# Wrap Up: Logical Representation of Sentence Meaning

JURAFSKY AND MARTIN CHAPTER 16

# Review: Desirable Properties for Meaning Representations

1. **Verifiability** – compare some meaning representation (MR) to a representation in a knowledge base (KB).
2. **Unambiguous Representations** – each ambiguous natural language meaning corresponds to a separate MR
3. **Canonical Forms** – paraphrases are collapsed to one MR
4. **Make Inferences** – draw valid conclusions based on the MR of inputs and its background knowledge in KB
5. **Match variables** – variables can be replaced by some object in the KB so an entire proposition will then match

# Model-Theoretic Semantics

A **model** allows us to bridge the gap between a formal representation and the world. The model stands in for a particular state of affairs in the world.

The **domain** of a model is the set of objects that are being represented. Each distinct thing (*person, restaurant, cuisine*) corresponds to a unique element in the domain

**Properties** of objects (like whether a restaurant is *expensive*) in a model correspond to sets of objects.

**Relations** between object (like whether a restaurant *serves* a cuisine) are are sets of tuples.

**Domain** $\mathscr{D} = \{a, b, c, d, e, f, g, h, i, j\}$

Matthew, Franco, Katie and Caroline $a, b, c, d$

Frasca, Med, Rio $e, f, g$

Italian, Mexican, Eclectic $h, i, j$

**Properties**

*Noisy* $Noisy = \{e, f, g\}$

Frasca, Med, and Rio are noisy

**Relations**

*Likes* $Likes = \{\langle a, f\rangle, \langle c, f\rangle, \langle c, g\rangle, \langle b, e\rangle, \langle d, f\rangle, \langle d, g\rangle\}$

Matthew likes the Med

Katie likes the Med and Rio

Franco likes Frasca

Caroline likes the Med and Rio

*Serves* $Serves = \{\langle f, j\rangle, \langle g, i\rangle, \langle e, h\rangle\}$

Med serves eclectic

Rio serves Mexican

Frasca serves Italian

**Domain**

Matthew, Franco, Katie and Caroline

Frasca, Med, Rio

Italian, Mexican, Eclectic

$\mathscr{D} = \{a, b, c, d, e, f, g, h, i, j\}$

$a, b, c, d$

$e, f, g$

*Katie likes Rio*

*Katie* → `c`

Rio → `g`

likes → `Likes`

**Properties**

*Noisy*

Frasca, Med, and Rio are noisy

**Relations**

*Likes*

Matthew likes the Med

Katie likes the Med and Rio

Franco likes Frasca

Caroline likes the Med and Rio

$$Likes = \{\langle a, f\rangle, \langle c, f\rangle, \langle c, g\rangle, \langle b, e\rangle, \langle d, f\rangle, \langle d, g\rangle\}$$

`<c,g>` ∈ `Likes`

so *Katie likes Rio*

is `True`

*Serves*

Med serves eclectic

Rio serves Mexican

Frasca serves Italian

# Review: First-Order Logic

FOL is a meaning representation language that satisfies the desirable qualities that we outlined. It provides a computational basis for **verifiability** and **inference**.

1. Truth conditions for logical formula involving terms and predicates like `Near(Goldie, Upenn)`
2. Truth conditions for multiple formula with logical connectives
3. Truth conditions for quantifiers

# Review: Logical Connectives

We can conjoin formula with logical connectives like **and (∧), or (∨), not** (¬), and **implies (⇒)**

Each one has a **truth table:**

| $P$ | $Q$ | $P \vee Q$ |
|---|---|---|
| $False$ | $False$ | $False$ |
| $False$ | $True$ | $True$ |
| $True$ | $False$ | $True$ |
| $True$ | $True$ | $True$ |

# Review: Quantifiers and Variables

There are two quantifiers in FOL:

1. **Existential quantifier** – ∃ – "There exists" – Is true if it holds at least one thing that matches the variable
2. **Universal quantifier** – ∀ – "For all" – Is true if it holds true for all of the things that match the variable

# Review: Existential Quantifier

*There is a restaurant that serves Mexican food near Penn*

We use the existential quantifier and a variable to represent the indefinite noun phrase “a restaurant”.

$$\exists x\ Restaurant(x) \wedge Serves(x, MexicanFood) \wedge Near((LocationOf(x), LocationOf(Penn)))$$

For this to be true there must be at least one object such that if we were to substitute it for the variable x, the resulting formula would be true.

# Review: Universal Quantifier

*All restaurants in Philly are closed.*

$$\forall x \, Restaurant(x) \land Is((LocationOf(x), Philadelphia) \Rightarrow Closed(x)$$

The ∀ operator states that for the logical formula to be true, the substitution of **any object** in the knowledge base for the **universally quantified variable** should result in a true formula.

# Lambda Notation

Lambda notation provides a way to abstract from fully specified FOL formulas by **matching variables**.

$$\lambda x.P(x)$$

Such expressions consist of the symbol λ , followed by one or more variables, followed by a FOL formula that makes use of those variables

This process of λ-reduction consists of a simple textual replacement of the λ variables and the removal of the λ.

$$\lambda x.P(x)\ (A)$$

$$P(A)$$

# Lambda Example

One λ-expression can be used within another:

$$\lambda x.\lambda y.Near(x,y)$$

Apply lambda λ-reduction to x:

$$\lambda x.\lambda y.Near(x,y)(Bacaro)$$

$$\lambda y.Near(Bacaro,y)$$

Apply lambda λ-reduction to y:

$$\lambda y.Near(Bacaro,y)(Centro)$$

$$Near(Bacaro,Centro)$$

λ -notation provides a way to incrementally gather arguments to a predicate

# Inference

A meaning representation language must support inference to add valid new propositions to a knowledge base or to determine the truth of propositions not explicitly in the KB.

**Modus ponens** provides  if-then reasoning:

$$\frac{\alpha \qquad \alpha \Rightarrow \beta}{\beta}$$

**α** is the **antecedent**

**β** is the **consequent**

# Inference

*VegetarianRestaurant*(*Leaf* )
$\forall x$ *VegetarianRestaurant*(*x*) $\Rightarrow$ *Serves*(*x*,*VegetarianFood*)

*Serves*(*Leaf* ,*VegetarianFood*)

Here, the formula *VegetarianRestaurant(Leaf )* matches the antecedent of the rule, thus allowing us to use modus ponens to conclude *Serves(Leaf ,VegetarianFood).*

# Forward Chaining

Forward chaining systems use modus ponens whenever new individual facts are added to the KB.

All applicable implication rules are found and applied, each resulting in the addition of new facts to the knowledge base.

These new propositions can also be used to fire implication rules applicable to them. The process continues until no further facts can be deduced.

# Backward Chaining

In **backward chaining**, modus ponens is run in reverse to prove specific propositions called **queries.**

1. See if the query formula is true by determining if it is present in the knowledge base.
2. If not, search for applicable implication rules present in the knowledge base.

Applicable rules are ones where the consequent matches the query formula. If there are any such rules, then the query can be proved if the antecedent of any one them can be shown to be true. This can be performed recursively.

# Backward Chaining ≠ Reasoning Backwards

Backward chaining reasons from queries to **known facts**.

