USC

USC Viterbi
School of Engineering

Automatically Constructing Semantic Web Services from Online Sources

Craig A. Knoblock

José Luis Ambite, Sirish Darbha, Aman Goel,
Kristina Lerman, Rahul Parundekar, and Tom Russ

University Southern California

Goal

USC Viterbi School of Engineering

- Automatically build semantic models for data and services available on the larger Web
- Construct models of these sources that are sufficiently rich to support querying and integration
  - Such models would make the existing semantic web tools and techniques more widely applicable
- Current focus:
  - Build models for the vast amount of structured and semi-structured data available
    - *Not just web services, but also form-based interfaces*
    - *E.g., Weather forecasts, flight status, stock quotes, currency converters, online stores, etc.*
  - Learn models for information-producing web sources and web services

Approach

USC Viterbi School of Engineering

- Start with an some initial knowledge of a domain
  - Sources and semantic descriptions of those sources
- Automatically
  - Discover related sources
  - Determine how to invoke the sources
  - Learn the syntactic structure of the sources
  - Identify the semantic types of the data
  - Build semantic models of the source
  - Construct semantic web services

Outline

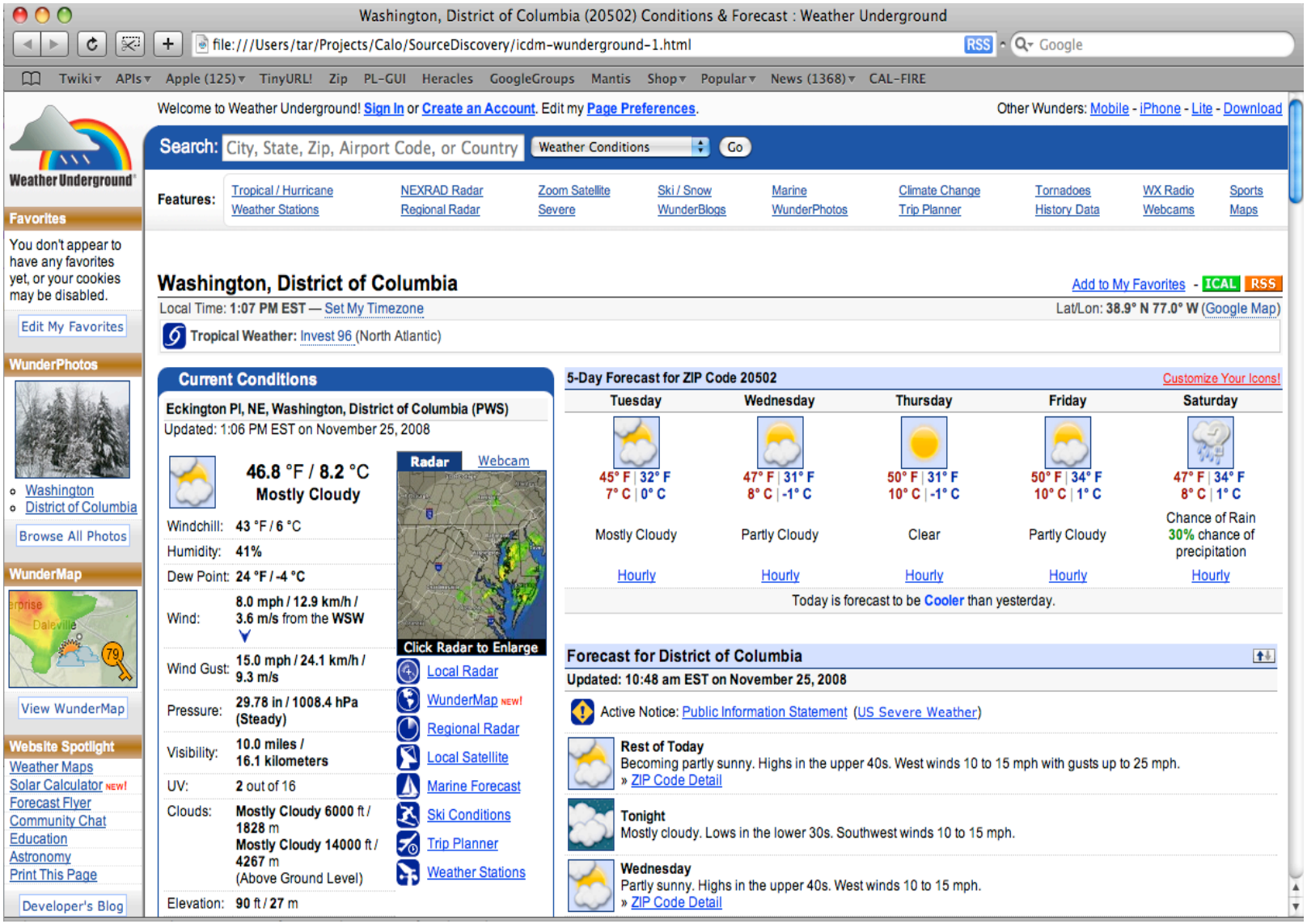
USC Viterbi School of Engineering

- Integrated Approach
  - Discovering related sources
  - Constructing syntactic models of the sources
  - Determining the semantic types of the data
  - Building semantic models of the sources
- Experimental Results
- Related Work
- Discussion

