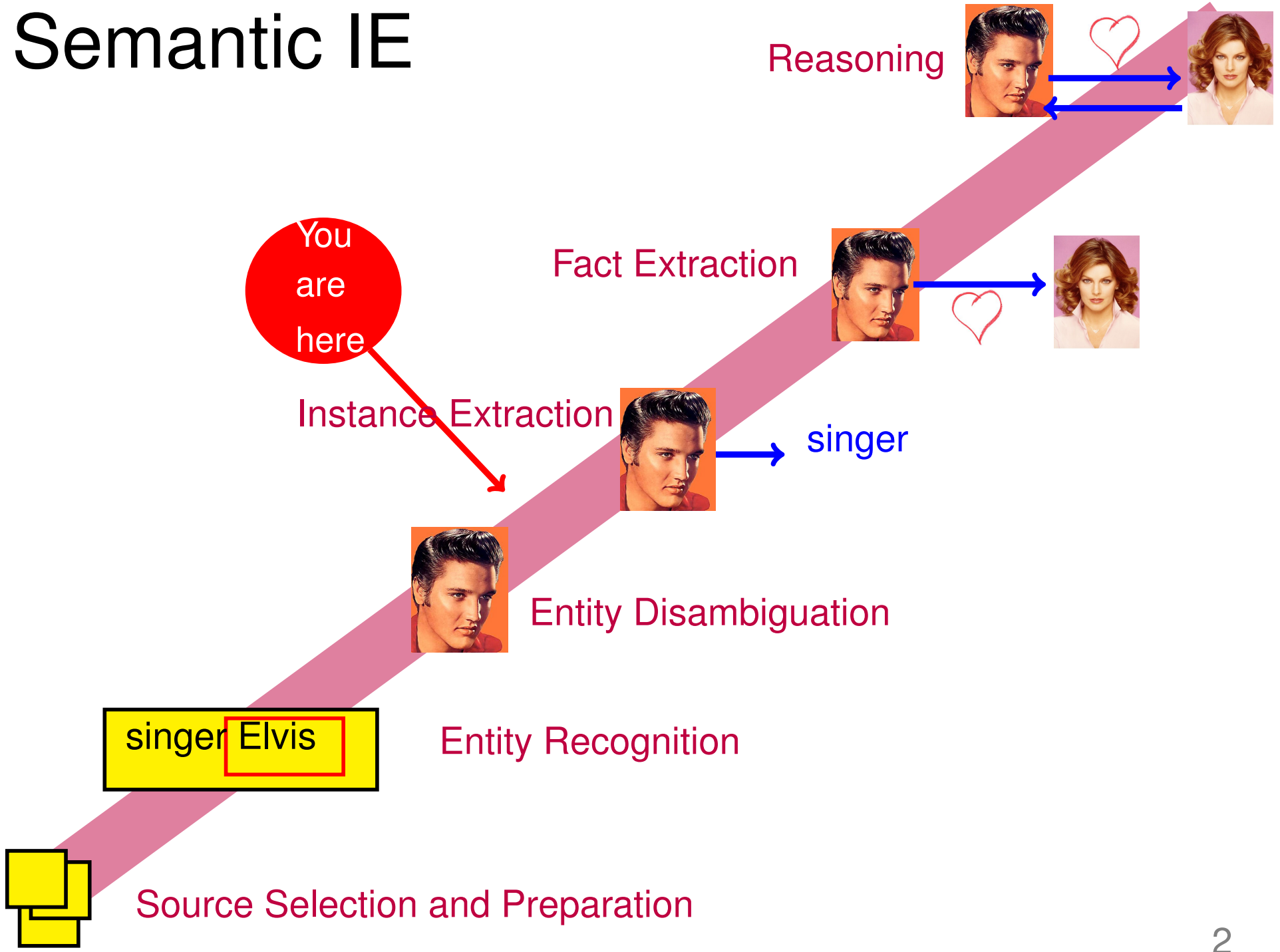
Instance Extraction

Fabian M. Suchanek

Semantic IE


Reasoning
Fact Extraction
You are here
Instance Extraction
singer
Entity Disambiguation
singer Elvis
Entity Recognition
Source Selection and Preparation

Def: Is-A

Is-A is the binary relation that holds between X and Y,
if X is an instance or a subclass of Y.

is-a(Lisa, girl)
is-a(girl, person)

In all of the following, we assume that the entity mentions in
our corpora have been disambiguated already (= are unambiguous).