Reasoning backwards goes from known consequents to **unknown antecedents.**

Reasoning backwards is an **invalid**.
Sometimes it is useful for determining what is **plausible.** Plausible reasoning from consequents to antecedents is known as **abduction.**

# Soundness and Completeness

Forward and backward chaining are both **sound** (meaning all inferences that we draw using them are **valid**). However, neither is **complete**.

The fact that they are not complete means that there are **valid inferences that cannot be found** by systems using these methods alone.

An alternative inference technique called **resolution** is but sound and complete, but it is far more computationally expensive, so most systems just use chaining.

# Representing Events

One limitation of FOL is that it only allows events with a **fixed arity** (number of arguments).

Ideally, there would be a direct mapping from a verb's subcategorization frame onto a FOL predicate.

*Leaf serves vegetarian fare*
*=*
*Serves*(*Leaf* ,*VegetarianFare*)

However, **events** are more complicated since events may involve a host of participants, props, times and locations.

# Representing Events

Choosing the correct number of arguments for the predicate representing the meaning of **eat** is tricky.

1. I ate.
2. I ate a turkey sandwich.
3. I ate a turkey sandwich at my desk.
4. I ate at my desk.
5. I ate lunch.
6. I ate a turkey sandwich for lunch.
7. I ate a turkey sandwich for lunch at my desk

# Event variables

**Event variables** to allow us to make assertions about an event without using a single predicate with a fixed number of arguments.

1. **Event predicates** are refactored with an existentially quantified variable as their **only** argument.

$$\exists e\ Eating(e)$$

2. We can introduce **additional predicates** to **represent the other information** we have about the event. These predicates take an event variable as their first argument and related FOL terms as their second argument

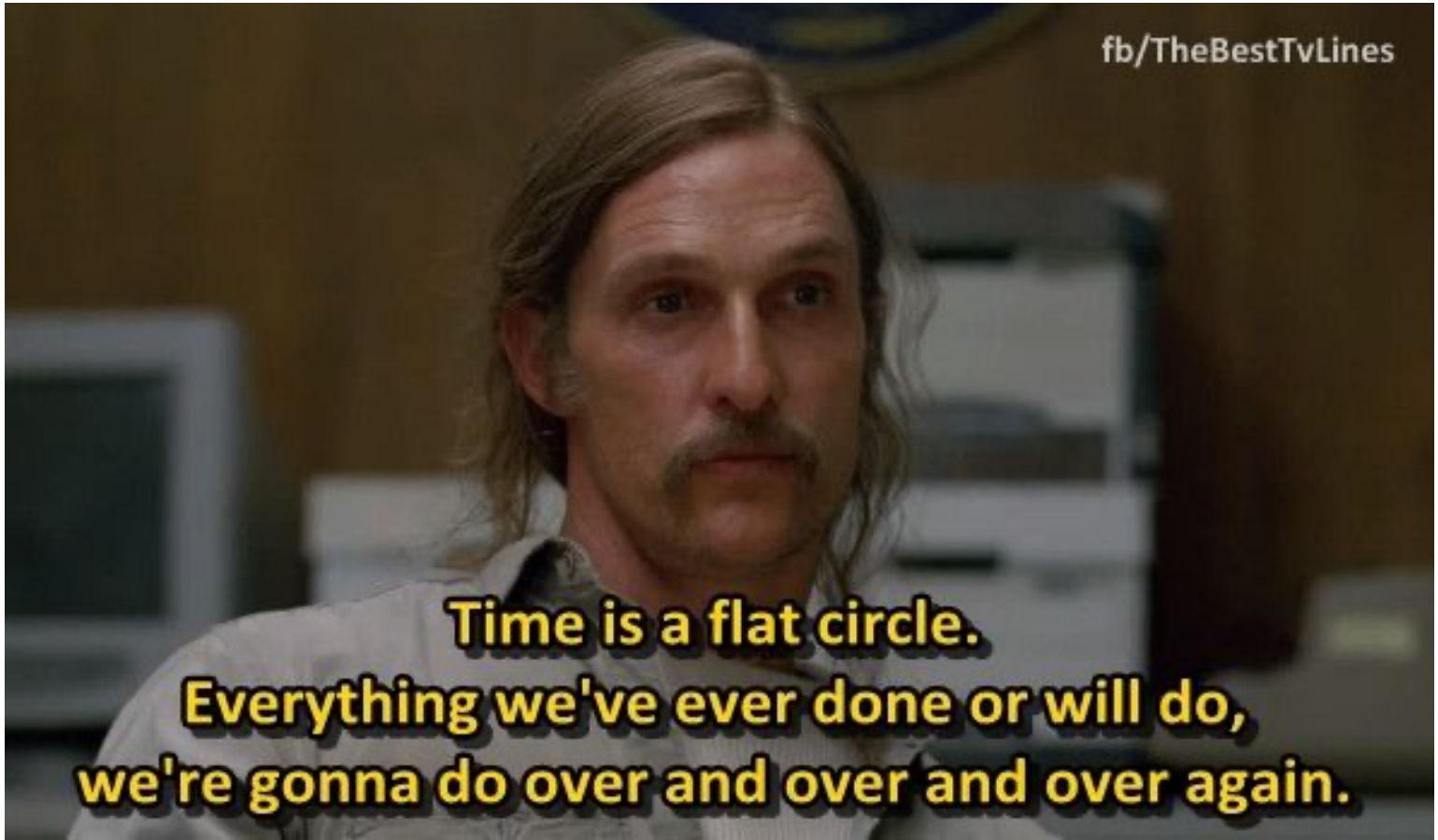
$$\exists e\ Eating(e) \wedge Eater(e, Speaker) \wedge Eaten(e, TurkeySandwich)$$

# Neo-Davidsonian Events

1. Events are captured with predicates that take a single event variable as an argument.
2. We don't need to specify a fixed number of arguments for a FOL predicate. Instead, we can glue on as many as are provided in the input.
3. No more roles are postulated than are mentioned in the input.
4. The connections across related inputs that share the same predicate are satisfied without the need for additional inference.

# Representing Time

**Temporal logic** gives us ways of representing when events occur.



# Representing Time

**Temporal logic** gives us ways of representing when events occur.

Time flows forward and that events are associated with either **points** or **intervals** in time, as on a timeline.

We can order distinct events by situating them on the timeline; one event precedes another if the flow of time leads from the first event to the second.

Represent the current moment in time as a point, and we can then easily define familiar notions of **past**, **present**, and **future**.

# Tense

These sentences only differ in the **tense** of the verb.