USC

Seed Source

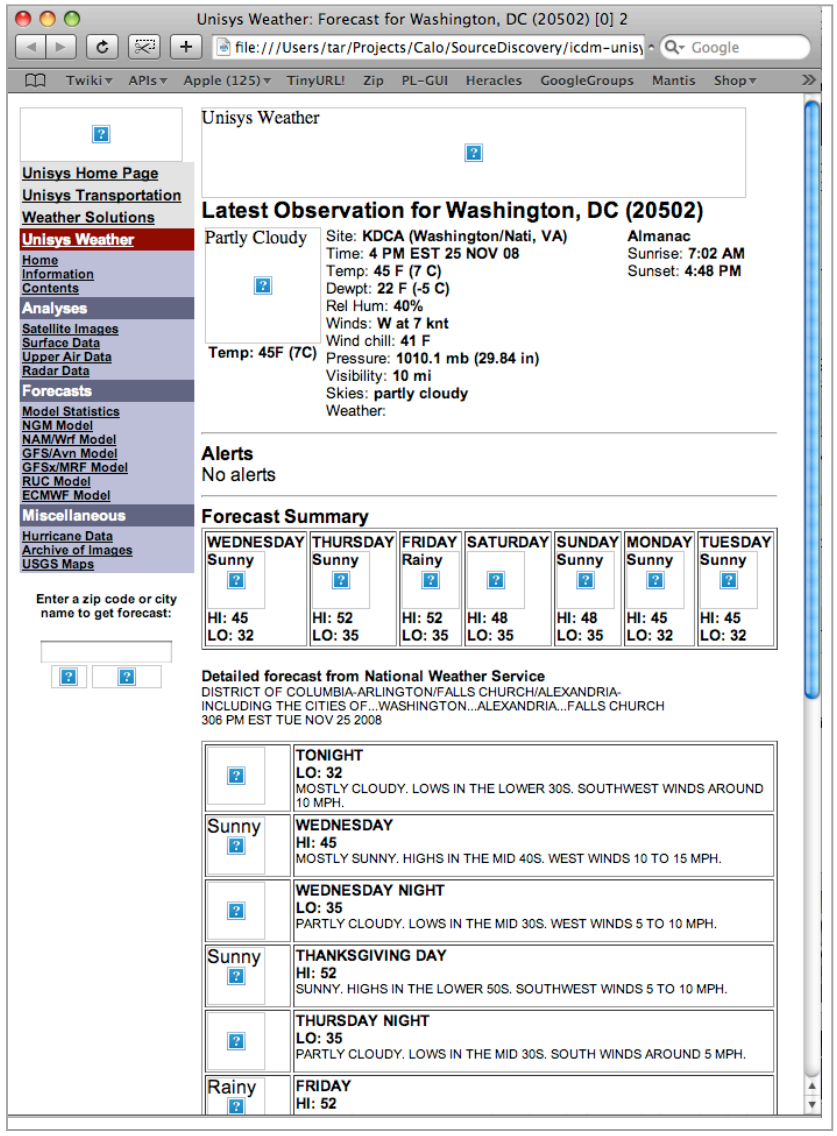
USC Viterbi School of Engineering



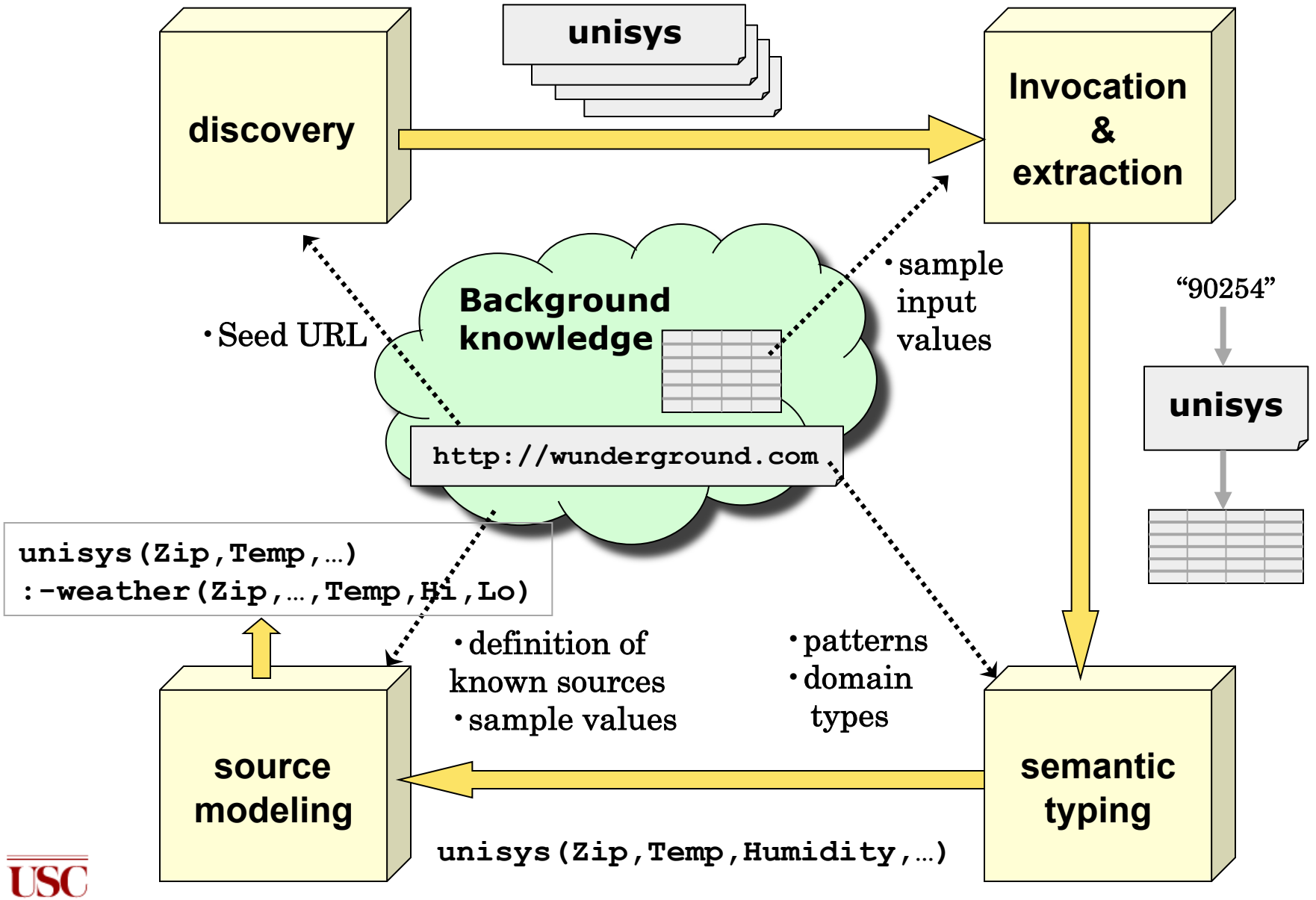
USC

Automatically Discover and Build Semantic Web Services for Related Sources

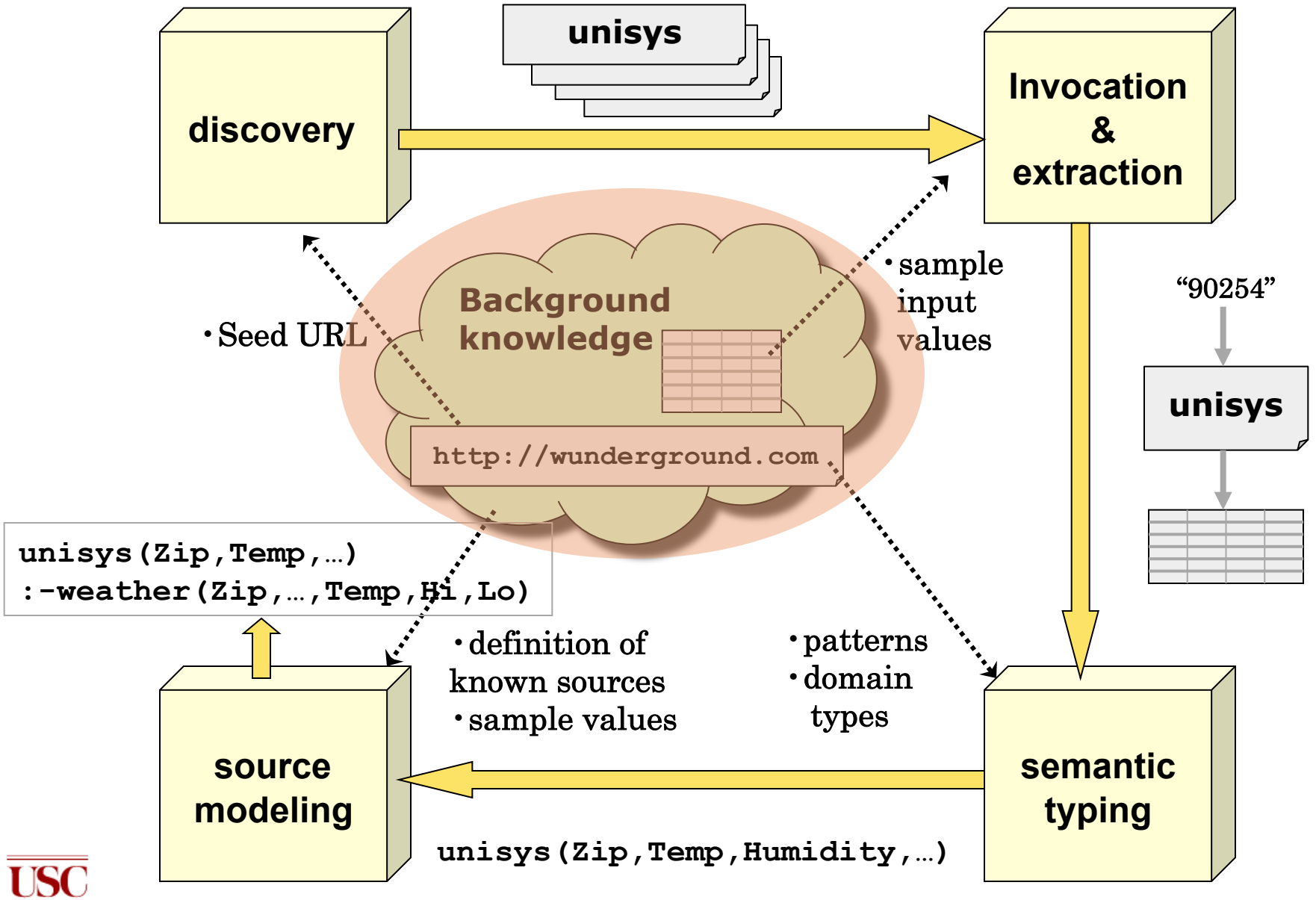
USC Viterbi School of Engineering



Integrated Approach



Background Knowledge



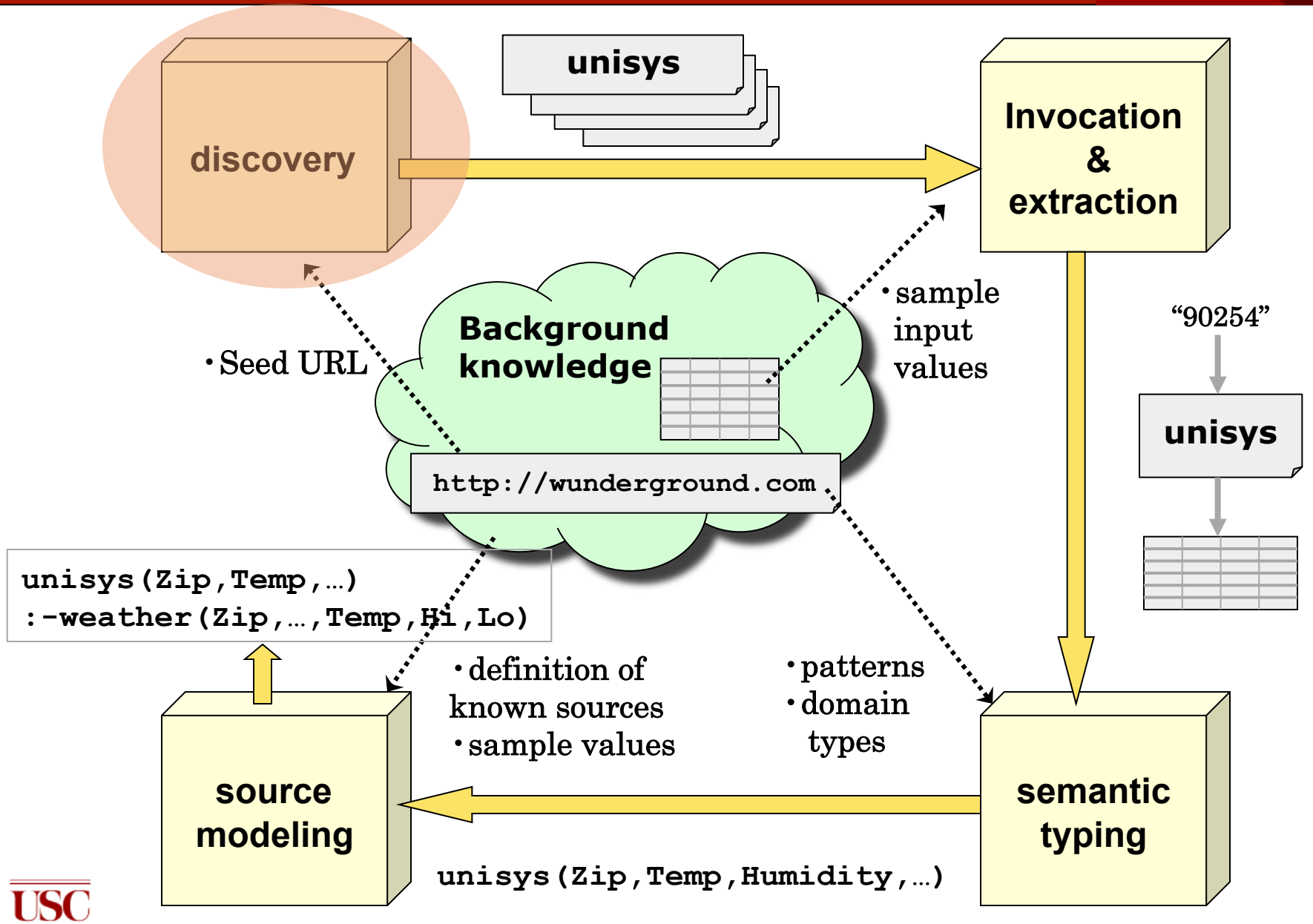
Background Knowledege

USC Viterbi School of Engineering

- Ontology of the inputs and outputs
  - e.g., TempF, Humidity, Zipcode;
- Sample values for each semantic type
  - e.g., "88 F" for TempF, and "90292" for Zipcode
- Domain input model
  - a weather source may accept Zipcode or City and State as input
