after the 2011 earthquake.
3. He is also accused of lying under oath and of leaking information *obtained* from a wiretap he supervised.

```
<roleset id="represent.01" name="stand for, correspond"
     vncls="29.2">
- <roles>
   <role descr="item / entity taking place of other" n="0" />
   <role descr="item / entity being substituted by the other" n="1" />
     </roles>

- <roleset id="grant.01" name="request" vncls="13.3 29.5">
- <roles>
   <role descr="granter" n="0" />
   <role descr="thing granted" n="1" />
   <role descr="benefactive, granted-to" n="2" />
     </roles>

<roleset id="leak.01" name="leak, let forth water sparingly"
     vncls="43.4">
- <roles>
   <role descr="thing leaking" n="0" />
   <role descr="substance leaked" n="1" />
     </roles>

<roleset id="obtain.01" name="get" vncls="13.5.2">
- <roles>
   <role descr="receiver" n="0" />
   <role descr="thing gotten" n="1" />
   <role descr="received from" n="2" />
     </roles>
```

SRL Task

School of Information Studies
Syracuse University

Automatic SRL

- Define an algorithm that will process text and recognize roles for each verb
- Assume previous levels of NLP on text
  - Part-of-speech (POS) tagging,
  - Parse trees or dependency trees
- Machine Learning classification approaches are typical