1. I arrived in New York.
2. I am arriving in New York.
3. I will arrive in New York.

A Neo-Davidsonian event representation would give us:

$\exists e\ \mathit{Arriving}(e) \wedge \mathit{Arriver}(e, \mathit{Speaker}) \wedge \mathit{Destination}(e, \mathit{NewYork})$

How can we represent the temporal information specified by the verb tense? *By predicating additional information about the event variable* ***e****.*

# Tense

**Past tense** – *I arrived in NY.*

$$\exists e\ \mathit{Arriving}(e) \wedge \mathit{Arriver}(e, \mathit{Speaker}) \wedge \mathit{Destination}(e, \mathit{NewYork}) \wedge \mathbf{\mathit{IntervalOf}}(e, i) \wedge \mathbf{\mathit{EndPoint}}(i, n) \wedge \mathbf{\mathit{Precedes}}(n, \mathit{Now})$$

**Present tense** – *I am arriving in NY.*

$$\exists e\ \mathit{Arriving}(e) \wedge \mathit{Arriver}(e, \mathit{Speaker}) \wedge \mathit{Destination}(e, \mathit{NewYork}) \wedge \mathbf{\mathit{IntervalOf}}(e, i) \wedge \mathbf{\mathit{MemberOf}}(i, \mathit{Now})$$

**Future tense** – *I will arrive in NY.*

$$\exists e\ \mathit{Arriving}(e) \wedge \mathit{Arriver}(e, \mathit{Speaker}) \wedge \mathit{Destination}(e, \mathit{NewYork}) \wedge \mathbf{\mathit{IntervalOf}}(e, i) \wedge \mathbf{\mathit{EndPoint}}(i, n) \wedge \mathbf{\mathit{Precedes}}(\mathit{Now}, n)$$

# Temporal Expressions

| Absolute | Relative | Durations |
|---|---|---|
| April 24, 1916 | yesterday | four hours |
| The summer of '77 | next semester | three weeks |
| 10:15 AM | two weeks from yesterday | six days |
| The 3rd quarter of 2006 | last quarter | the last three quarters |

Lexical triggers for temporal expressions:

| Category | Examples |
|---|---|
| Noun | *morning, noon, night, winter, dusk, dawn* |
| Proper Noun | *January, Monday, Ides, Easter, Rosh Hashana, Ramadan, Tet* |
| Adjective | *recent, past, annual, former* |
| Adverb | *hourly, daily, monthly, yearly* |

Temporal normalization

- mapping a temporal expression to either normalization a specific point in time or to a duration

# Other Temporal Phenomena

**Order** – a pair of events can *precede* or *follow* each other, they or *overlap* and may have different onset times.

**Duration** – we might want to model how long an event typically takes

- *I went to college to get a degree*
- *I went to the salon to get a haircut*

**Frequency / periodicity**

- *Presidential elections take place ever 4 years*

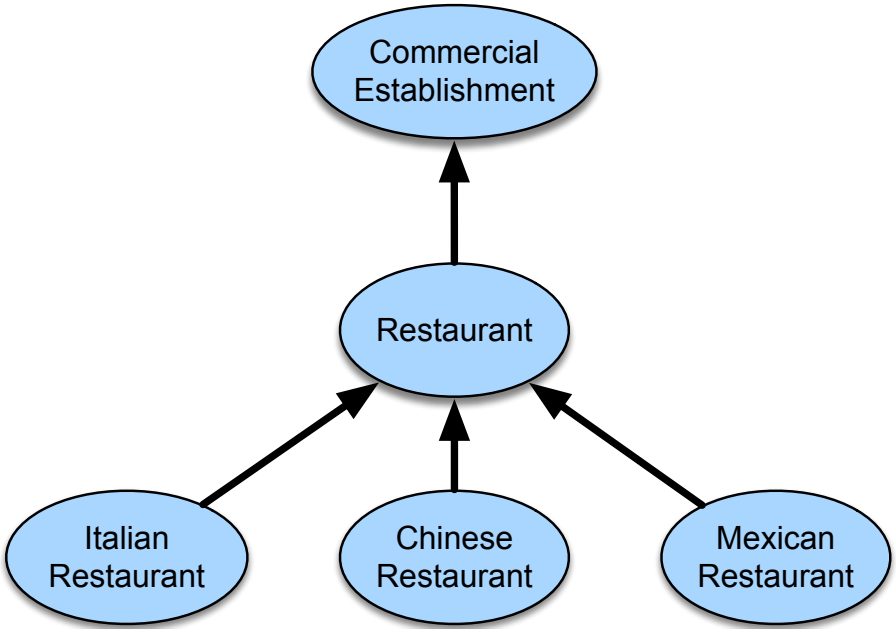
**Stationarity** – permanent versus impermanent events

- *The Statue of Liberty is in New York*
- *The Coronavirus is in New York*

# Description Logic

Description Logics use **ontologies** to encode relationships between type of entities, and allow us to specify rules associated with certain types like

$$\text{ItalianRestaurant} \sqsubseteq \text{Restaurant} \sqcap \exists \text{hasCuisine.ItalianCuisine}$$



# Conclusion

Formal meaning representations that capture the meaning-related content of linguistic inputs. These representations are intended to bridge the gap from language to common-sense knowledge of the world.

Meaning representation languages like FOL specify syntax and semantics of these representations. Many others have been used in NLP and AI.

Important elements of semantic representation including states and events can be captured in FOL.

Modelling temporal phenomena is an important challenge.

# Information Extraction

JURAFSKY AND MARTIN CHAPTER 18

# Information Extraction (IE)

- Information extraction (IE), turns the unstructured text information into structured data
- Populate a relational database to enable further processing, support queries

# Template Filling

Citing high fuel prices, United Airlines said Friday it has increased fares by $6 per round trip on flights to some cities also served by lower cost carriers. American Airlines, a unit of AMR Corp., immediately matched the move, spokesman Tim Wagner said. United, a unit of UAL Corp., said the increase took effect Thursday and applies to most routes where it competes against discount carriers, such as Chicago to Dallas and Denver to San Francisco.

| | | |
|---|---|---|
| FARE-RAISE ATTEMPT: | LEAD AIRLINE: | UNITED AIRLINES |
| | AMOUNT: | $6 |
| | EFFECTIVE DATE: | 2006-10-26 |
| | FOLLOWER: | AMERICAN AIRLINES |

# Steps in IE

- NER and co-reference resolution
- **relation** extraction
  - spouse-of, child-of, employer-of, partof, membership-in, located-in
- event extraction
- temporal expression normalization
- template filling

# Named Entity Recognition

Citing high fuel prices, [ORG **United Airlines**] said [TIME **Friday**] it has increased fares by [MONEY **$6**] per round trip on flights to some cities also served by lower-cost carriers. [ORG **American Airlines**], a unit of [ORG **AMR Corp.**], immediately matched the move, spokesman [PER **Tim Wagner**] said. [ORG **United**], a unit of [ORG **UAL Corp.**], said the increase took effect [TIME **Thursday**] and applies to most routes where it competes against discount carriers, such as [LOC **Chicago**] to [LOC **Dallas**] and [LOC **Denver**] to [LOC **San Francisco**].

| Type | Tag | Sample Categories | Example sentences |
|---|---|---|---|
| People | PER | people, characters | **Turing** is a giant of computer science. |
| Organization | ORG | companies, sports teams | The **IPCC** warned about the cyclone. |
| Location | LOC | regions, mountains, seas | The **Mt. Sanitas** loop is in **Sunshine Canyon**. |
| Geo-Political Entity | GPE | countries, states, provinces | **Palo Alto** is raising the fees for parking. |
| Facility | FAC | bridges, buildings, airports | Consider the **Golden Gate Bridge**. |
| Vehicles | VEH | planes, trains, automobiles | It was a classic **Ford Falcon**. |

# Named Entity Recognition

- Find spans of text that constitute proper names
  - Ambiguity of segmentation
    - What's an entity and what isn't
    - Where the boundaries are
- Classify the type of the entity
  - Type ambiguity

# Category Ambiguity in NER

| Name | Possible Categories |
|---|---|
| *Washington* | Person, Location, Political Entity, Organization, Vehicle |
| *Downing St.* | Location, Organization |
| *IRA* | Person, Organization, Monetary Instrument |
| *Louis Vuitton* | Person, Organization, Commercial Product |

Examples of type ambiguities in the use of the name Washington

[$_{PER}$ Washington] was born into slavery on the farm of James Burroughs.
[$_{ORG}$ Washington] went up 2 games to 1 in the four-game series.
Blair arrived in [$_{LOC}$ Washington] for what may well be his last state visit.
In June, [$_{GPE}$ Washington] passed a primary seatbelt law.
The [$_{VEH}$ Washington] had proved to be a leaky ship, every passage I made...