  - Sample input values
- Known sources (seeds)
  - e.g., http://wunderground.com
- Source descriptions in Datalog or RDF
  - wunderground($Z,CS,T,F0,S0,Hu0,WS0,WD0,P0,V0,FL1,FH1,S1,FL2,FH2,S2,
    FL3,FH3,S3,FL4,FH4,S4,FL5,FH5,S5) :-
    weather(0,Z,CS,D,T,F0,_,_,S0,Hu0,P0,WS0,WD0,V0)
    weather(1,Z,CS,D,T,_,FH1,FL1,S1,_,_,_,_,_),
    weather(2,Z,CS,D,T,_,FH2,FL2,S2,_,_,_,_,_),
    weather(3,Z,CS,D,T,_,FH3,FL3,S3,_,_,_,_,_),
    weather(4,Z,CS,D,T,_,FH4,FL4,S4,_,_,_,_,_),
    weather(5,Z,CS,D,T,_,FH5,FL5,S5,_,_,_,_,_).

USC

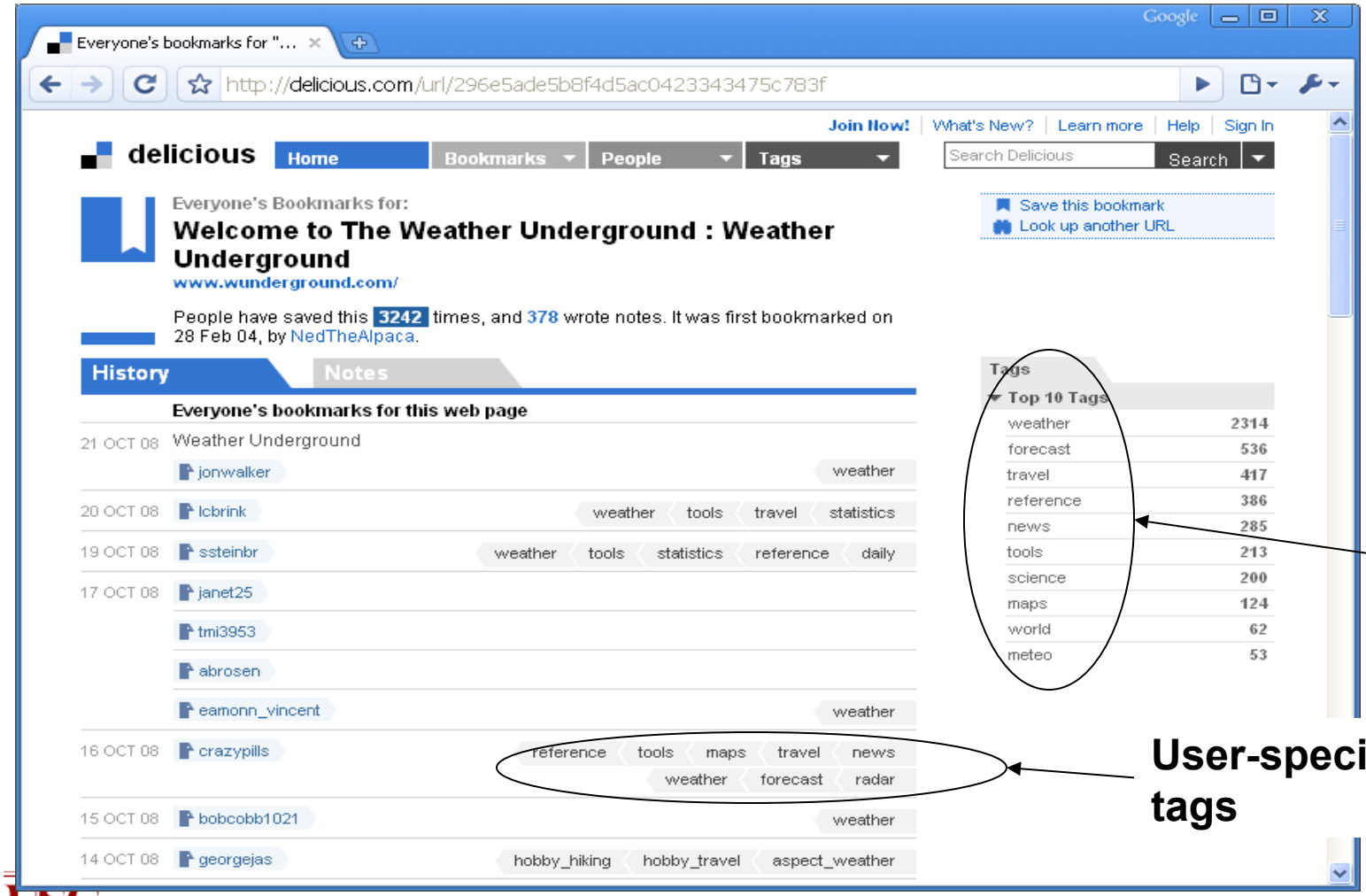
Source Discovery



Source Discovery [Plangprasopchok and Lerman]