Is-A Extraction

Is-A Extraction is the task of extracting Is-A facts from a corpus.

(Different from NEA, the class names are not given upfront.)

In the Simpson episode ”HOMR”, Doctor Monson discovers a crayon in Homer’s brain and removes it. His IQ goes up from 55 to 105, but he feels uncomfortable and wants it back. Moe, who is not only a bartender but also an unlicensed physician, puts the crayon back, returning Homer to the idiot.

Is-A Extraction

Is-A Extraction is the task of extracting Is-A facts from a corpus.

In the Simpson episode ”HOMR”, Doctor Monson discovers a crayon in Homer’s brain and removes it. His IQ goes up from 55 to 105, but he feels uncomfortable and wants it back. Moe, who is not only a bartender but also an unlicensed physician, puts the crayon back, returning Homer to the idiot.

HOMR	is-a	Simpson episode
Monson	is-a	Doctor
Homer	is-a	idiot
Moe	is-a	bartender
Moe	is-a	unlicensed physician

Def: Hearst Patterns

A Hearst pattern is a simple textual pattern that indicates an is-a fact.

”Y such as X”

An idiot such as Homer. → is-a(Homer, idiot)

Def: Hearst Patterns

A Hearst pattern is a simple textual pattern that indicates an is-a fact.

”Y such as X”

An idiot such as Homer. → is-a(Homer, idiot)

...many activists, such as Lisa...

is-a(Lisa, activist)

...some animals, such as dogs...

is-a(dog, animal)

...some scientists, such as computer scientists...

...some plants, such as nuclear power plants....

Def: Hearst Patterns

A Hearst pattern is a simple textual pattern that indicates an is-a fact.

”Y such as X”

An idiot such as Homer. → is-a(Homer, idiot)

...many activists, such as Lisa...

is-a(Lisa, activist)

...some animals, such as dogs...

is-a(dog, animal)

...some scientists, such as computer scientists...

is-a(computer, scientist) ?

...some plants, such as nuclear power plants....

is-a(nuc.Pow.Plants, plants) ?

Def: Hearst Patterns

A Hearst pattern is a simple textual pattern that indicates an is-a fact.

"Y such as X"

An idiot such as Homer. → is-a(Homer, idiot)

...many activists, such as Lisa...

is-a(Lisa, activist)

...some animals, such as dogs...

is-a(dog, animal)

...some scientists, such as computer scientists...

is-a(computer, scientist) ?

...some plants, such as nuclear power plants....

is-a(nuc.Pow.Plants, plants) ?

=> Hearst patterns have to be combined with NER and dis-ambiguation to yield entity facts.

Def: Classical Hearst Patterns

The classical Hearst Patterns are

Y such as X+

such Y as X+

X+ and other Y

Y including X+

Y, especially X+

...where X+ is a list of names of the form "X_1 ,...,X_{n-1} (and—or)? X_n".

(In the original paper, the X_i are noun phrases)

These imply is-a(X_i ,Y).

(assuming that the words are disambiguated)

Marti Hearst: Automatic Acquisition of Hyponyms

Task: Classical Hearst Patterns

Apply

1. Y such as X+
2. such Y as X+
3. X+ and other Y
4. Y including X+
5. Y, especially X+

I lived in such countries as Germany, France, and Bavaria.

He wrote about fictional entities such as Homer, Lisa, and Bielefeld.

I love people that are not genies, especially Homer.