# NER Algorithms & Evaluations

- Word to word sequence labelling
  - Capture boundary and type
- Feature based
- Neural

- Precision, Recall, F1
  - Segmentation component can cause problems

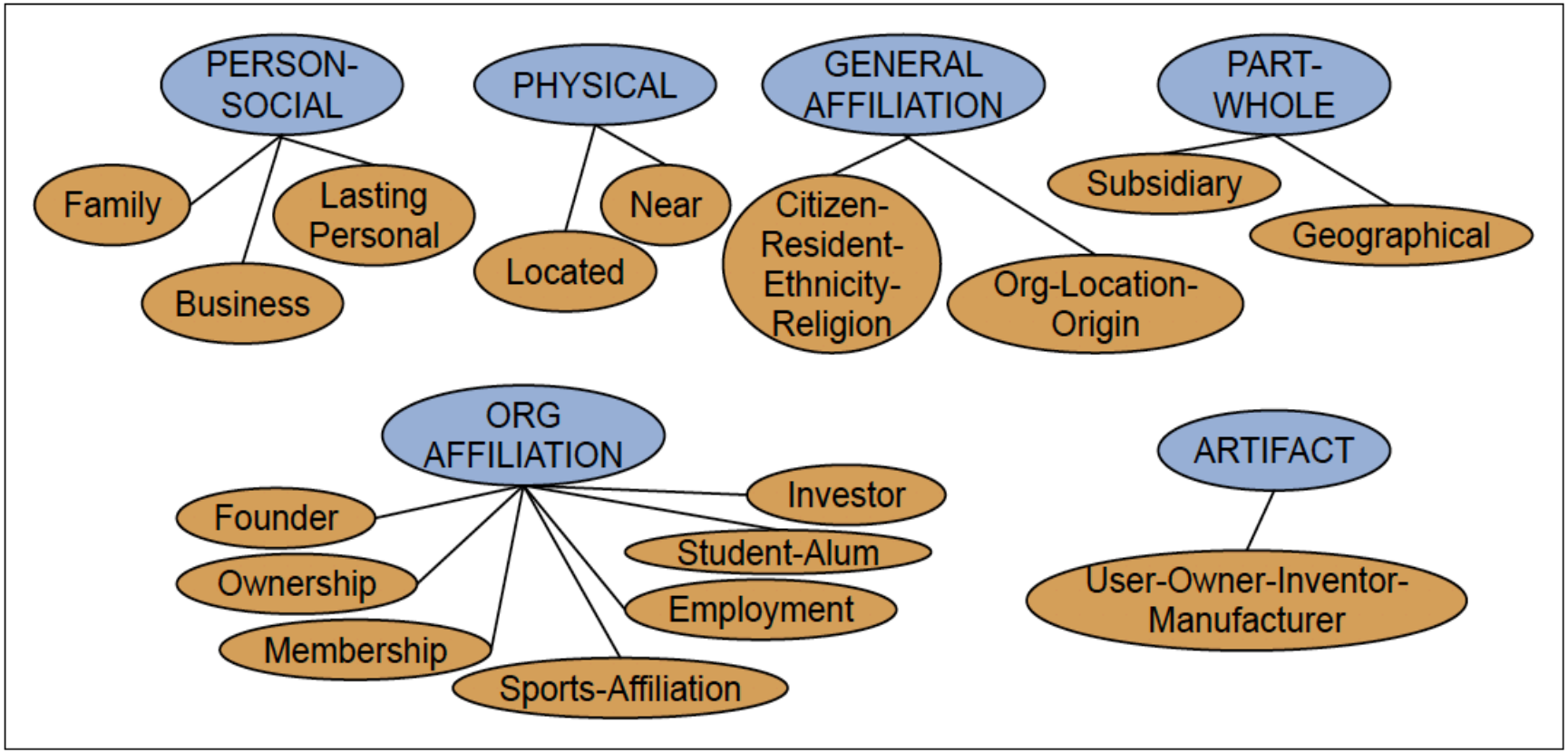
# Relation Extraction

- A relation consists of a set of ordered tuples over elements of a domain
- The domain elements corresponds to the named entities

| | |
|---|---|
| **Domain** | $\mathscr{D} = \{a, b, c, d, e, f, g, h, i\}$ |
| United, UAL, American Airlines, AMR | $a, b, c, d$ |
| Tim Wagner | $e$ |
| Chicago, Dallas, Denver, and San Francisco | $f, g, h, i$ |
| **Classes** | |
| United, UAL, American, and AMR are organizations | $Org = \{a, b, c, d\}$ |
| Tim Wagner is a person | $Pers = \{e\}$ |
| Chicago, Dallas, Denver, and San Francisco are places | $Loc = \{f, g, h, i\}$ |
| **Relations** | |
| United is a unit of UAL<br>American is a unit of AMR | $PartOf = \{\langle a, b\rangle, \langle c, d\rangle\}$ |
| Tim Wagner works for American Airlines | $OrgAff = \{\langle c, e\rangle\}$ |
| United serves Chicago, Dallas, Denver, and San Francisco | $Serves = \{\langle a, f\rangle, \langle a, g\rangle, \langle a, h\rangle, \langle a, i\rangle\}$ |

Citing high fuel prices, [ORG **United Airlines**] said [TIME **Friday**] it has increased fares by [MONEY **$6**] per round trip on flights to some cities also served by lower-cost carriers. [ORG **American Airlines**], a unit of [ORG **AMR Corp.**], immediately matched the move, spokesman [PER **Tim Wagner**] said. [ORG **United**], a unit of [ORG **UAL Corp.**], said the increase took effect [TIME **Thursday**] and applies to most routes where it competes against discount carriers, such as [LOC **Chicago**] to [LOC **Dallas**] and [LOC **Denver**] to [LOC **San Francisco**].