USC Viterbi School of Engineering

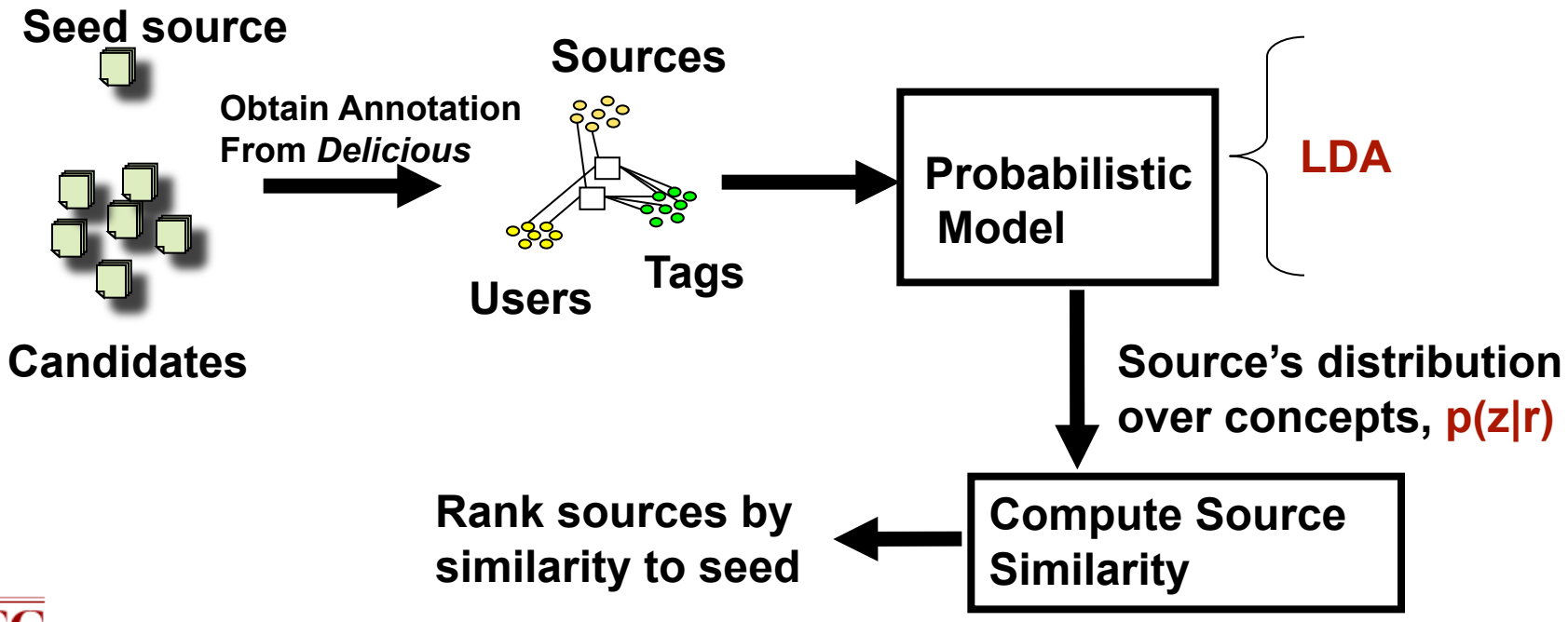
- Leverage user-generated tags on the social bookmarking site del.icio.us to discover sources similar to the seed



Exploiting Social Annotations for Resource Discovery

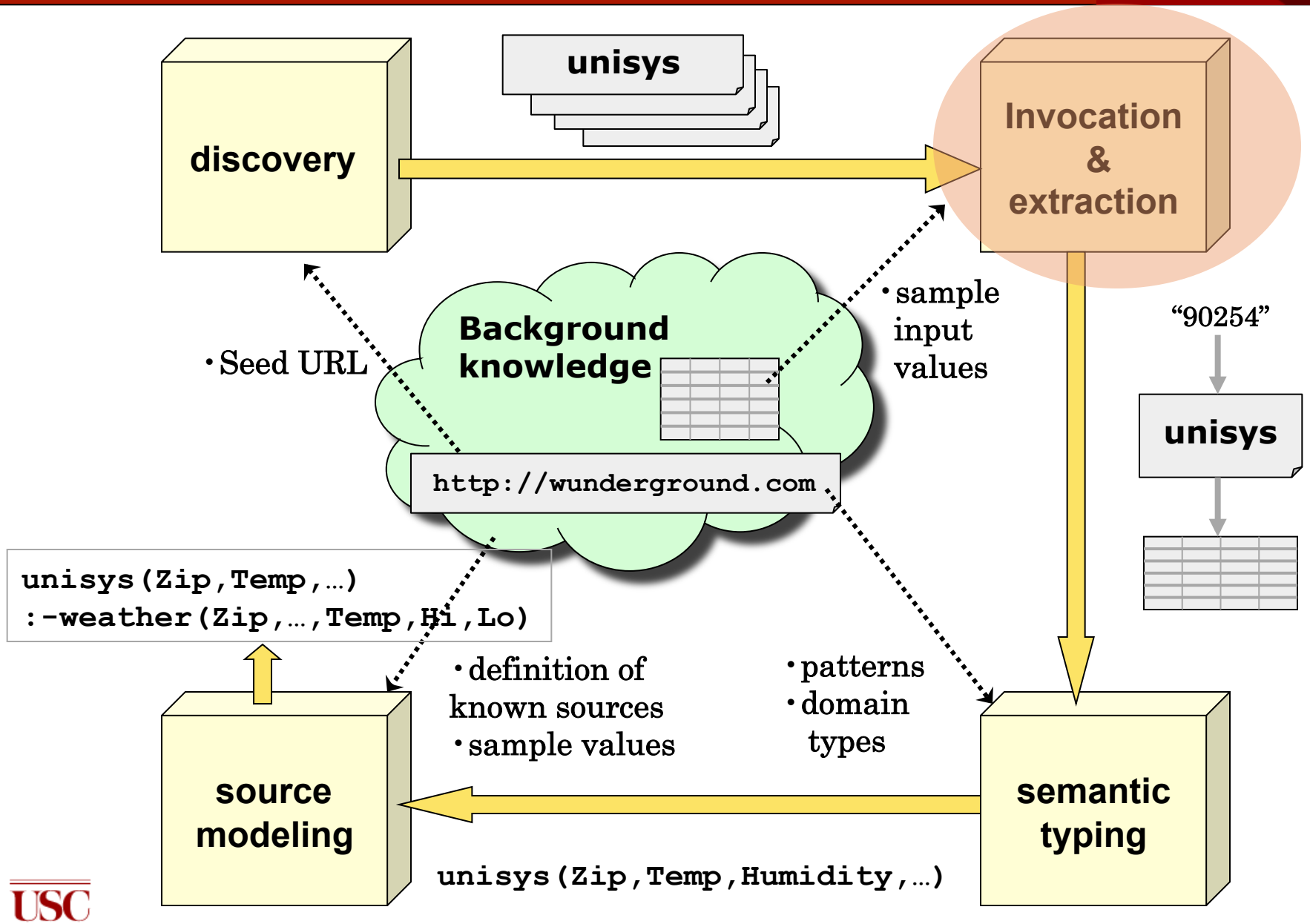
USC Viterbi School of Engineering

- Resource discovery task : "*given a seed source, find other most similar sources*"
  - Gather a corpus of <user, source, tag> bookmarks from del.icio.us
  - Use probabilistic modeling to find hidden topics in the corpus
  - Rank sources by similarity to the seed within topic space



USC

Source Invocation & Extraction



Target Source Invocation

USC Viterbi School of Engineering

- To invoke the target source, we need to locate the form and determine the appropriate input values
  1. Locate the form
  2. Try different data type combinations as input
     - *For weather, only one input - location, which can be zipcode or city/state*
  3. Submit Form
  4. Keep successful invocations

Form Input

USC

Inducing Extraction Templates

USC Viterbi School of Engineering

- Template: a sequence of alternating slots and stripes
  - stripes are the common substrings among all pages
  - slots are the placeholders for data
- Induction: Stripes are discovered using the Longest Common Subsequence algorithm

