Semantic Technologies

Exploiting deduction and abduction services for information retrieval

Ralf Moeller
Hamburg University of Technology

Semantic Technologies: Data Descriptions

- Ontologies (specified in, e.g., OWL, UML)
  (aka knowledge bases, represented in DLs plus safe rules)
- Deductive databases (e.g., Datalog),
- Logic programming (e.g., Answer-Set-Programming)
- Constraint-Solving Approaches (e.g., constraint DBs)

- Combinations with
  - Probability theory
  - Fuzzy set theory / Rough set theory

What can semantic technologies be used for today?

- Combinatorial problem solving (e.g., Sudoku)
- Systematic construction of domain models
  - Consistency checks, subsumption checks
  - Explanation, modularization, concept suggestions
- Instance retrieval / query answering
  - Almost main-memory-scalable
    for expressive ontology languages
  - Secondary-memory-scalable (almost)
    for data-oriented ontology languages
- Data integration

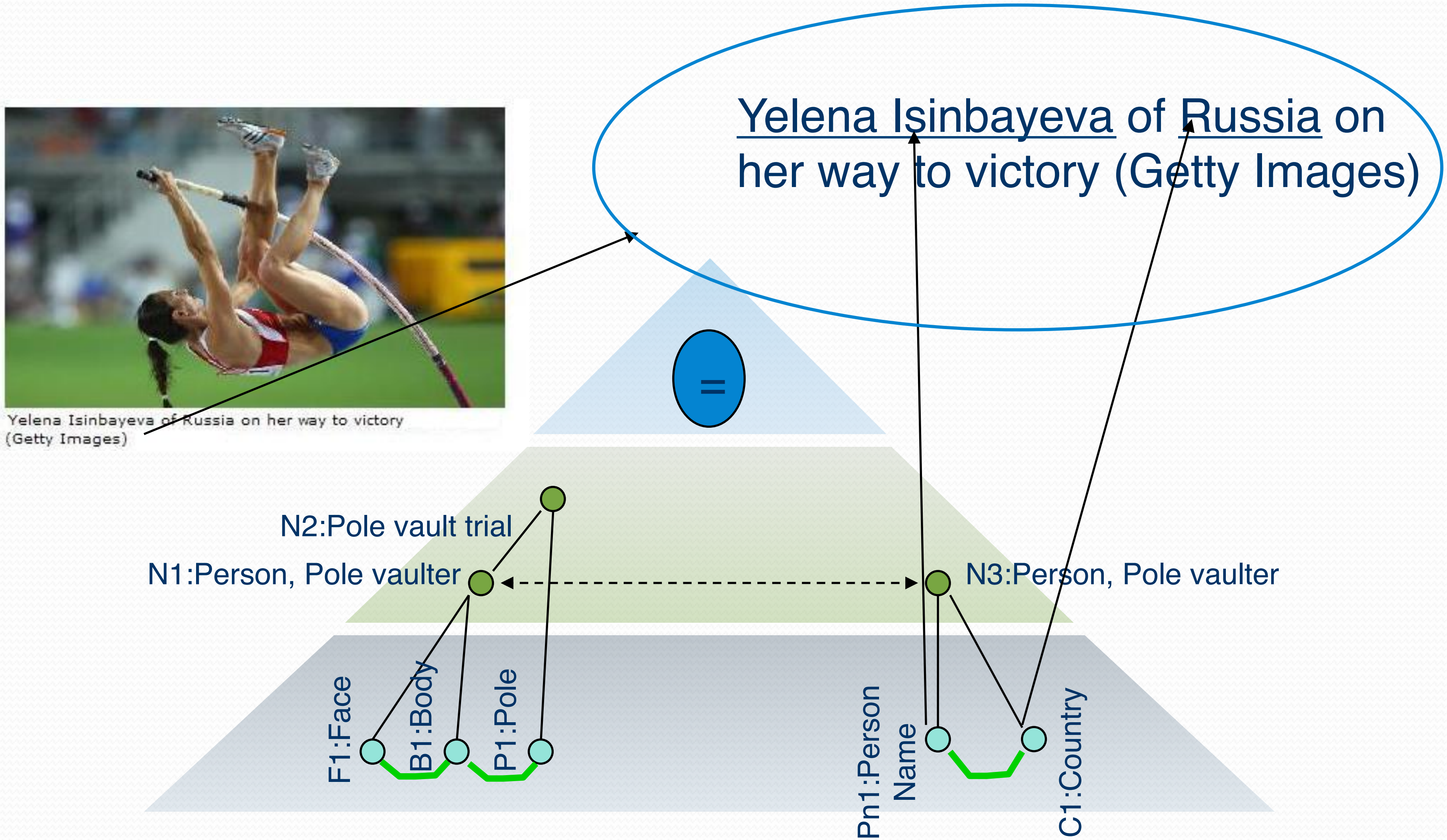
This list is probably incomplete

There's More to Semantic Technologies in the Future

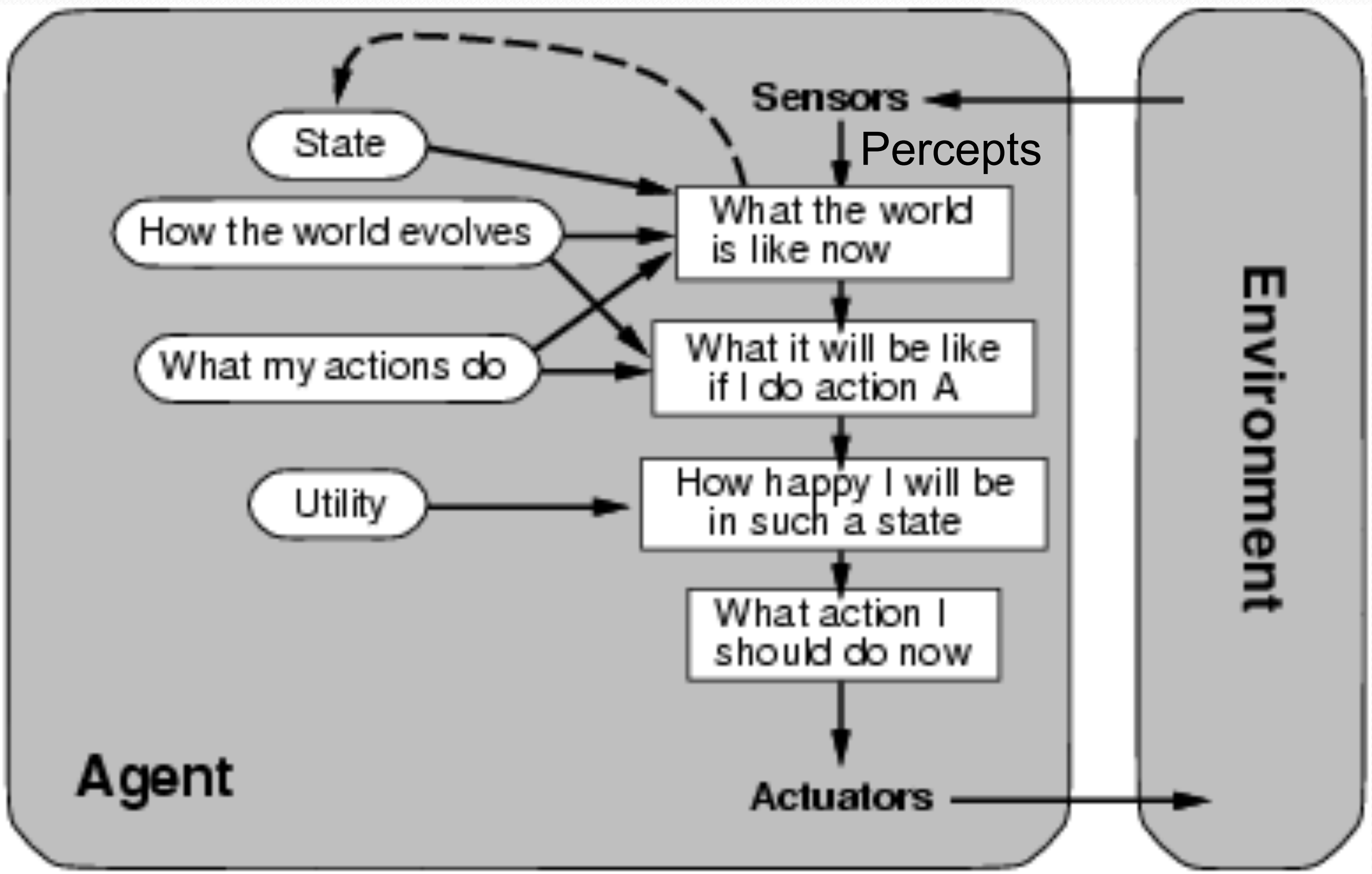
- Semantic Technologies now:
  - Data generation / query augmentation
- In some cases, only content-oriented retrieval is possible (e.g., for text, image, video, or audio data)
- Symbolic content descriptions are not directly available but have to be derived/provided (manual **annotation** / automatic **interpretation**)

Interpreting Non-Symbolic Data

Example 1: Web Pages

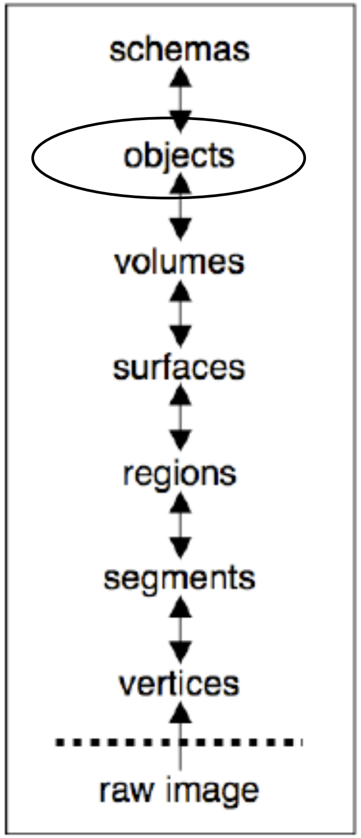


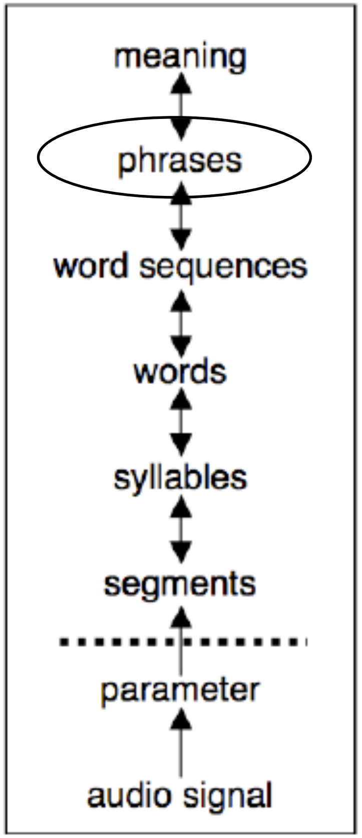
General View: Agents