Tim Wagner **[is a spokesman]** for American Airlines.
United **[is a unit of]** UAL Corp.
American **[is a unit of]** AMR.




| Relations | Types | Examples |
|---|---|---|
| Physical-Located | PER-GPE | **He** was in **Tennessee** |
| Part-Whole-Subsidiary | ORG-ORG | **XYZ**, the parent company of **ABC** |
| Person-Social-Family | PER-PER | **Yoko**'s husband **John** |
| Org-AFF-Founder | PER-ORG | **Steve Jobs**, co-founder of **Apple**... |

# Unified Medical Language System

| Entity | Relation | Entity |
|---|---|---|
| Injury | disrupts | Physiological Function |
| Bodily Location | location-of | Biologic Function |
| Anatomical Structure | part-of | Organism |
| Pharmacologic Substance | causes | Pathological Function |
| Pharmacologic Substance | treats | Pathologic Function |

Given a medical sentence like:

Doppler echocardiography can be used to diagnose left anterior descending artery stenosis in patients with type 2 diabetes

Extract the UMLS relation:

Echocardiography (Doppler) **Diagnoses** Acquired stenosis

# Wikipedia info boxes

**Barack Obama**

44th President of the United States

| | |
|---|---|
| **In office** | January 20, 2009 – January 20, 2017 |
| **Vice President** | Joe Biden |
| **Preceded by** | George W. Bush |
| **Succeeded by** | Donald Trump |
| **United States Senator from Illinois** | |
| **In office** | January 3, 2005 – November 16, 2008 |
| **Preceded by** | Peter Fitzgerald |
| **Succeeded by** | Roland Burris |
| **Member of the Illinois Senate from the 13th district** | |
| **In office** | January 8, 1997 – November 4, 2004 |
| **Preceded by** | Alice Palmer |
| **Succeeded by** | Kwame Raoul |
| **Personal details** | |
| **Born** | Barack Hussein Obama II<br>August 4, 1961 (age 58)<br>Honolulu, Hawaii, U.S. |
| **Political party** | Democratic |

# RDF Triples

- Resource Description Framework
- **RDF triple** is a tuple of
  - Entity-relation-entity, aka
  - Subject-predicate-object expression
- **Subject**: University of Pennsylvania
- **Predicate**: location
- **Object**: Philadelphia, PA

## Zachary G. Ives

FOLLOW

Professor of Computer and Information Science, University of Pennsylvania
Verified email at cis.upenn.edu - Homepage

Databases  data integration  distributed systems  web data management

| TITLE | CITED BY | YEAR |
|---|---|---|
| Dbpedia: A nucleus for a web of open data<br>S Auer, C Bizer, G Kobilarov, J Lehmann, R Cyganiak, Z Ives<br>The semantic web, 722-735 | 4119 | 2007 |

DBpedia is a community effort to extract structured information from Wikipedia and to make this information available on the Web. DBpedia allows you to ask sophisticated queries against datasets derived from Wikipedia and to link other datasets on the Web to Wikipedia data. We describe the extraction of the DBpedia datasets, and how the resulting information is published on the Web for human-and machine-consumption. We describe some emerging applications from the DBpedia community and show how website authors can facilitate DBpedia content within their sites. Finally, we present the current status of interlinking DBpedia with other open datasets on the Web and outline how DBpedia could serve as a nucleus for an emerging Web of open data.

# Freebase, WordNet, other ontologies

- Freebase relations:
  - people/person/nationality
  - location/location/contains
  - people/person/place-of-birth
  - biology/organism classification
- WordNet relations:
  - is-a, instance-of
  - hypernyms/hyponyms
  - Giraffe is-a ruminant is-a ungulate is-a mammal is-a vertebrate is-a animal…

# Strategies for relation extraction

- Hand-written patterns
- Supervised machine learning
- Semi-supervised machine learning
  - Bootstrapping
  - Distant supervision
- Unsupervised machine learning

# Hearst Patterns

Agar is a substance prepared from a mixture of red algae, such as Gelidium, for laboratory or industrial use.

She suggests that the following lexico-syntactic pattern

$$NP_0 \text{ such as } NP_1 \{,NP_2 \ldots, (\text{and}|\text{or}) NP_i\}, \; i \geq 1$$

implies the following semantics

$$\forall NP_i, i \geq 1, hyponym(Np_i, NP_0$$

allowing us to infer

*hyponym(*Gelidium, red algae*)*

| | |
|---|---|
| NP {, NP}* {,} (and\|or) other $NP_H$ | temples, treasuries, and other important civic buildings |
| $NP_H$ such as {NP,}* {(or\|and)} NP | red algae such as Gelidium |
| such $NP_H$ as {NP,}* {(or\|and)} NP | such authors as Herrick, Goldsmith, and Shakespeare |
| $NP_H$ {,} including {NP,}* {(or\|and)} NP | common-law countries, including Canada and England |
| $NP_H$ {,} especially {NP}* {(or\|and)} NP | European countries, especially France, England, and Spain |

# Machine learning techniques

- **Supervised**: training corpus annotated with manually annotated with fixed set of relations and entities
- **Semi-supervised**: high-precision seed patters, or seed tuples, are used to bootstrap more examples
- **Distant supervision**: start with a huge number of seeds, learn noisy pattern fields (e.g. 100k examples of `birth-place-of` from infoboxes, help learn the corresponding text patterns)

# Semisupervised Relation Extraction

**function** BOOTSTRAP(*Relation R*) **returns** *new relation tuples*

*tuples* ← Gather a set of seed tuples that have relation *R*
**iterate**
*sentences* ← find sentences that contain entities in *tuples*
*patterns* ← generalize the context between and around entities in *sentences*
*newpairs* ← use *patterns* to grep for more tuples
*newpairs* ← *newpairs* with high confidence
*tuples* ← *tuples* + *newpairs*
**return** tuples

Bootstrapping proceeds by taking the entities in the seed pair, and then finding sentences (on the web, or whatever dataset we are using) that contain both entities.

# Example

Task: Create airline/hub pairs

Seed: **Ryanair** has a hub at **Charleroi**

use this seed fact to discover new patterns by finding other mentions of this relation in our corpus

Sentences found:

Budget airline **Ryanair**, which uses **Charleroi** as a hub, scrapped all weekend flights out of the airport.

All flights in and out of **Ryanair**'s Belgian hub at **Charleroi** airport were grounded on Friday...

A spokesman at **Charleroi**, a main hub for **Ryanair**, estimated that 8000 passengers had already been affected.

Patterns extracted:

/ [ORG], which uses [LOC] as a hub /

/ [ORG]'s hub at [LOC] /

/ [LOC] a main hub for [ORG] /

# Distant Supervision

**function** DISTANT SUPERVISION(*Database D, Text T*) **returns** *relation classifier C*

**foreach** relation *R*
  **foreach** tuple (*e1*,*e2*) of entities with relation *R* in *D*
    *sentences* ← Sentences in *T* that contain *e1* and *e2*
    *f* ← Frequent features in *sentences*
    *observations* ← observations + new training tuple (*e1*, *e2*, *f*, *R*)
  *C* ← Train supervised classifier on *observations*
  **return** *C*

**R:** place of birth

**(e1, e2):** <Edwin Hubble, Marshfield>, <Albert Einstein,Ulm>, etc

**Sentences:** Hubble was born in Marshfield; Einstein, born (1879), Ulm; Hubble's birthplace in Marshfield..., etc

# Open IE

Unsupervised relation extraction

Find all strings of words that satisfy the tripe relation.