Sample Page 1

```
<img src="images/Sun.png" alt="Sunny"><br>
<font face="Arial, Helvetica, sans-serif">
 <small><b>Temp: 72F (22C)</b></small></font>
<font face="Arial, Helvetica, sans-serif">
 <small>Site: <b>KSMO (Santa_Monica_Mu, CA)</b><br>
      Time: <b>11 AM PST 10 DEC 08</b>
```

Sample Page 2

```
<img src="images/Clouds.png" alt="Cloudy"><br>
<font face="Arial, Helvetica, sans-serif">
 <small><b>Temp: 37F (2C)</b></small></font>
<font face="Arial, Helvetica, sans-serif">
 <small>Site: <b>KAGC (Pittsburgh/Alle, PA)</b><br>
      Time: <b>2 PM EST 10 DEC 08</b>
```

Induction

Template

```
<img src="images/*.png" alt="*"><br>
<font face="Arial, Helvetica, sans-serif">
 <small><b>Temp: * (*)</b></small></font>
<font face="Arial, Helvetica, sans-serif">
 <small>Site: <b>* (*, *)</b><br>
      Time: <b>* 10 DEC 08</b>
```

Slot

Stripe

USC

Data Extraction with Templates

USC Viterbi School of Engineering

- To extract data: Find data in slots by locating the stripes of the template on unseen page:

Unseen Page

```
<img src="images/Sun.png" alt="Sunny"><br>
<font face="Arial, Helvetica, sans-serif">
  <small><b>Temp: 71F (21C)</b></small></font>
<font face="Arial, Helvetica, sans-serif">
  <small>Site: <b>KCQT (Los_Angeles_Dow, CA)</b><br>
       Time: <b>11 AM PST 10 DEC 08</b>
```

Induced Template

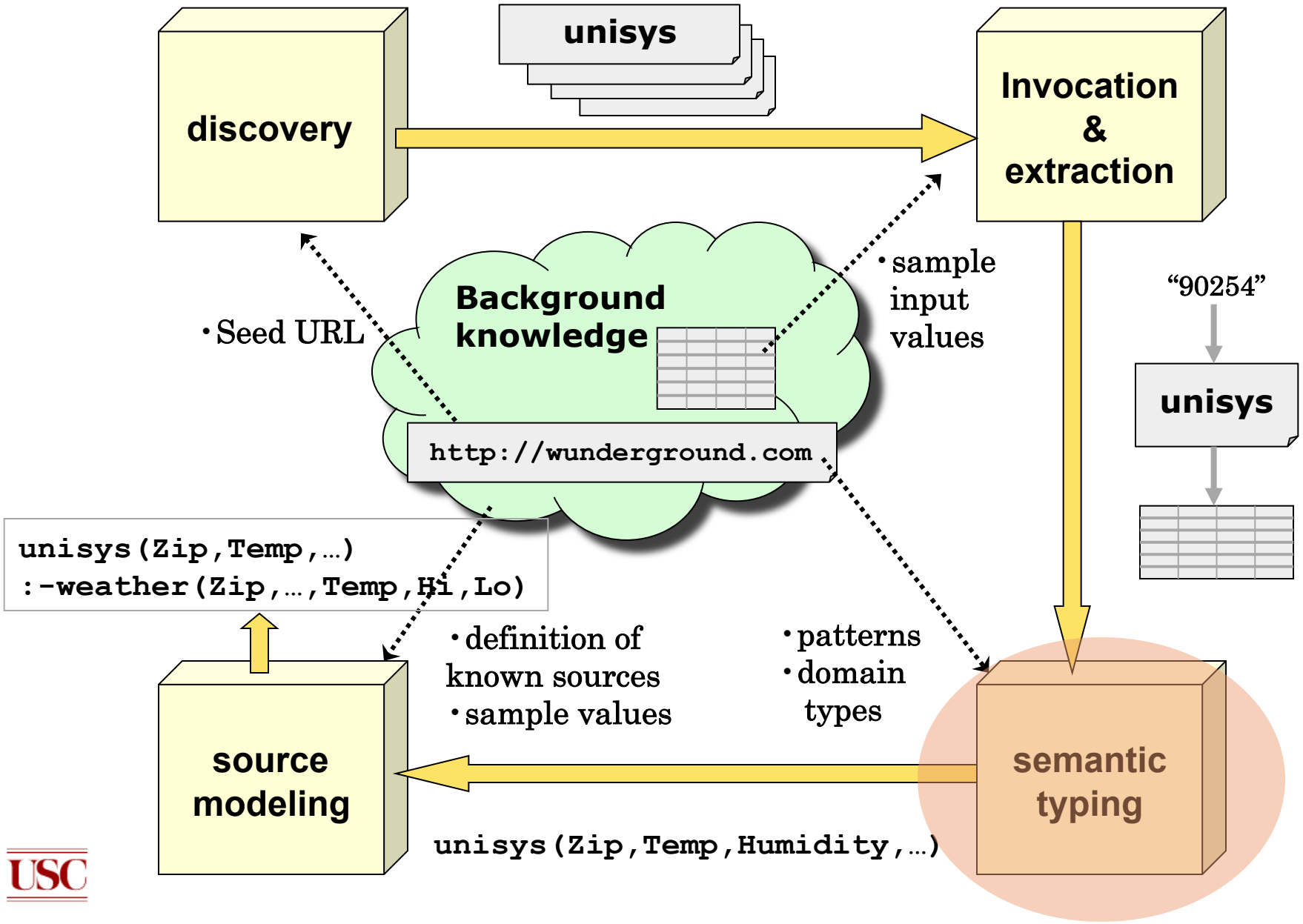
```
<img src="images/*.png" alt="*"><br>
<font face="Arial, Helvetica, sans-serif">
  <small><b>Temp: * (*)</b></small></font>
<font face="Arial, Helvetica, sans-serif">
  <small>Site: <b>* (*, *)</b><br>
       Time: <b>* 10 DEC 08</b>
```

Extracted Data

Sun	Sunny	71F	21C	KCQT	Los_Angeles_Dow	CA	11 AM PST

USC

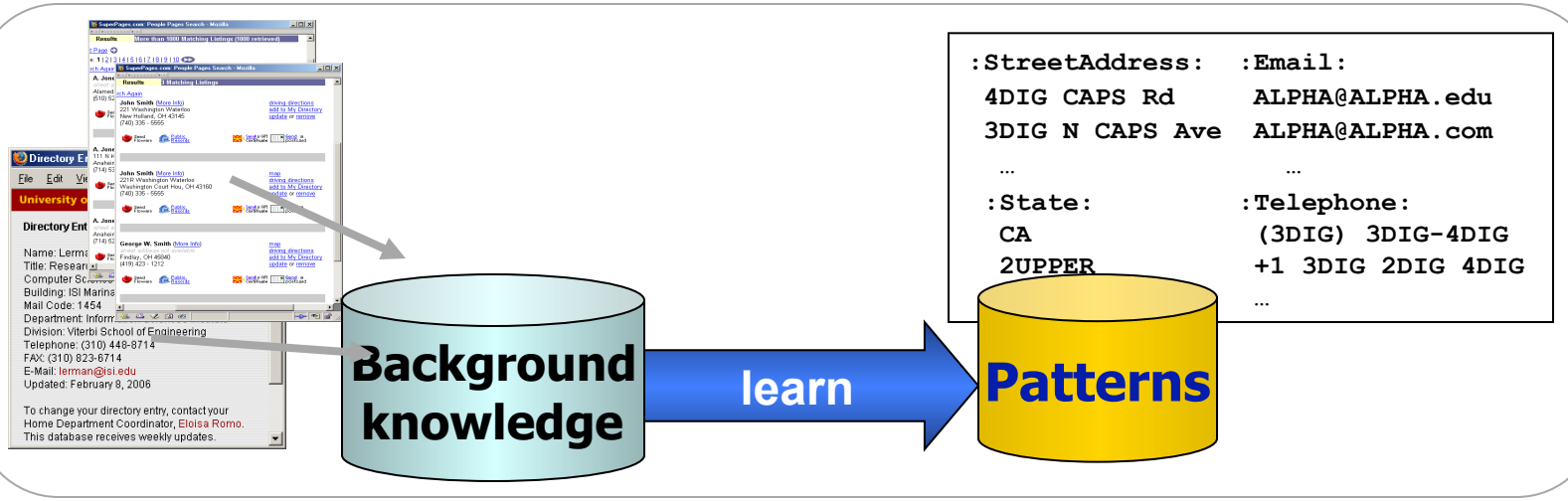
Semantic Typing



Semantic Typing [Lerman, Plangprasopchok, & Knoblock]