How to Interpret Percepts?



Taken from: AIMA, Russell & Norvig, 1995

Derivation of Percepts





Principles of Interpretation

- **Explanation** of percepts/observations
  - Compute possible explanations in a defined space
    - Space defined using logical formulas
    - Number of constants not limited
    - Interpretation = Explanation = Abduction
      - Traditional approach: … [Hobbs et al. 93], [Shanahan 05]
- **Control** (ranking)
  - Accept (additional) explanations only if the probability that observations are true (given the additional explanations) is significantly increased.
- **Focus** of attention (forgetting)

Percepts and Interpretation as an Abox in RDF or OWL = Metadata

$$
\begin{array}{rcl}
mailman_1 & : & Mailman \\
bicycle_1 & : & Bicycle \\
mail\_deliv_1 & : & MailDelivery \\
(mail\_deliv_1, mailman_1) & : & hasPart \\
(mail\_deliv_1, bicycle_1) & : & hasPart \\
(mail\_deliv_1, url\_1) & : & hasURL \\
(mailman_1, url\_2) & : & hasURL \\
(bicycle_1, url\_3) & : & hasURL \\
(url\_1) & : & = \texttt{"http://www.img.de/image-1.jpg"} \\
(url\_2) & : & = \texttt{"http://www.img.de/image-1.jpg\#(200,400)/(300/500)"} \\
(url\_3) & : & = \texttt{"http://www.img.de/image-1.jpg\#(100,400)/(150/500)"} \\
garbageman_1 & : & Garbageman \\
garbageman_2 & : & Garbageman \\
garbagetruck_1 & : & Garbage\_Truck \\
garbage\_coll_1 & : & Garbage\_Collection \\
(garbage\_coll_1, garbageman_1) & : & hasPart \\
(garbage\_coll_1, garbageman_2) & : & hasPart \\
(garbage\_coll_1, garbagetruck_1) & : & hasPart \\
(garbage\_coll_1, url\_4) & : & hasURL \\
 & \ldots &
\end{array}
$$