United has a hub in Chicago, which is the headquarters of United Continental Holdings.

r1: <United, has a hub in, Chicago>

r2: <Chicago, is the headquarters of, United Continental Holdings>

# Evaluation of Relation Extraction

- Supervised
  - Test sets with human annotated, gold-standard relations and computing precision, recall, and F-measure
- Semi and Unsupervised
  - Human evaluations
  - Compute precision at different levels of recall

# Temporal Expression Extraction

| Absolute | Relative | Durations |
|---|---|---|
| April 24, 1916 | yesterday | four hours |
| The summer of '77 | next semester | three weeks |
| 10:15 AM | two weeks from yesterday | six days |
| The 3rd quarter of 2006 | last quarter | the last three quarters |

Lexical triggers for temporal expressions:

| Category | Examples |
|---|---|
| Noun | *morning, noon, night, winter, dusk, dawn* |
| Proper Noun | *January, Monday, Ides, Easter, Rosh Hashana, Ramadan, Tet* |
| Adjective | *recent, past, annual, former* |
| Adverb | *hourly, daily, monthly, yearly* |

- Temporal expression recognition
- Temporal normalization
  - mapping a temporal expression to either normalization a specific point in time or to a duration

# Event Extraction

The task of event extraction is to identify mentions of **events** in texts. An event mention is **any expression denoting an event or state** that can be assigned to a **point in time** or an **interval in time**.

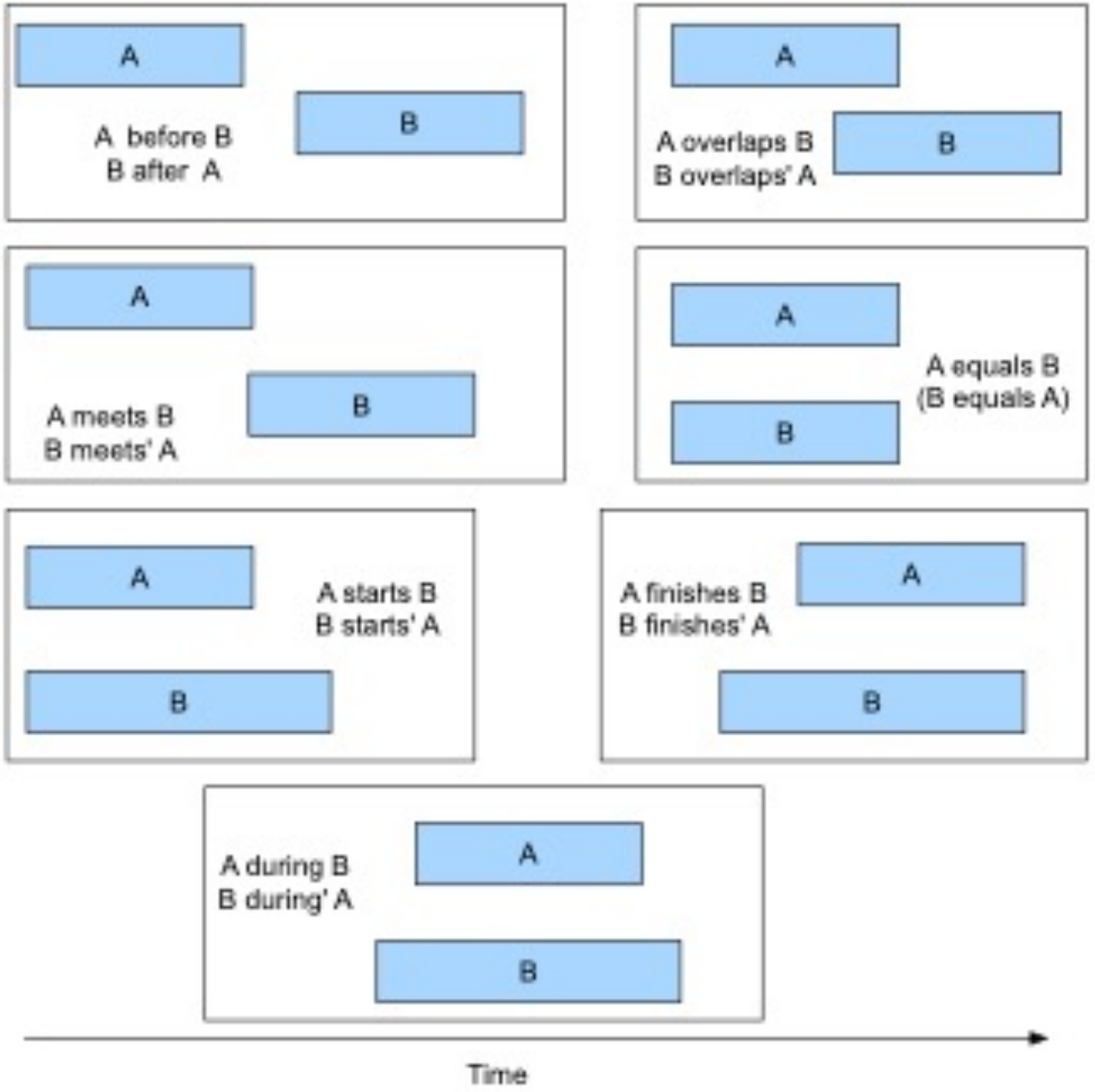
In English events are typically expressed **verbs** like exploded. However, there are also some **nouns**, like explosion, that denote an event.

Some “light verbs” like *make*, *take*, and *have* often do not denote events. Instead, the event is often expressed by the nominal direct object (*took a **flight***).

# Event Extraction

[EVENT Citing] high fuel prices, United Airlines [EVENT said] Friday it has [EVENT increased] fares by $6 per round trip on flights to some cities also served by lower-cost carriers. American Airlines, a unit of AMR Corp., immediately [EVENT matched] [EVENT the move], spokesman Tim Wagner [EVENT said]. United, a unit of UAL Corp., [EVENT said] [EVENT the increase] took effect Thursday and [EVENT applies] to most routes where it [EVENT competes] against discount carriers, such as Chicago to Dallas and Denver to San Francisco.

Events can be classified as **actions**, **states**, **reporting events, perception events**, etc. The aspect, tense, and modality of each event also needs to be extracted.


A
B
A before B
B after A
A
B
A overlaps B
B overlaps' A
A
B
A meets B
B meets' A
A
B
A equals B
(B equals A)
A
B
A starts B
B starts' A
A
B
A finishes B
B finishes' A
A
B
A during B
B during' A
Time

# Temporal ordering of events

Delta Air Lines earnings soared 33% to a record in the fiscal first quarter, bucking the industry trend toward declining profits.