USC Viterbi School of Engineering

✓**Idea: Learn a model of the content of data and use it to recognize new examples**



Person	Address	Work
E Lewis	3518 Hilltop Rd	(419) 531 - 0504
Andrew Lewis	3543 Larchmont Pkwy	(518) 474 -
C. S. Lewis	555 Willow Run Dr	(612) 578 -
Carmen Jones	355 Morgan Ave N	(612) 522 -
John Jones	3574 Brookside Rd	(555) 531 - 9566
Location	**State_prov**	**Postal_code**
Toledo	OH	64325-3000
Toledo	OH	64356
Seattle	WA	8422
Seattle	WA	8435
Omaha	NE	52456-6444

label

:FullName:	:StreetAddress:	:Telephone:
E Lewis	3518 Hilltop Rd	(419) 531 - 0504
drew Lewis	3543 Larchmont Pkwy	(518) 474 - 4799
Lewis	555 Willow Run Dr	(612) 578 - 5555
rmen Jones	355 Morgan Ave N	(612) 522 - 5555
John Jones	3574 Brookside Rd	(555) 531 - 9566
:City:	**:State:**	**:Zipcode:**
Toledo	OH	64325-3000
Toledo	OH	64356
Seattle	WA	8422
Seattle	WA	8435
Omaha	NE	52456-6444

USC

Labeling New Data

USC Viterbi School of Engineering

- Use learned patterns to link new data to types in the ontology
  - Score how well patterns describe a set of examples
    - *Number of matching patterns*
    - *How many tokens of the example match pattern*
    - *Specificity of the matched patterns*
  - Output top-scoring types

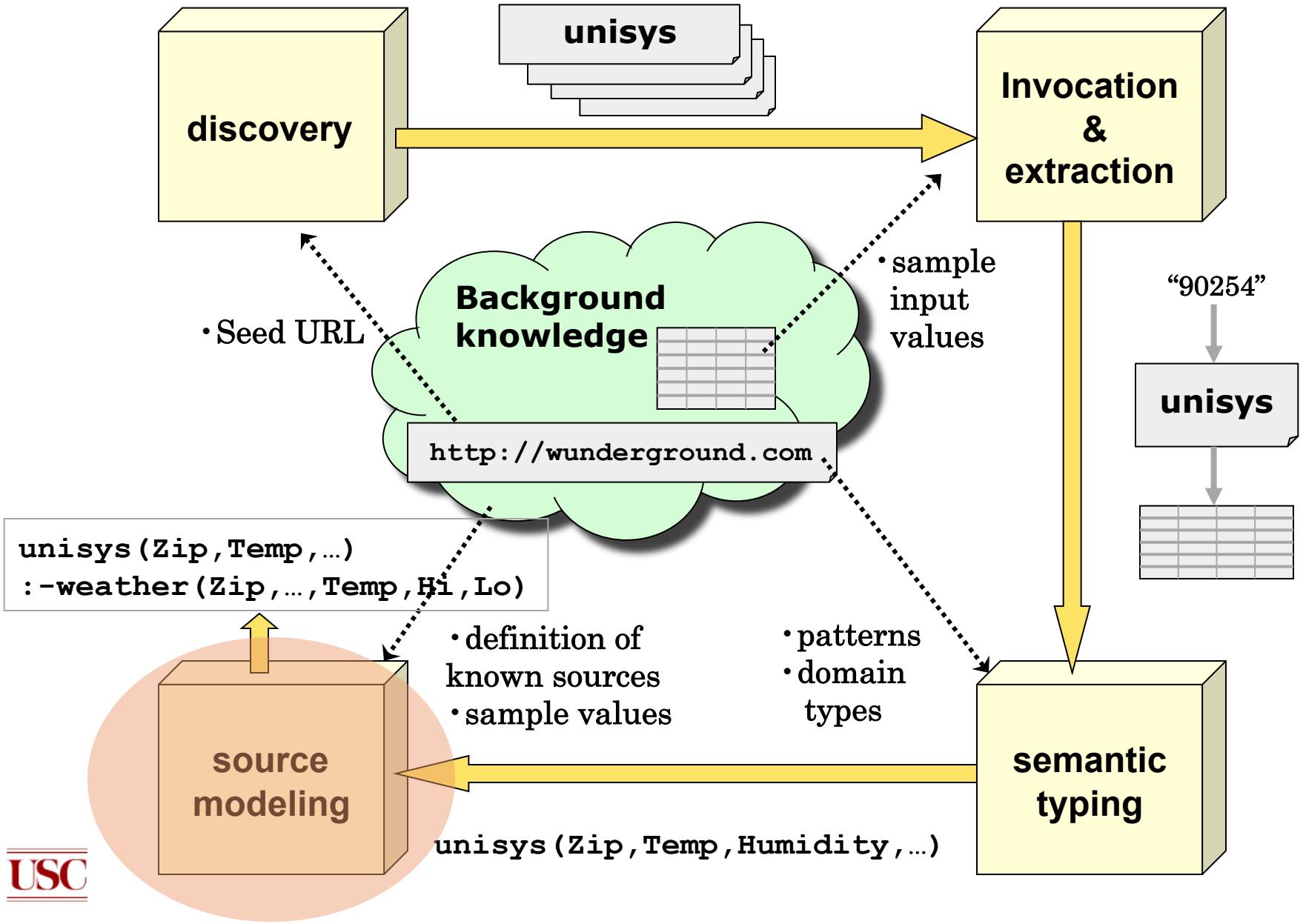
Person	Address	Work
E Lewis	3518 Hilltop Rd	(419) 531 - 0504
Andrew Lewis	3543 Larchmont Pkwy	(518) 474 - 4799
C. S. Lewis	555 Willow Run Dr	(612) 578 - 5555
Carmen Jones	355 Morgan Ave N	(612) 522 - 5555
John Jones	3574 Brookside Rd	(555) 531 - 9566
Location	**State_prov**	**Postal_code**
Toledo	OH	64325-3000
Toledo	OH	64356
Seattle	WA	8422
Seattle	WA	8435
Omaha	NE	52456-6444

patterns

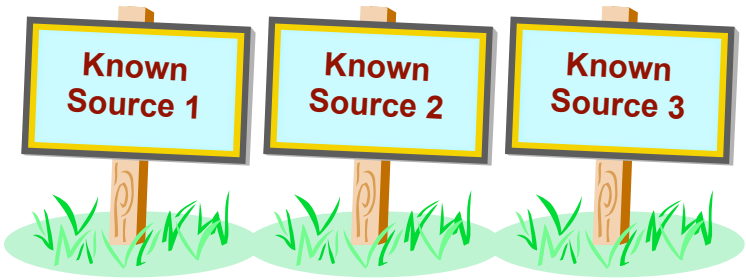
```
:StreetAddress:   :Email:
 4DIG CAPS Rd      ALPHA@ALPHA.edu
 3DIG N CAPS Ave   ALPHA@ALPHA.com
  …                  …
 :State:           :Telephone:
  CA                (3DIG) 3DIG-4DIG
  2UPPER            +1 3DIG 2DIG 4DIG
  …                 …
```

USC

Source Modeling [Carman & Knoblock]

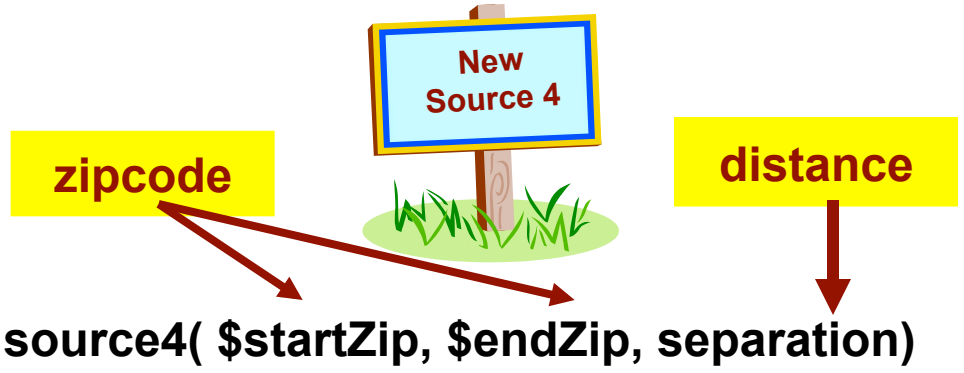


Inducing Source Definitions

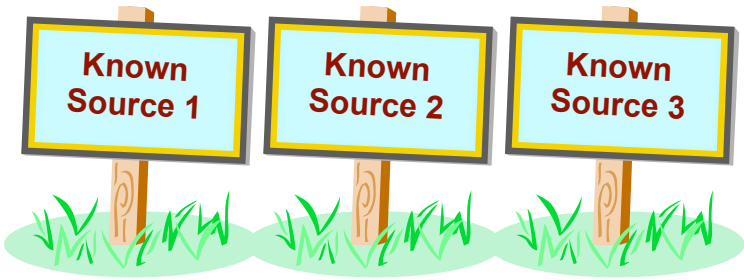



```
source1($zip, lat, long) :-
    centroid(zip, lat, long).

source2($lat1, $long1, $lat2, $long2, dist) :-
    greatCircleDist(lat1, long1, lat2, long2, dist).

source3($dist1, dist2) :-
    convertKm2Mi(dist1, dist2).
```