Assume That We Have the Metadata…

- Then, IR can be realized as QA

$$ImageQuery_1 := \{(X, Y) \mid \quad MailDelivery(X), Bicycle(Y), hasPart(X, Y)\}$$

$$(mail\_deliv_1, bicycle_1)$$

$$URLQuery_1 := \{(X, value(X)) \mid \quad hasURL(mail\_deliv_1, X)\}$$

$$(url_1, \texttt{"http://www.img.de/image-1.jpg"})$$

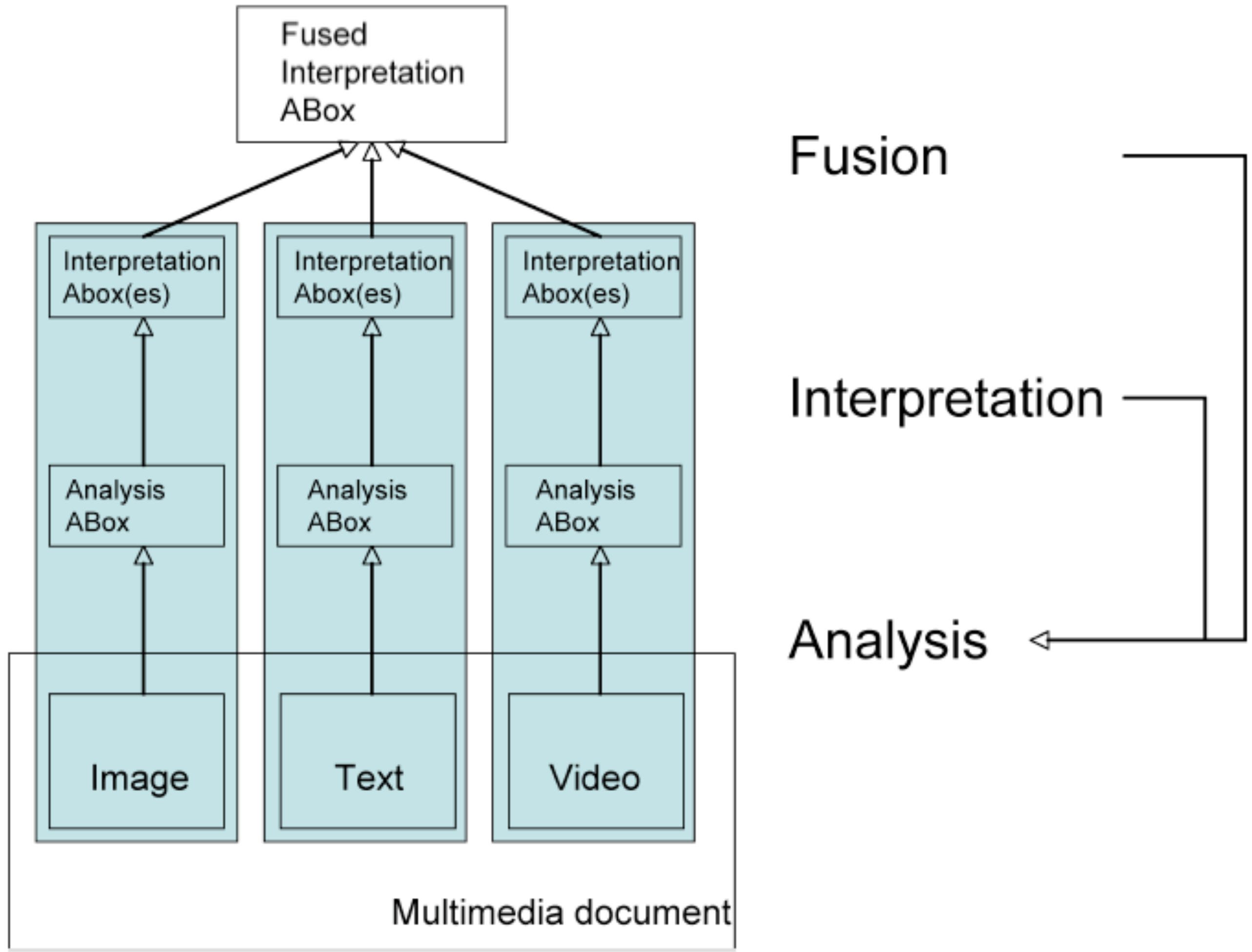
… QA w.r.t. a Tbox, of course

$$Mailman \sqsubseteq Postal\_Employee$$

$$Mailman \equiv Postman$$

Need not necessarily be the same Tbox as has been used for creating the metadata

How to Derive Metadata?



Formal Approach

- Logic-based
- Abduction (Hobbs 1993, Shanahan 2005)

$$\Sigma \cup \Delta \models \Gamma$$

1st Modality: Still Images

Analysis Abox Γ

$$\begin{array}{rcl} pole_1 & : & Pole \\ jumper_1 & : & Human \\ bar_1 & : & Horizontal\_Bar \\ (bar_1, jumper_1) & : & near \\ (jumper_1, pole_1) & : & touches \end{array}$$

Tbox Σ

(in-knowledge-base athletics)

(define-primitive-role **has-profession**)
(define-primitive-role **has-part**)
(define-primitive-role **has-component**)

(define-primitive-concept **phase**)
(define-primitive-concept **image**)
(define-primitive-concept **human**)
(define-primitive-concept **sports-equipment**)
(define-primitive-concept **body**)
(define-primitive-concept **face**)

(disjoint phase image human sports-equipment body face)

Tbox Σ (cntnd.)

(define-primitive-concept **athlete** human)
(define-primitive-concept **jumper** athlete)
(define-primitive-concept **pad** sports-equipment)
(define-primitive-concept **pole** sports-equipment)
(define-primitive-concept **javelin** sports-equipment)
(define-primitive-concept **horizontal-bar** sports-equipment)

Tbox Σ (cntnd.)

```
(define-primitive-concept jumping-event
  (and event (some has-part jumper)
  (at-most 1 has-part jumper)))

(define-primitive-concept pole-vault
  (and jumping-event
        (some has-part pole)
        (some has-part horizontal-bar)
        (some has-part foam-mat)))

(define-primitive-concept high-jump
  (and jumping-event
      (some has-part horizontal-bar)
      (some has-part foam-mat)))

(define-primitive-concept pv-in-start-phase phase)
(define-primitive-concept pv-in-end-start-phase phase)
(define-primitive-concept hj-in-jump-phase phase)
```

Role of the Tbox

- Introduces the signature of the domain
  - Sets up the domain vocabulary (e.g., Athletics)
  - Sets up the vocabulary to specify the organization of metadata (MCO)
- Imposes restrictions (define the models of the KB)
  - such that additional data is implicitly added (deduction) to interpretations,
  - and (potentially abducible) explanations causing inconsistencies are discarded

Need a little more...