- Soaring$_{e1}$ is **included in** the fiscal first quarter$_{t58}$
- Soaring$_{e1}$ is **simultaneous with** the bucking$_{e3}$
- Declining$_{e4}$ **includes** soaring$_{e1}$

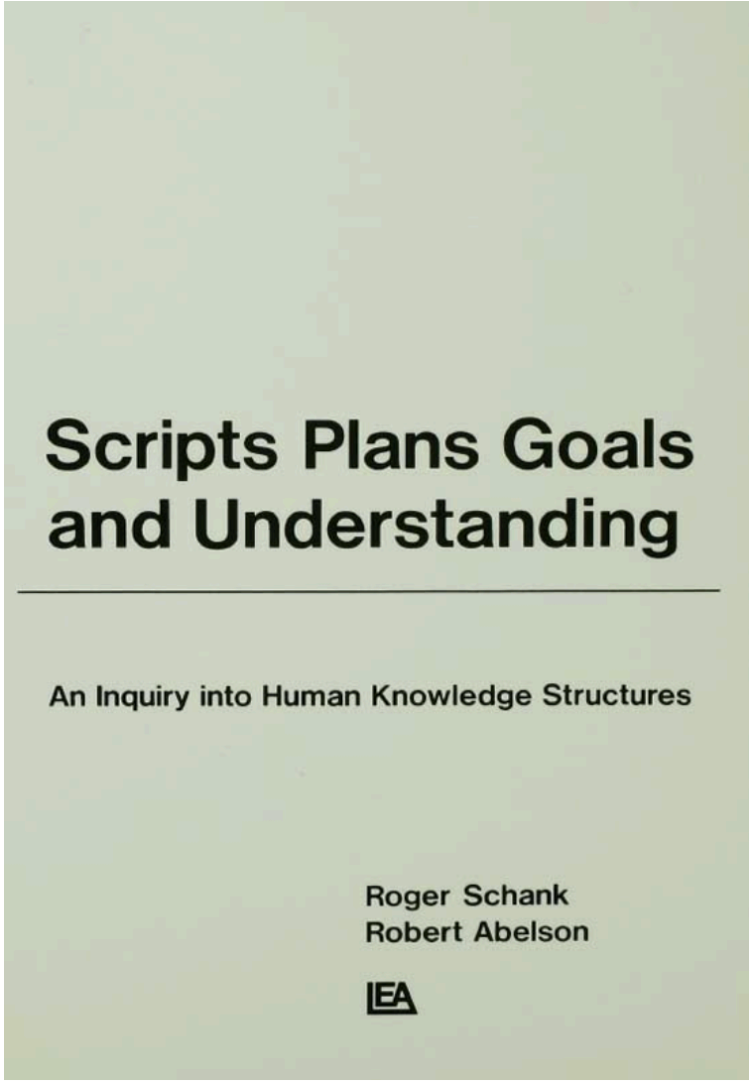
# Scripts

How do people organize all the knowledge they must have in order to understand? How do people know what behavior is appropriate for a particular situation?

**Scripts** consist of prototypical sequences of sub-events, participants, and their roles.

The strong expectations provided by these scripts can help with the classification of entities, the assignment of entities into roles and relations.

Most critically scripts can be used to draw inferences that fill in things that have been left unsaid.



Script: RESTAURANT
Track: Coffee Shop
Props: Tables
Menu
F-Food
Check
Money

Roles: S-Customer
W-Waiter
C-Cook
M-Cashier
O-Owner

Entry conditions: S is hungry.
S has money.

Results: S has less money
O has more money
S is not hungry
S is pleased (optional)

Scene 1: Entering

S **PTRANS** S into restaurant
S **ATTEND** eyes **to** tables
S **MBUILD** where to sit
S **PTRANS** S **to** table
S **MOVE** S **to** sitting position

Scene 2: Ordering

(menu on table) S **PTRANS** menu **to** S

(W brings menu)
W **PTRANS** W **to** table
W **ATRANS** menu **to** S

(S asks for menu)
S **MTRANS** signal **to** W
W **PTRANS** W **to** table
S **MTRANS** 'need menu' **to** W
W **PTRANS** W **to** menu

S **MTRANS** food list **to CP**(S)
* S **MBUILD** choice of F
S **MTRANS** signal **to** W
W **PTRANS** W **to** table
S **MTRANS** 'I want F' **to** W

W **PTRANS** W **to** C
W **MTRANS** (**ATRANS** F) **to** C

C **MTRANS** 'no F' **to** W
W **PTRANS** W **to** S
W **MTRANS** 'no F' **to** S
(go back to *) or
(go to Scene 4 at no pay path)

C **DO** (prepare F script)
to Scene 3

# Unsupervised Learning of Narrative Event Chains

**Nathanael Chambers** and **Dan Jurafsky**
Department of Computer Science
Stanford University
Stanford, CA 94305
{natec,jurafsky}@stanford.edu

## Abstract

Hand-coded *scripts* were used in the 1970-80s as knowledge backbones that enabled inference and other NLP tasks requiring deep semantic knowledge. We propose unsupervised induction of similar schemata called *narrative event chains* from raw newswire text.

A narrative event chain is a partially ordered set of events related by a common protagonist. We describe a three step process to learning narrative event chains. The first uses unsupervised distributional methods to learn narrative relations between events sharing coreferring arguments. The second applies a temporal classifier to partially order the connected events. Finally, the third prunes and clusters self-contained chains from the space of events. We introduce two evaluations: the *narrative cloze* to evaluate event relatedness, and an *order coherence* task to evaluate narrative order. We show a 36% improvement over baseline for narrative prediction and 25% for temporal coherence.

## 1 Introduction

This paper induces a new representation of structured knowledge called **narrative event chains** (or narrative chains). Narrative chains are partially ordered sets of events centered around a common **protagonist**. They are related to structured sequences of participants and events that have been called **scripts** (Schank and Abelson, 1977) or *Fillmorean frames*. These participants and events can be filled in and instantiated in a particular text situation to draw inferences. Chains focus on a single actor to facilitate learning, and thus this paper addresses the three tasks of chain induction: *narrative event induction*, *temporal ordering of events* and *structured selection* (pruning the event space into discrete sets).