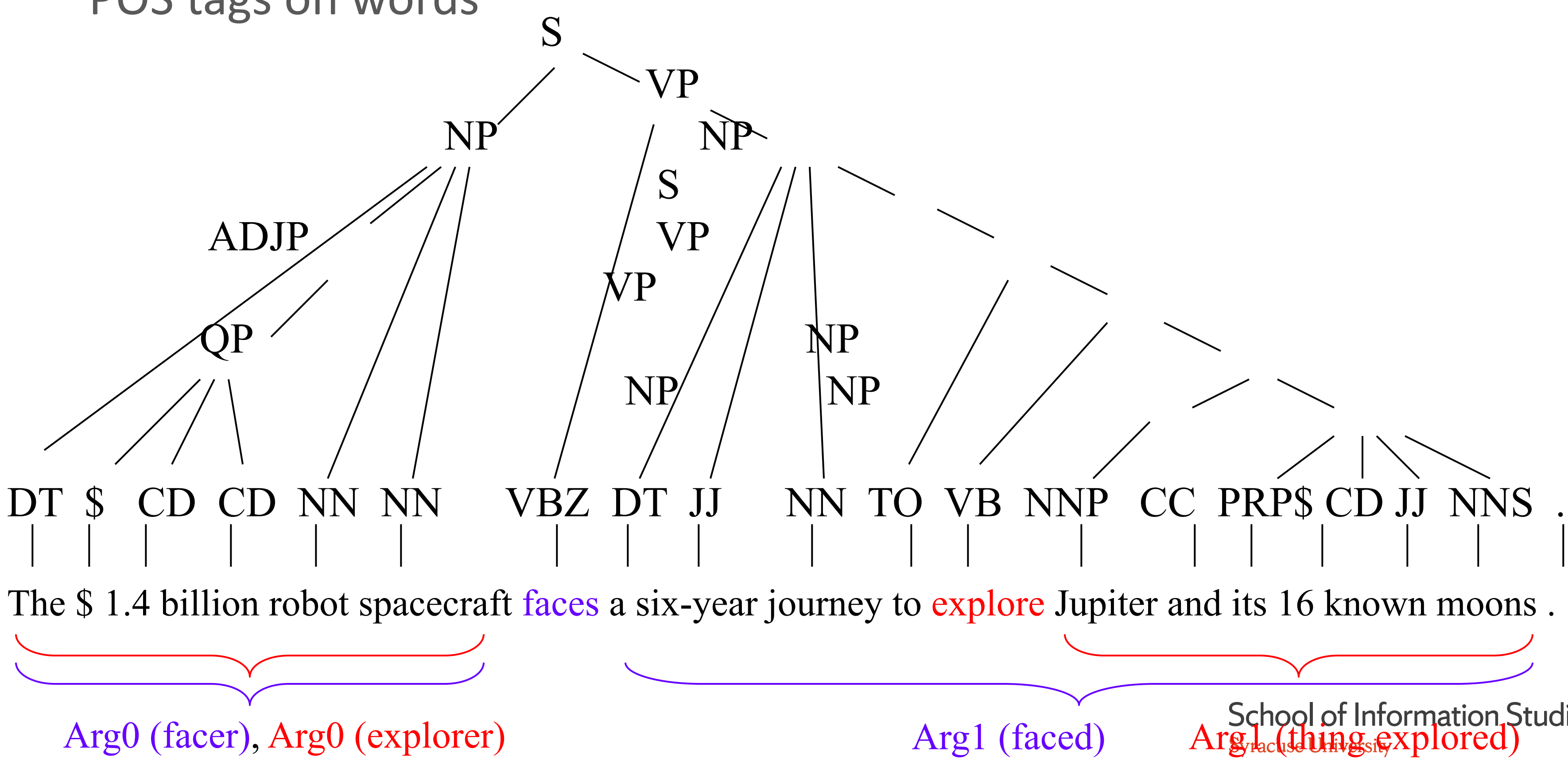
Machine Learning Approach

- Task: given a verb in a sentence, the problem is to find and label all arguments
- Reformulate as a **classification task**: For each constituent in the parse tree of the sentence, label it as to what argument, if any, it is for the verb
- For each constituent, define features of semantic roles
  - Each feature describes some aspect of a text phrase that can help determine its semantic role of a verb
    - Examples include what the verb is, POS tags, position in parse tree, etc.
- Machine Learning process:
  - Training a classifier on Treebank annotated with semantic roles
    - PropBank or FrameNet
  - Then classify syntactic phrases as to their roles

Parse Tree Constituents

Each syntactic constituent is a candidate for labeling

Define features from sentence processed into parse tree with POS tags on words

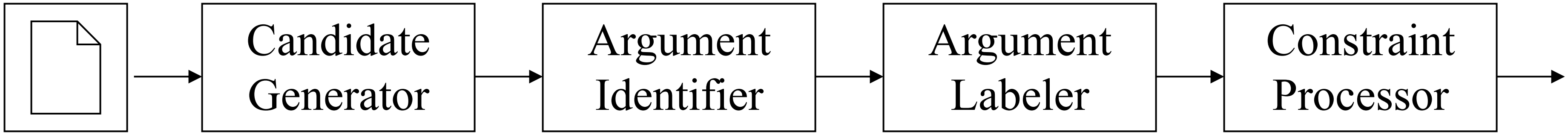


Difficulties for classification

- For each verb in a sentence, the number of constituents in the parse tree are large compared to the number of semantic roles
  - Can be hundreds of constituents eligible to be labeled a role
- What should the features be?
  - Words are typically the features for an NLP problem
    - Sometimes called bag-of-words (BOW)
  - Need more about the syntactic structure as well as other potential clues
  - Typical number of features can be up to 20,000, requiring a classification algorithm that is robust for large numbers of features

Typical architecture with 2 step classifier

- Steps of the architecture
  - Candidate Generator: filter out implausible constituents from the parse trees
  - Argument Identifier: use a machine learning classifier to decide if each of the remaining constituents is an argument to the verb
  - Argument Labeler: N binary classifiers, each producing a probability estimate of whether an argument should have that label (Arg0-Arg5, ArgM's, etc.)
  - Do some final constraint processing



Typical Argument Features

These features are defined for each constituent:

- PREDICATE: The predicate word from the training data.
  - “face” and “explore”
  - Usually stemmed or lemmatized
- PHRASE TYPE: The phrase label of the argument candidate.
  - Examples are NP for phrases, or may be POS tag if a single word
- POSITION: Whether the argument candidate is before or after the predicate.
- VOICE: Whether the predicate is in active or passive voice.
  - Passive voice is recognized if a past participle verb is preceded by a form of the verb “be” within 3 words.

SRL constraints

- Results of the labeling classifier are probabilities for each label of whether it labels that constituent

- Use these with constraints to assign a label
  - Two constituents of the same predicate cannot have the same argument label,
  - A constituent cannot have more than one label for one predicate

Current performance of SRL

- **Best English SRL** results combining parse trees or combining the parsing task with the SRL task (joint inference) are at F-measure of 80 – 82
  - Similar to other semantic tasks with performance in low 80s
- Question: **Can applications make good use of SRL?**
  - SRL tools are not as generally available as good parsing tools
  - Results not as accurate as POS tagging (~97) or parsing (~92)
  - There are systems requiring the semantics in general domain text that have used SRL to give semantic representations
    - IBM's Watson Question Answering system