- Abduction rules

$$
\begin{aligned}
touches(Y,Z) &\leftarrow Pole\_Vault(X), \\
&\quad PV\_InStartPhase(X), \\
&\quad hasPart(X,Y), Jumper(Y), \\
&\quad hasPart(X,Z), Pole(Z). \\
near(Y,Z) &\leftarrow Pole\_Vault(X), \\
&\quad PV\_InEndStartPhase(X), \\
&\quad hasPart(X,Y), Horizontal\_Bar(Y), \\
&\quad hasPart(X,Z), Jumper(Z). \\
near(Y,Z) &\leftarrow High\_Jump(X), \\
&\quad HJ\_InJumpPhase(X), \\
&\quad hasPart(X,Y), Horizontal\_Bar(Y), \\
&\quad hasPart(X,Z), Jumper(Z). \\
&\ldots
\end{aligned}
$$

… and maybe a few other things

- "Forward" rules (which extend an Abox)

$$touches(X, Z) :- hasPart(X, Y), touches(Y, Z).$$
$$near(X, Z) :- hasPart(X, Y), near(Y, Z).$$

Abduction: Refined Approach

- Refined version:

$$\Sigma \cup \Delta \cup \Gamma_1 \models \Gamma_2.$$

Bona fide assertions

Assertions requiring fiat
("fiat assertions")

$\mathbf{\Gamma}$

$$\begin{array}{rcl} pole_1 & : & Pole \\ jumper_1 & : & Human \\ bar_1 & : & Horizontal\_Bar \\ (bar_1, jumper_1) & : & near \\ (jumper_1, pole_1) & : & touches \end{array}$$

Abduction: Explanation Step

- Refined version:

$$\Sigma \cup \Delta \cup \Gamma_1 \models \Gamma_2$$

Bona fide assertions

Assertions requiring fiat ("fiat assertions")

$$\Gamma_1 = \Gamma \setminus \Gamma_2$$

Γ

$$\begin{array}{rcl} pole_1 & : & Pole \\ jumper_1 & : & Human \\ bar_1 & : & Horizontal\_Bar \end{array}$$

Γ_2

$$\begin{array}{rcl} \boxed{(bar_1, jumper_1) : near} \\ (jumper_1, pole_1) & : & touches \end{array}$$

Multiple Interpretations?

Ukraine's Andrey Sokolovskiy clears 2.38m in Rome (Getty Images)

Analysis Abox

$$
\begin{aligned}
bar_2 &: Horizontal\_Bar \\
jumper_2 &: Human \\
(bar_2, jumper_2) &: near
\end{aligned}
$$

Multiple Explanations

$$
\begin{array}{rcl}
jumper_2 & : & Human \\
bar_2 & : & Horizontal\_Bar \\
(bar_2, jumper_2) & : & near \\
hj_2 & : & High\_jump \\
hj_2 & : & HJ\_InJumpPhase \\
(hj_2, jumper_2) & : & hasPart \\
(hj_2, bar_2) & : & hasPart
\end{array}
$$

$$
\begin{array}{rcl}
jumper_2 & : & Human \\
bar_2 & : & Horizontal\_Bar \\
(bar_2, jumper_2) & : & near \\
pv_2 & : & Pole\_Vault \\
pv_2 & : & PV\_InEndStartPhase \\
(pv_2, jumper_2) & : & hasPart \\
(pv_2, bar_2) & : & hasPart
\end{array}
$$

Another Example: Ranking

High Jump?

$$\begin{aligned} bar_3 &: Horizontal\_Bar \\ jumper_3 &: Jumper \\ pole_3 &: Pole \\ (bar_3, jumper_3) &: near \end{aligned}$$

Also should be able to explain the pole!

2^{nd} Modality: Text

A new world record in this year's event was missed.
The remaining famous athlete touched
the crossbar and failed 2.40m.

$$\begin{array}{rcl} object_1 & : & Event \\ object_2 & : & Horizontal\_Bar \\ object_3 & : & Athlete \\ object_3 & : & Famous \\ (object_1, object_2) & : & precedes \\ (object_1, object_3) & : & precedes \end{array}$$

Text Interpretation Knowledge: $hasPart \sqsubseteq precedes$

Shallow Text Interpretation

Abduction query:

$$Q_2 := \{() \mid precedes(object_1, object_2), precedes(object_1, object_3)\}$$