- Step 1: classify input & output semantic types



Generating Plausible Definition




```
source1($zip, lat, long) :-
    centroid(zip, lat, long).

source2($lat1, $long1, $lat2, $long2, dist) :-
    greatCircleDist(lat1, long1, lat2, long2, dist).

source3($dist1, dist2) :-
    convertKm2Mi(dist1, dist2).
```

- Step 1: classify input & output semantic types
- Step 2: generate plausible definitions

```
source4($zip1, $zip2, dist):-
    source1(zip1, lat1, long1),
    source1(zip2, lat2, long2),
    source2(lat1, long1, lat2, long2, dist2),
    source3(dist2, dist).
```

```
source4($zip1, $zip2, dist):-
    centroid(zip1, lat1, long1),
    centroid(zip2, lat2, long2),
    greatCircleDist(lat1, long1, lat2, long2, dist2),
    convertKm2Mi(dist1, dist2).
```

Invoke and Compare the Definition

USC Viterbi School of Engineering

- Step 1: classify input & output semantic types
- Step 2: generate plausible definitions
- Step 3: invoke service & compare output

```
source4($zip1, $zip2, dist):-
    source1(zip1, lat1, long1),
    source1(zip2, lat2, long2),
    source2(lat1, long1, lat2, long2, dist2),
    source3(dist2, dist).
```

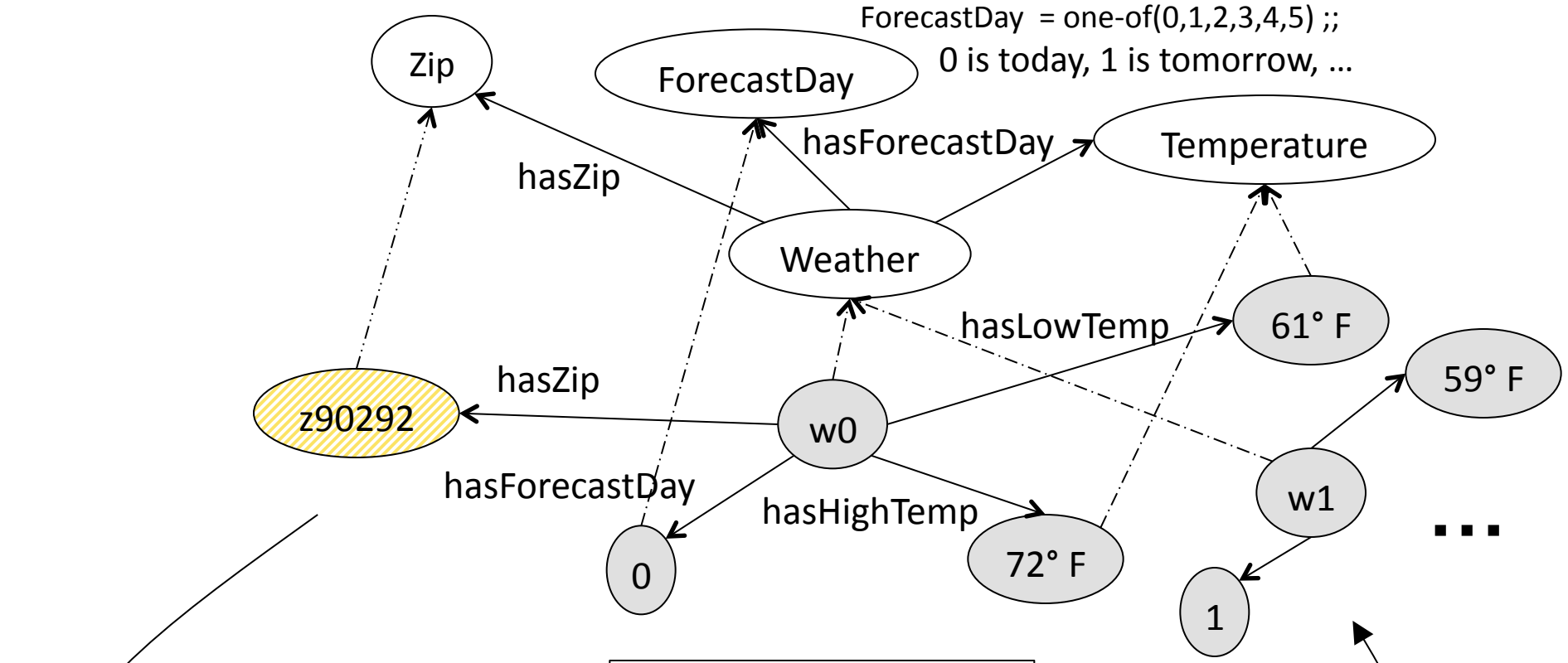
```
source4($zip1, $zip2, dist):-
    centroid(zip1, lat1, long1),
    centroid(zip2, lat2, long2),
    greatCircleDist(lat1, long1, lat2, long2,dist2),
    convertKm2Mi(dist1, dist2).
```

match

$zip1	$zip2	dist *(actual)*	dist *(predicted)*
80210	90266	842.37	843.65
60601	15201	410.31	410.83
10005	35555	899.50	899.21

Constructing the Semantic Web Service

USC Viterbi School of Engineering



z90292 hasName 90292 .

DEIMOS generated Web Service

w1 hasZIP z90292 .
w1 hasTemp 61° F .
…
w1 hasZIP z90292 .
w2 hasLowTemp 59° F .

Legend:

ontology

RDF Input

RDF output

Background Source Descriptions

USC Viterbi School of Engineering

```
wunderground( $Z,CS,T,F0,C0,S0,Hu0,WS0,WD0,P0,V0,FL1,FH1,S1,
                    FL2,FH2, S2,FL3,FH3,S3,FL4,FH4,S4,FL5,FH5,S5):-
   Weather(_w0),hasForecastDay(_w0,0),hasZIP(_w0,Z),
      hasCityState(_w0,CS),hasTimeWZone(_w0,T),
      hasCurrentTemperatureFarenheit(_w0,F0),
      hasCurrentTemperatureCentigrade(_w0,C0),
      hasSkyConditions(_w0,S0),hasHumidity(_w0,Hu0),
      hasPressure(_w0,P0), hasWindSpeed(_w0,_ws1),
      WindSpeed(_ws1), hasWindSpeedInMPH(_ws1,WS0),
      hasWindDir(_ws1,WD0), hasVisibilityInMi(_w0,V0),
   Weather(_w1), hasForecastDay(_w1,1), hasZIP(_w1,Z),
      hasCityState(_w1,CS), hasLowTemperatureFarenheit(_w1,FL1),
      hasHighTemperatureFarenheit(_w1,FH1), hasSkyConditions(_w1,S1),
   …

convertC2F($C,F) :- centigrade2farenheit(C,F)
```

USC

Target explained using background sources

USC Viterbi School of Engineering

```
unisys($Z,_,_,_,_,_,_,_,_,F9,_,C,_,F13,F14,Hu,_,F17,_,_,_,_,S22,_,S24,
        _,_,_,_,_,_,_,_,_,_,_,S35,S36,_,_,_,_,_,_,_,_,_,_) :-
    wunderground(Z,_,_,F9,_,Hu,_,_,_,_,F14,F17,S24,_,_,S22,_,_,
                 S35,_,_,S36,F13,_,_),
    convertC2F(C,F9)
```