Example: Hearst on the Web

"cities such as"

Web Images Maps Shopping More Search tools

About 79,800,000 results (0.19 seconds)

These 12 Hellholes Are Examples Of What The Rest Of America Wi...
theeconomiccollapseblog.com/.../these-12-hellholes-are-examples-of-wh...
Jul 15, 2012 – The reality is that most of the country has been experiencing a slow decline for a very long time and once thriving **cities such as** Gary, Indiana ...

City - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/City
Every city expansion would imply a new circle (canals together with town walls). In **cities such as** Amsterdam, Haarlem, and also Moscow, this pattern is still ...

try it out

finish>92

Def: Set Expansion

Set Expansion is the task of, given names of instances of a class ("seeds"), extracting more such instance names from a corpus.

cities: {Springfield, Seattle}

↓ Set Expansion

cities: {Springfield, Seattle, Washington, Chicago, ...}

Def: Recursive Pattern Application

Recursive Pattern Application is the following algorithm for set expansion:

0. Start with the seeds

cities: {Austin, Seattle}

Def: Recursive Pattern Application

Recursive Pattern Application is the following algorithm for set expansion:

0. Start with the seeds

cities: {Austin, Seattle}

1. Find the pattern “X, Y, and Z” in the corpus.

Seattle, Chicago, and Austin

Def: Recursive Pattern Application

Recursive Pattern Application is the following algorithm for set expansion:

0. Start with the seeds

cities: {Austin, Seattle}

1. Find the pattern "X, Y, and Z" in the corpus.

Seattle, Chicago, and Austin

2. If 2 variables match known instance names, add the match of the 3rd.

Def: Recursive Pattern Application

Recursive Pattern Application is the following algorithm for set expansion:

0. Start with the seeds

cities: {Austin, Seattle}

1. Find the pattern "X, Y, and Z" in the corpus.

Seattle, Chicago, and Austin

2. If 2 variables match known instance names, add the match of the 3rd.

cities: {Austin, Seattle, Chicago}

3. Go to 1

Task: Recursive Pattern Appl.

cities: {Springfield, Austin, Seattle}

... Austin, Seattle, and Houston...

Task: Recursive Pattern Appl.

cities: {Springfield, Austin, Seattle}

... Austin, Seattle, and Houston...

cities: {Springfield, Austin, Seattle, Houston}

Task: Recursive Pattern Appl.

cities: {Springfield, Austin, Seattle}

... Austin, Seattle, and Houston...

cities: {Springfield, Austin, Seattle, Houston}

... Houston, Chicago, and Springfield...

Task: Recursive Pattern Appl.

cities: {Springfield, Austin, Seattle}

... Austin, Seattle, and Houston...

cities: {Springfield, Austin, Seattle, Houston}

... Houston, Chicago, and Springfield...

cities: {Springfield, Austin, Seattle, Houston, Chicago}

Task: Recursive Pattern Appl.

cities: {Springfield, Austin, Seattle}

... Austin, Seattle, and Houston...

cities: {Springfield, Austin, Seattle, Houston}

... Houston, Chicago, and Springfield...

cities: {Springfield, Austin, Seattle, Houston, Chicago}

... Austin, Texas, and Seattle, Washington...

Task: Recursive Pattern Appl.

cities: {Springfield, Austin, Seattle}

... Austin, Seattle, and Houston...

cities: {Springfield, Austin, Seattle, Houston}

... Houston, Chicago, and Springfield...

cities: {Springfield, Austin, Seattle, Houston, Chicago}

... Austin, Texas, and Seattle, Washington...