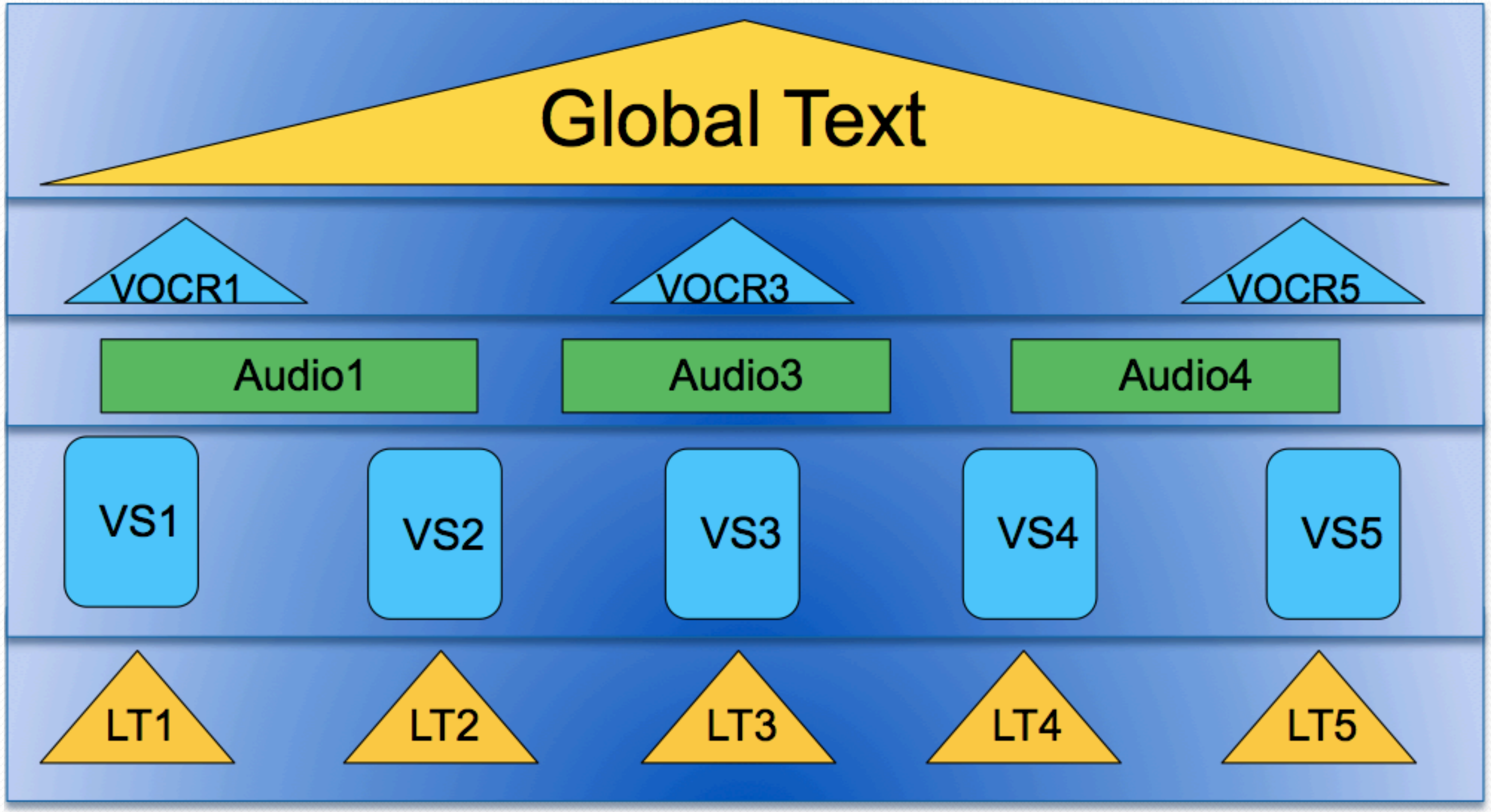
Interpretation:

$$\begin{array}{rcl} object_1 & : & High\_Jump \\ (object_1, object_2) & : & hasPart \\ (object_1, object_3) & : & hasPart \end{array}$$

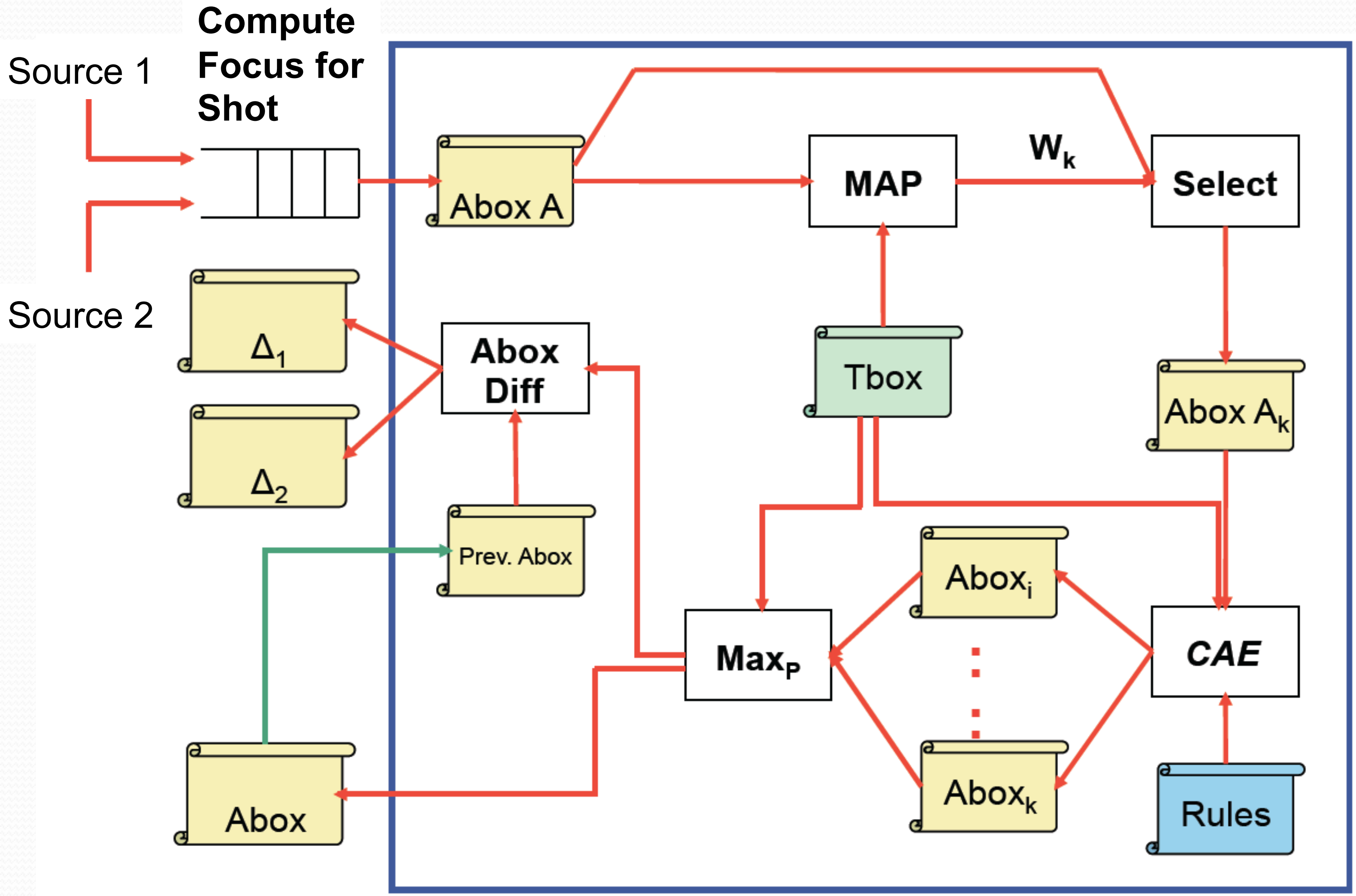
3rd Modality Scene Understanding

- Prerequisite: Video (automatically) decomposed into small shots (video + audio data)
- Data-driven (low-level) results are computed
  - Object recognition/tracking (incl. object relations)
  - Speech recognition
  - Video OCR
- Shot-relevant Abox assertions are collected
  - Lower-level analysis results
  - Interpretation results from the interpretation Abox of previous shot (focus of attention)
- Derivation of shot-specific interpretations