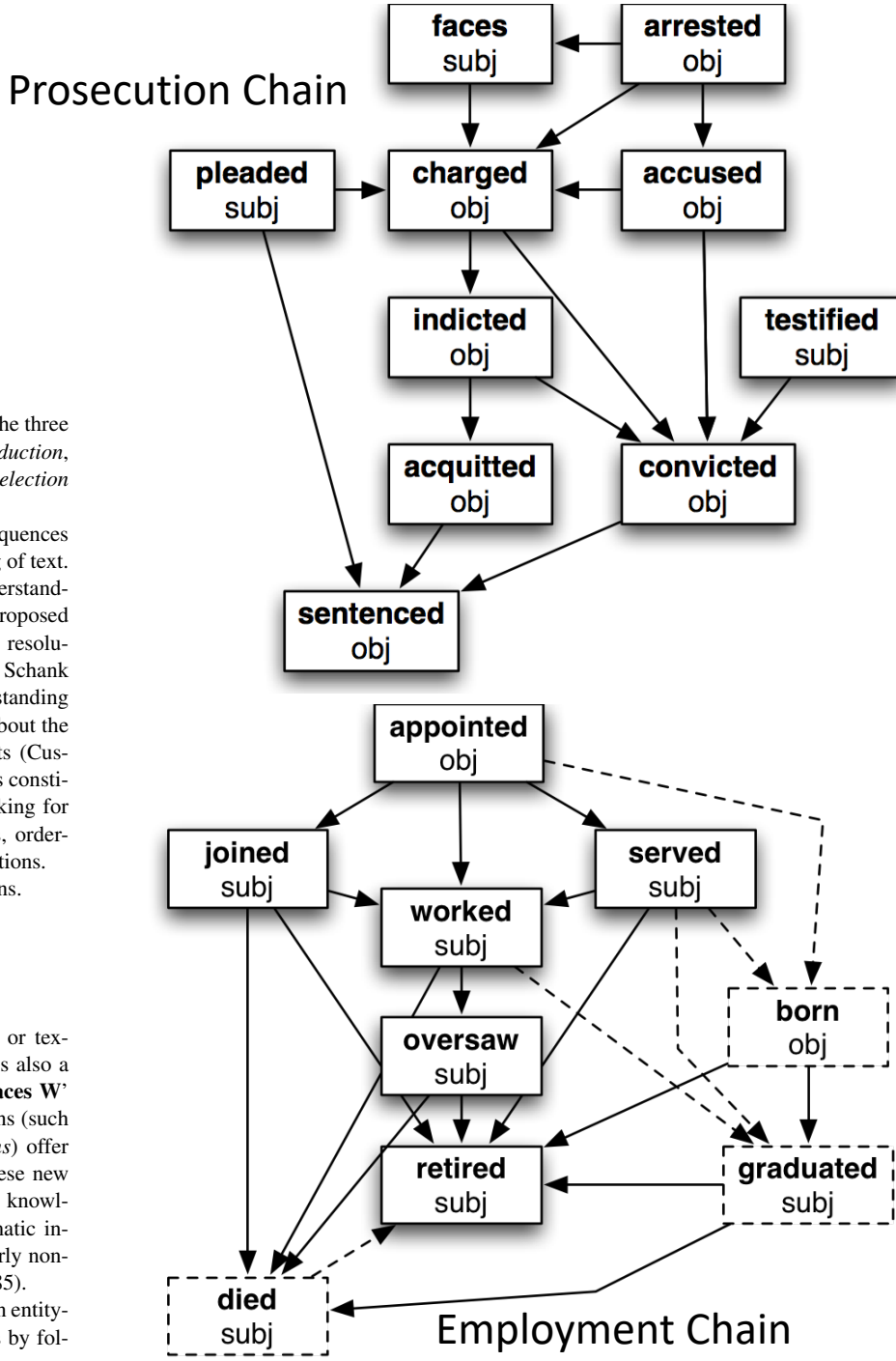
Learning these prototypical schematic sequences of events is important for rich understanding of text. Scripts were central to natural language understanding research in the 1970s and 1980s for proposed tasks such as summarization, coreference resolution and question answering. For example, Schank and Abelson (1977) proposed that understanding text about restaurants required knowledge about the Restaurant Script, including the participants (Customer, Waiter, Cook, Tables, etc.), the events constituting the script (entering, sitting down, asking for menus, etc.), and the various preconditions, ordering, and results of each of the constituent actions.

Consider these two distinct narrative chains.

| | |
|---|---|
| **_ accused X** | **W joined _** |
| **X claimed _** | **W served _** |
| **X argued** | **W oversaw _** |
| **_ dismissed X** | **W resigned** |

It would be useful for question answering or textual entailment to know that '**X denied _**' is also a likely event in the left chain, while '**_ replaces W**' temporally follows the right. Narrative chains (such as *Firing of Employee* or *Executive Resigns*) offer the structure and power to directly infer these new subevents by providing critical background knowledge. In part due to its complexity, automatic induction has not been addressed since the early non-statistical work of Mooney and DeJong (1985).

The first step to narrative induction uses an entity-based model for learning narrative relations by fol-



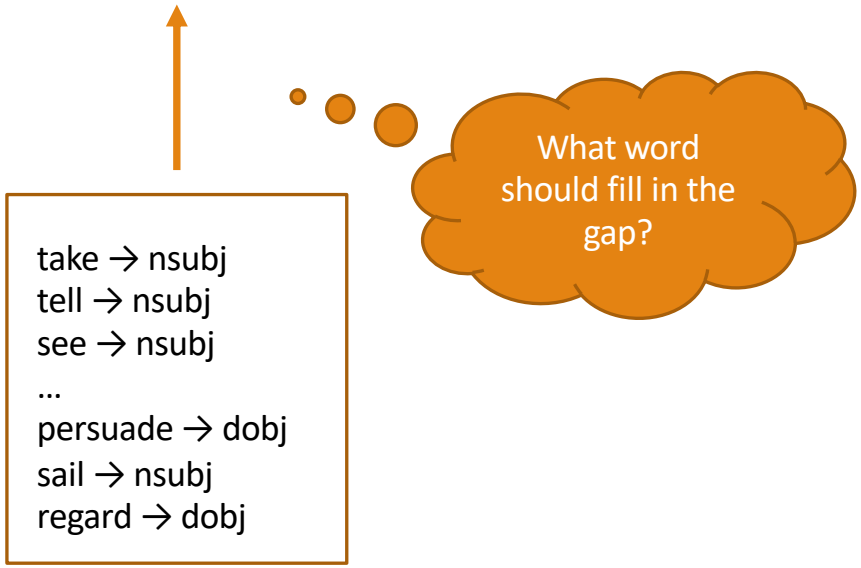
# Narrative Cloze Test

A **cloze test** removes words from a text and asks the participant to fill in the missing language item. Cloze tests require the ability to understand **context** and **vocabulary** in order to identify the correct language or part of speech that belongs in the deleted passages. This exercise is commonly administered for the assessment of native and second language learning and instruction.

Today, I went to the _____ and bought some milk and eggs. I knew it was going to rain, but I forgot to take my _____, and ended up getting wet on the way.

# Narrative Cloze Task

$event_1$ $event_2$ _____ $event_4$ … $event_n$



# Template Filling

Templates represent scripts with a fixed set of slots that take slot-filler values

Train two separate supervised systems

1. Template recognition
2. Role-filler extraction

Most earlier systems were based on handwritten regular expressions and grammar rules.

| | | |
|---|---|---|
| FARE-RAISE ATTEMPT: | LEAD AIRLINE: | UNITED AIRLINES |
| | AMOUNT: | $6 |
| | EFFECTIVE DATE: | 2006-10-26 |
| | FOLLOWER: | AMERICAN AIRLINES |

# The Gun Violence Database


Ellie Pavlick and Chris Callison-Burch, University of Pennsylvania

# Goals of the GVDB

Collect data about gun violence in the US to facilitate **public health research**.

Draw sample from local newspapers and television stations that publish online.

Use machine learning and crowdsourcing to **extract structured data from text**.

# GVDB Template

**Chicago Police release Laquan McDonald shooting video | National News**

Three seconds. On a dashcam video clock, that's the amount of time between the moment when two officers have their guns drawn and the point when Laquan McDonald falls to the ground. The video, released to the public for the first time late Tuesday, is a key piece of evidence in a case that's sparked protests in Chicago and has landed an officer behind bars. The 17-year-old McDonald was shot 16 times on that day the video shows in October 2014. Chicago police Officer Jason Van Dyke was charged Tuesday with first-degree murder….

Incident #1053

| | |
|---|---|
| City | |
| Date | |
| Shooter | |
| Victim | |
| Victim Killed | |

Person #1014

| | |
|---|---|
| Name | |
| Gender | |
| Age | |
| Race | |