Precision may suffer over time

Def: Semantic Drift

Semantic Drift is the problem in Set Expansion that names of instances of other classes get into the set.

cities: {Springfield, Austin, Seattle, Houston}

... Houston, Chicago, and Springfield...

cities: {Springfield, Austin, Seattle, Houston, Chicago}

... Austin, Texas, and Seattle, Washington...

cities: {Chicago, Seattle, ..., Texas}

Def: Table Set Expansion

Table Set Expansion is the following algorithm for set expansion:

0. Start with the seeds — countries: {Russia, China}

1. Find HTML tables where one column contains 2 known instance names

Largest Countries in the World

view as: list / slideshow / map

▲	Country	Total Area (sq km)
1.	Russia	17,098,242
2.	Canada	9,984,670
3.	United States	9,826,675
4.	China	9,596,961

2. Add all column entries to the set — countries: {Russia, China, Canada, United States}

3. Go to 1

Example: Table Set Expansion

countries: {Russia, China, Brazil}

Example: Table Set Expansion

countries: {Russia, China, Brazil}

Richest Countries in the World

view as: list / slideshow / map

▲	Country	GDP
1.	United States	$15,290,000,000,000
2.	China	$11,440,000,000,000
3.	India	$4,515,000,000,000
4.	Japan	$4,497,000,000,000
5.	Germany	$3,139,000,000,000
6.	Russia	$2,414,000,000,000

Example: Table Set Expansion

countries: {Russia, China, Brazil}

Richest Countries in the World

view as: list / slideshow / map

▲	Country	GDP
1.	United States	$15,290,000,000,000
2.	China	$11,440,000,000,000
3.	India	$4,515,000,000,000
4.	Japan	$4,497,000,000,000
5.	Germany	$3,139,000,000,000
6.	Russia	$2,414,000,000,000

countries: {Russia, China, Brazil, United States, Japan, India, Germ.}

Example: Table Set Expansion

countries: {Russia, China, Brazil, United States, Japan, India, Germ.}

Countries with the Largest Armed Forces in the World

view as: list / slideshow / map

▲	Country	Total armed forces
1.	China	2,255,000
2.	United States	1,456,850
3.	India	1,325,000
4.	Russia	1,058,000
5.	Korea, South	687,000
6.	Pakistan	620,000
7.	Iran	540,000

countries: {Russia, ..., Germany, Korea, South, Pakistan, Iran}

Example: Table Set Expansion

countries: {Russia, ..., Germany, Korea, South, Pakistan, Iran}

Countries and dependencies

Rank	Country	To km
—	*World*	51 (196
1	Russia	1 (6
—	*Antarctica*	1 (5
2	Canada	(3
3	China	(3
4	America	(3

countries: {
Russia,...,
World,
Antarctica,
America}

Example: Table Set Expansion

countries: {
Russia,...,
World,
Antarctica,
America}

All continents:
Antarctica
Africa
Asia
America
Australia
Europe

Example: Table Set Expansion

countries: {
Russia,...,
World,
Antarctica,
America}

All continents:
Antarctica
Africa
Asia
America
Australia
Europe

Semantic Drift may occur

Summary: Set Expansion

Set Expansion extends a set of instance names. We saw 2 methods:

1. Recursively applied patterns

X, Y, and Z

2. Table Set Expansion

Richest Countries in the World

view as: list / slideshow / map

▲	Country	GDP
1.	United States	$15,290,000,000,000
2.	China	$11,440,000,000,000
3.	India	$4,515,000,000,000
4.	Japan	$4,497,000,000,000
5.	Germany	$3,139,000,000,000
6.	Russia	$2,414,000,000,000

Summary: Is-A Extraction

Is-a finds names of instance/class or subclass/superclass pairs.