Video Metadata



Interpretation Architecture



Controlling Interpretation

- Based on Markov logic networks [Domingos et al.]
- Knowledge base (syntax, signature, and semantics)

$$MLN = (\mathcal{F}_{MLN}, \mathcal{W}_{MLN})$$

$$(\vec{X}) =$$

$$rand\_vars((\mathcal{F}, \mathcal{W})) := \{P(\underline{C}) \mid P(\underline{C}) \text{ is mentioned in some grounded formula } F \in \mathcal{F}\}$$

$$\mathbf{P}_{MLN}(\vec{X}) = (P(\vec{X} = \vec{x}_1), \ldots, P(\vec{X} = \vec{x}_n))$$

$$P(\vec{X} = \vec{x}) = \frac{1}{Z} exp(\sum_{i=1}^{|\mathcal{F}_{MLN}|} w_i n_i(\vec{x}))$$

$$Z = \sum_{\vec{x} \in \vec{X}} \exp(\sum_{i=1}^{|\mathcal{F}_{MLN}|} w_i n_i(\vec{x}))$$

Another part of the KB

- Weighted rules

$$5\,\forall z\, CarEntry(z) \wedge hasObject(z,x) \wedge hasEffect(z,y) \rightarrow Car(x) \wedge DoorSlam(y) \wedge causes(x,y)$$

$$5\,\forall z\, EnvConference(z) \wedge hasSubEvent(z,x) \wedge hasLocation(z.y) \rightarrow CarEntry(x) \wedge Building(y) \wedge OccursAt(x,y)$$

MLN Query Answering

- **Probability query**:
  - Syntax: $P_{MLN}(x_1 \wedge \ldots \wedge x_m \mid \vec{e})$
  - Semantics:

$$P_{rand\_vars(MLN)}(x_1 \wedge \ldots \wedge x_m \mid \vec{e}) \; w.r.t. \; \mathbf{P}_{MLN}(rand\_vars(MLN))$$

- **Most-probable world query** (Maximum A-Posterior, MAP)

$$MAP_{MLN}(\vec{e}) := \vec{e} \cup argmax_{\vec{x}} \frac{1}{Z_e} \exp\left(\sum_i w_i n_i\left(\vec{x}, \vec{e}\right)\right)$$

which can be slightly optimized s.th.

$$MAP_{MLN}(\vec{e}) := \vec{e} \cup argmax_{\vec{x}} \sum_i w_i n_i\left(\vec{x}, \vec{e}\right)$$

Concept-based Abduction Engine: Basic Idea

1. Given a set of observations Γ , try to explain each assertion (Explanation Step as discussed before)
2. Each explanation introduces new assertions
3. Apply rules to each explanation (possibly gives us new assertions as well)
4. Continue with step 1. unless none of the explanations derived in this round cause the probability that the initial observations are true to increase "substantially"
5. Return the explanations (to be ranked afterwards)

Example

Tbox:

$$CarEntry \sqsubseteq \neg DoorSlam$$

...

Abduction rules:

$$Causes(x,y) \leftarrow CarEntry(z), HasObject(z,x), HasEffect(z,y), Car(x), DoorSlam(y)$$

$$OccursAt(x,y) \leftarrow EnvConference(z), HasSubEvent(z,x), HasLocation(z,y), CarEntry(x), Building(y)$$

Forward rules:

$$\forall x \quad CarEntry(x) \rightarrow \exists y \quad Building(y), OccursAt(x,y)$$

Weighted rules:

$$5\, \forall z\, CarEntry(z) \wedge hasObject(z,x) \wedge hasEffect(z,y) \rightarrow Car(x) \wedge DoorSlam(y) \wedge causes(x,y)$$

$$5\, \forall z\, EnvConference(z) \wedge hasSubEvent(z,x) \wedge hasLocation(z.y) \rightarrow CarEntry(x) \wedge Building(y) \wedge OccursAt(x,y)$$

Formulas are extremely simplified to make them fit on a slide.

Example (cont.)

Γ

$$\begin{array}{ll} 1.3 & Car(C_1) \\ 1.2 & DoorSlam(DS_1) \\ -0.3 & EngineSound(ES_1) \\ & Causes(C_1, DS_1). \end{array}$$

Combination of audio and video for this focus

MAP

Ground atoms	W
$Car(C_1)$	1
$DoorSlam(DS_1)$	1
$EngineSound(ES_1)$	0
$Causes(C_1, DS_1)$	1

Select

Γ'

$$\begin{array}{ll} 1.3 & Car(C_1) \\ 1.2 & DoorSlam(DS_1) \\ & Causes(C_1, DS_1). \end{array}$$

Example (cont.)

Γ'

1.3	$Car(C_1)$
1.2	$DoorSlam(DS_1)$
	$Causes(C_1, DS_1).$

$$\Delta_1 = \{CarEntry(Ind_{42}), HasObject(Ind_{42}, C_1), HasEffect(Ind_{42}, DS_1)\}$$

Abox A_1

1.3	$Car(C_1)$
1.2	$DoorSlam(DS_1)$
	$Causes(C_1, DS_1).$
	$CarEntry(Ind_{42}).$
	$HasObject(Ind_{42}, C_1).$
	$HasEffect(Ind_{42}, DS_1).$

$$P(Car(C_1) \wedge DoorSlam(DS_1) \wedge Causes(C_1, DS_1) \mid \Delta_1) = 0.840$$

Example (cont.)

$$\forall x \quad CarEntry(x) \rightarrow \exists y \quad Building(y), OccursAt(x,y)$$

$$\Delta_f = \{Building(Ind_{43}), OccursAt(Ind_{42}, Ind_{43})\}$$

1.3	$Car(C_1)$
1.2	$DoorSlam(DS_1)$
	$Causes(C_1, DS_1).$
	$CarEntry(Ind_{42}).$
	$HasObject(Ind_{42}, C_1).$
	$HasEffect(Ind_{42}, DS_1).$
	$Building(Ind_{43}).$
	$OccursAt(Ind_{42}, Ind_{43}).$

$$\Delta_2 = \{EnvConference(Ind_{44}), hasSubEvent(Ind_{44}, Ind_{42}), hasLocation(Ind_{44}, Ind_{43}), \ldots\}$$

Example (cont.)

Abox $\mathcal{A}_2$

1.3	$Car(C_1)$
1.2	$DoorSlam(DS_1)$
	$Causes(C_1, DS_1).$
	$CarEntry(Ind_{42}).$
	$HasObject(Ind_{42}, C_1).$
	$HasEffect(Ind_{42}, DS_1).$
	$Building(Ind_{43}).$
	$OccursAt(Ind_{42}, Ind_{43}).$
	$EnvConference(Ind_{44}).$
	$HasSubEvent(Ind_{44}, Ind_{42}).$
	$HasLocation(Ind_{44}, Ind_{43}).$
	…

Δ_1 ($CarEntry$, $HasObject$, $HasEffect$)

Δ_f ($Building$, $OccursAt$)

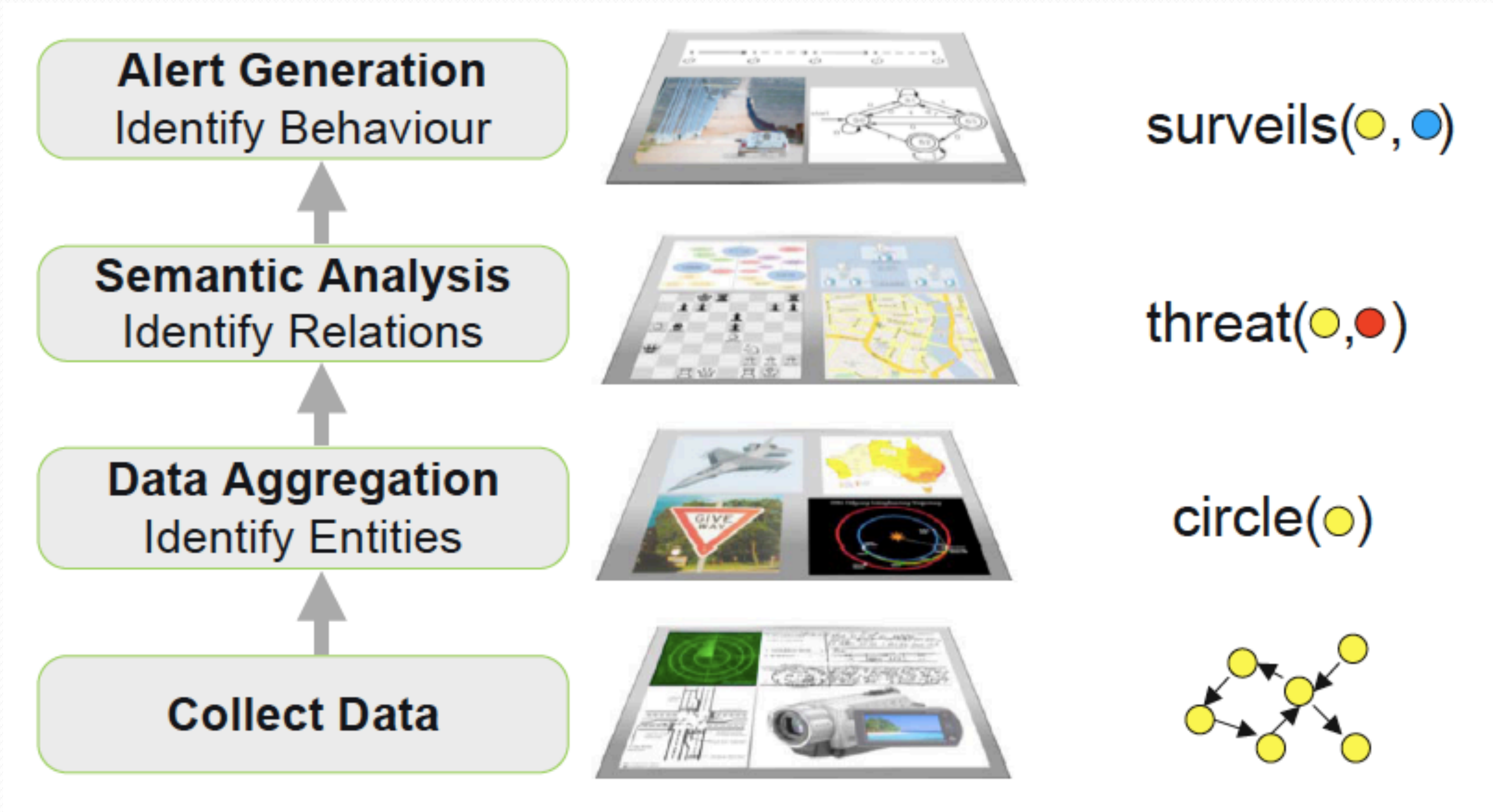
Δ_2 ($EnvConference$, $HasSubEvent$, $HasLocation$)

$$P(Car(C_1) \wedge DoorSlam(DS_1) \wedge Causes(C_1, DS_1) \mid \Delta_1 \cup \Delta_f \cup \Delta_2) = 0.819$$

The termination condition is fulfilled.

→ **Abox A_1 is considered as the output Abox.**

Situation Awareness



Taken from: A Novel Architecture for Situation Awareness Systems, Baader et al. 09

Thank you

- This work has been supported by
  - the European Commission (Projects CASAM and BOEMIE),
  - and the Deutsche Forschungsgemeinschaft (DFG) (Project PRESINT)
- Thanks also to all members of my group at TUHH,
  - in particular Oliver Gries, Anahita Nafissi, Maurice Rosenfeld, Kamil Sokolski, Michael Wessel

Rules with Time Variables

$$
\begin{aligned}
High\_Jump\_Event_{[T_1,T_2]}(X,Y) \quad :- \quad & accelerate\_horizontally_{[T_1,T_3]}(Y), \\
& vertical\_upward\_movement_{[T_3,T_4]}(Y), \\
& turn_{[T_4,T_5]}(Y), \\
& vertical\_downward\_movement_{[T_5,T_2]}(Y). \\
& Jumper(Y), \\
& High\_Jump(X), \\
& hasPart(X,Y),
\end{aligned}
$$