We saw 2 methods:

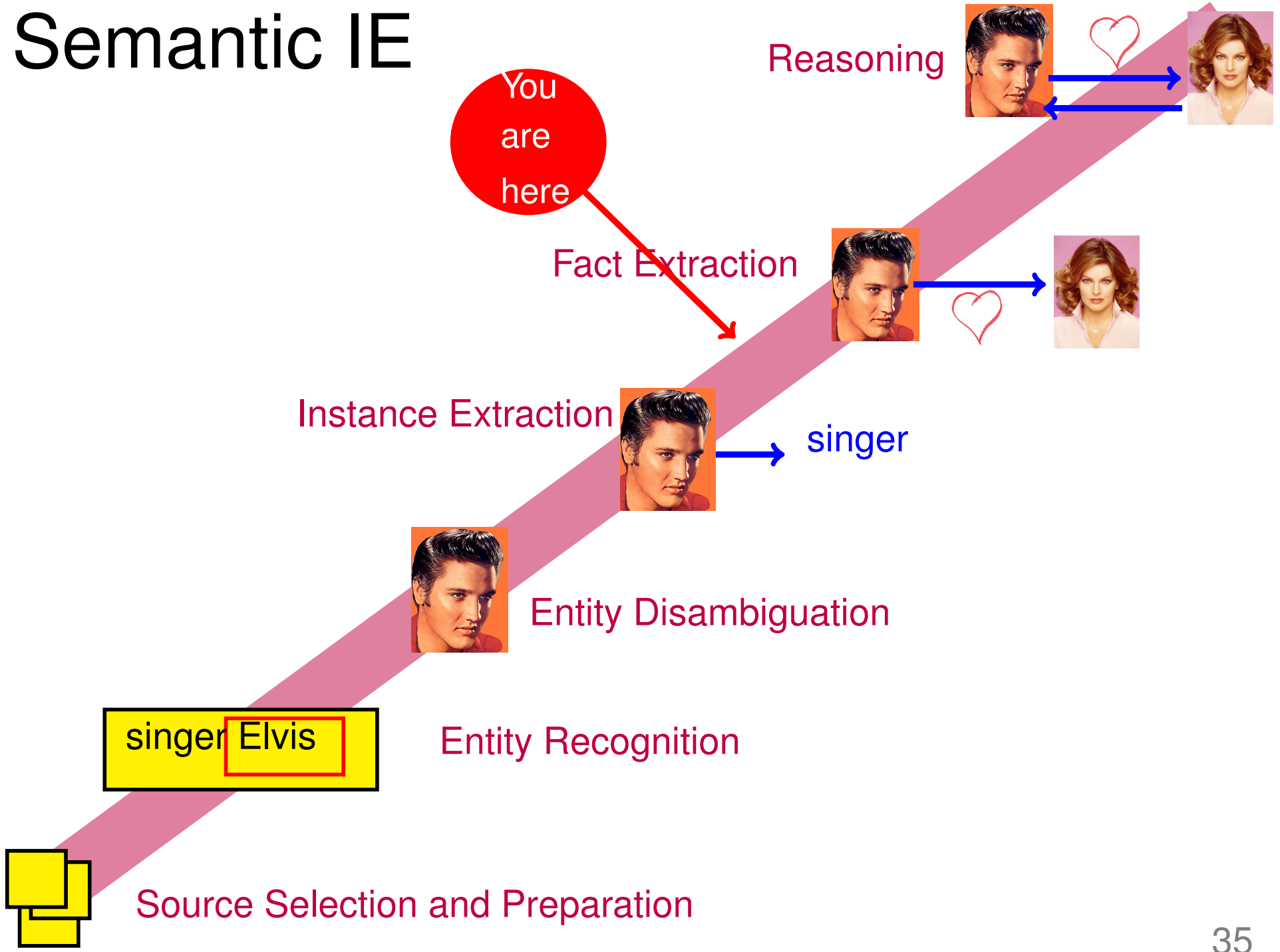
1. Hearst Patterns

   vegetarians such as Lisa

2. Set Expansion

   cities: {Chicago, Springfield}

Semantic IE


Reasoning
You are here
Fact Extraction
Instance Extraction
singer
Entity Disambiguation
singer Elvis
Entity Recognition
Source Selection and Preparation

References

Marti Hearst: Automatic Acquisition of Hyponyms

Learning Arguments and Supertypes of Semantic Relations

Knowledge Harvesting from Text and Web Sources

-$>$dipre