Temporal Propositions

- High-level interpretation results:

$$\begin{aligned} accelerate\_horizontally_{[219,224]} & \quad (moving\_object_1) \\ vertical\_upward\_movement_{[224,226]} & \quad (moving\_object_1) \\ turn_{[226,228]} & \quad (moving\_object_1) \\ vertical\_downward\_movement_{[228,230]} & \quad (moving\_object_1) \end{aligned}$$

$$\begin{aligned} moving\_object_1 &: \quad Jumper \\ event_1 &: \quad High\_Jump \\ (event_1, moving\_object_1) &: \quad hasPart \end{aligned}$$

- Query: $\{(X)_{[T_1,T_2]} \mid High\_Jump\_Event_{[T_1,T_2]}(X,Y)\}$
- Result: $\{(((X, event_1)), (T_1, (219, 223)), (T_2, (229, 230)))\}$

CAE: Conceptual Abduction Engine

Function CAE(Ω, Ξ, Σ, $\mathcal{R}$, S, $\mathfrak{A}$):
Input: a strategy function Ω, a termination function Ξ, a background KB Σ
rules $\mathcal{R}$, a scoring function S, and an agenda $\mathfrak{A}$
Output: a set of interpretation Aboxes $\mathfrak{I}'$
$\mathfrak{I}' := \{assign\_level(l, \mathfrak{A})\}$;
repeat
  $\mathfrak{I} := \mathfrak{I}'$;
  $(\mathcal{A}, \alpha) := \Omega(\mathfrak{I})$ // $\mathcal{A} \in \mathfrak{I}$, $\alpha \in \mathcal{A}$ s.th. $requires\_fiat(\alpha^l)$ holds;
  $l = l + 1$;
  $\mathfrak{I}' := (\mathfrak{A} \setminus \{\mathcal{A}\}) \cup assign\_level(l, explanation\_step(\Sigma, \mathcal{R}, S, \mathcal{A}, \alpha))$;
until $\Xi(\mathfrak{I})$ *or no* $\mathcal{A}$ *and* α *can be selected such that* $\mathfrak{I}' \neq \mathfrak{I}$;
return $\mathfrak{I}'$

Interpret

Function Interpret($\mathfrak{A}$, $CurrentI$, Γ, $\mathcal{T}$, $\mathcal{FR}$, $\mathcal{BR}$, $\mathcal{WR}$, ϵ)
Input: an agenda $\mathfrak{A}$, a current interpretation Abox $CurrentI$, an Abox of observations Γ, a Tbox $\mathcal{T}$, a set of forward chaining rules $\mathcal{FR}$, a set of backward chaining rules $\mathcal{BR}$, a set of weighted rules $\mathcal{WR}$, and the desired precision of the results ϵ
Output: an agenda $\mathfrak{A}'$, a new interpretation Abox $NewI$, and Abox differences for additions Δ_1 and omissions Δ_2

$i := 0$;
$p_0 := P(\Gamma, \Gamma, \mathcal{R}, \mathcal{WR}, \mathcal{T})$;
$\Xi := \lambda(\mathfrak{A}) \bullet \{i := i + 1; p_i := \max_{\mathcal{A} \in \mathfrak{A}} P(\Gamma, \mathcal{A} \cup \mathcal{A}_0, \mathcal{R}, \mathcal{WR}, \mathcal{T}); \mathbf{return} \mid p_i - p_{i-1} \mid < \frac{\epsilon}{i}\}$;
$\Sigma := (\mathcal{T}, \emptyset)$;
$\mathcal{R} := \mathcal{FR} \cup \mathcal{BR}$;
$S := \lambda((\mathcal{T}, \mathcal{A}_0)), \mathcal{R}, \mathcal{A}, \Delta) \bullet P(\Gamma, \mathcal{A} \cup \mathcal{A}_0 \cup \Delta, \mathcal{R}, \mathcal{WR}, \mathcal{T})$;
$\mathfrak{A}' := CAE(\Omega, \Xi, \Sigma, \mathcal{R}, S, \mathfrak{A})$;
$NewI = argmax_{\mathcal{A} \in \mathfrak{A}'}(P(\Gamma, \mathcal{A}, \mathcal{R}, \mathcal{WR}, \mathcal{T}))$;
$\Delta_1 = AboxDiff(NewI, CurrentI)$; `// additions`
$\Delta_2 = AboxDiff(CurrentI, NewI)$; `// omissions`
return $(\mathfrak{A}', NewI, \Delta_1, \Delta_2)$;

Media Interpretation Agent

Function MI_Agent(Q, $Partners$, Die, $(\mathcal{T}, \mathcal{A}_0), \mathcal{FR}, \mathcal{BR}, \mathcal{WR}, \epsilon$)
Input: a queue of percept results Q, a set of partners $Partners$, a function Die for the termination process, a background knowledge set $(\mathcal{T}, \mathcal{A}_0)$, a set of forward chaining rules $\mathcal{FR}$, a set of backward chaining rules $\mathcal{BR}$, a set of weighted rules $\mathcal{WR}$, and the desired precision of the results ϵ
Output: –

$CurrentI = \emptyset$;
$\mathfrak{A}'' = \{\emptyset\}$;
repeat
$\Gamma := extractObservations(Q)$;
$W := MAP(\Gamma, \mathcal{WR}, \mathcal{T})$;
$\Gamma' := Select(W, \Gamma)$;
$\mathfrak{A}' := filter(\lambda(\mathcal{A}) \bullet consistent_{\Sigma}(\mathcal{A}), map(\lambda(\mathcal{A}) \bullet \Gamma' \cup \mathcal{A} \cup \mathcal{A}_0 \cup forward\_chain(\Sigma, \mathcal{FR}, \Gamma' \cup \mathcal{A} \cup \mathcal{A}_0),$
$\{select(MAP(\Gamma' \cup \mathcal{A} \cup \mathcal{A}_0, \mathcal{WR}, \mathcal{T}), \Gamma' \cup \mathcal{A} \cup \mathcal{A}_0) \mid \mathcal{A} \in \mathfrak{A}''\}))$;
$(\mathfrak{A}'', NewI, \Delta_1, \Delta_2) := Interpret(\mathfrak{A}', CurrentI, \Gamma', \mathcal{T}, \mathcal{FR}, \mathcal{BR}, \mathcal{WR} \cup \Gamma, \epsilon)$;
$CurrentI := NewI$;
$Communicate(\Delta_1, \Delta_2, Partners)$;
$\mathfrak{A}'' := manage\_agenda(\mathfrak{A}'')$;
until $Die()$;