USC

Learned Target Source Description

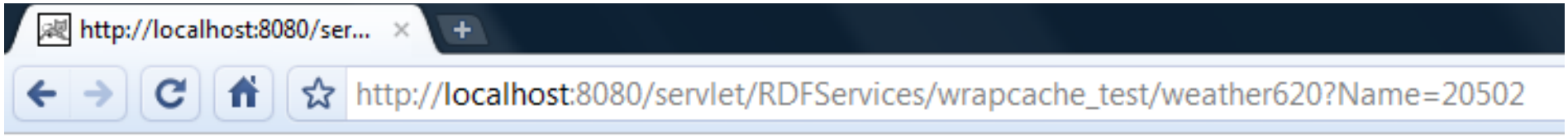
USC Viterbi School of Engineering

```
unisys($Z,_,_,_,_,_,_,_,F9,_,C,_,F13,F14,Hu,_,F17,_,_,_,_,S22,_,S24,_,_,_,
        _,_ ,_,_,_,_,_,_,S35,S36,_,_,_,_,_,_,_,_,_,_) :-
   Weather(_w0),hasForecastDay(_w0,0),hasZIP(_w0,Z),
     hasCurrentTemperatureFarenheit(_w0,F9), centigrade2farenheit(C,F9),
     hasCurrentTemperatureCentigrade(_w0,C), hasHumidity(_w0,Hu0),
  Weather(_w1),hasForecastDay(_w1,1), hasZIP(_w1,Z),
     hasCityState(_w1,CS), hasTimeWZone(_w1,T),
     hasLowTemperatureFarenheit(_w1,F14),
     hasHighTemperatureFarenheit(_w1,F17), hasSkyConditions(_w1,S24),
  Weather(_w2),hasForecastDay(_w2,2), hasZIP(_w2,Z),
     hasSkyConditions(_w2,S22),
  Weather(_w3),hasForecastDay(_w3,3), hasZIP(_w3,Z),
     hasSkyConditions(_w3,S35),
  Weather(_w4),hasForecastDay(_w4,4), hasZIP(_w4,Z),
     hasSkyConditions(_w4,S36),
  Weather(_w5),hasForecastDay(_w5,5), hasZIP(_w5,Z),
      hasLowTemperatureFarenheit(_w5,F13).
```

USC

Web Service Invocation

USC Viterbi School of Engineering




```
weatherforecast7709080 rdf:type WeatherForecast .
weatherforecast7709080 hasZIP "20502" .
weatherforecast7709080 hasCurrentTemperatureFarenheit "71F" .
windspeed7365415 rdf:type WindSpeed .
weatherforecast7709080 hasWindSpeed windspeed7365415 .
weatherforecast8455262 rdf:type WeatherForecast .
weatherforecast8455262 hasLowTemperature "49 F (9 C)" .
weatherforecast3087280 rdf:type WeatherForecast .
weatherforecast3087280 hasHighTemperature "71 F (21 C)" .
```

USC

Outline

USC Viterbi School of Engineering

- Integrated Approach
  - Discovering related sources
  - Constructing syntactic models of the sources
  - Determining the semantic types of the data
  - Building semantic models of the sources
- Experimental Results
- Related Work
- Discussion

USC

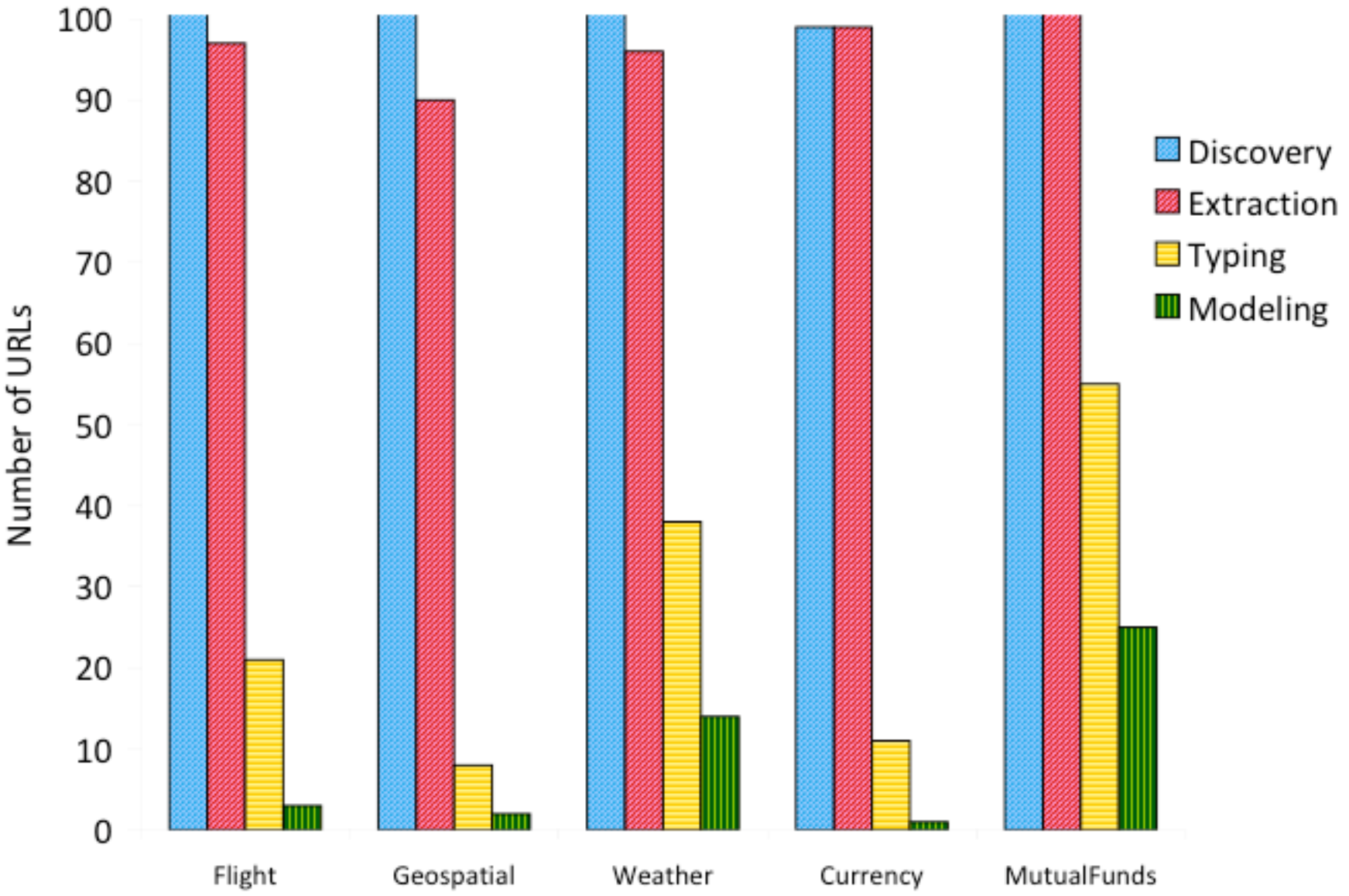
Experimental Evaluation

USC Viterbi School of Engineering

- Experiments in 5 domains
  - Flight – lookup the current status of a flight
  - Geospatial – map streeet addresses into lat/long coordinates
  - Weather – find the current and forecasted weather
  - Currency – convert between various currencies
  - Mutual Funds – look up current data on a mutual fund
- Evaluation:
  - 1) Can the system correctly learn a model for those sources that perform the same task
  - 2) What is the precision and recall of the attributes in the model

Candidate Sources after Each Step

USC Viterbi School of Engineering



USC

Evaluation of the Models

USC Viterbi School of Engineering

domain	Precision	Recall	F_1-measure
weather	0.64	0.29	0.39
geospatial	1.00	0.86	0.92
flights	0.69	0.35	0.46
currency	1.00	1.00	1.00
mutualfund	0.72	0.30	0.42

Outline

USC Viterbi School of Engineering

- Integrated Approach
  - Discovering related sources
  - Constructing syntactic models of the sources
  - Determining the semantic types of the data
  - Building semantic models of the sources
- Experimental Results
- Related Work
- Discussion

USC

Related Work

USC Viterbi School of Engineering

- ILA & Category Translation (Perkowitz & Etzioni 1995)
  - Learn functions describing operations on internet
  - Assumes single input and single tuple as output
- Metadata-based classification of data types used by Web services and HTML forms (Hess & Kushmerick, 2003)
  - Naïve Bayes classifier
  - Only classified the source type, no model
- Use NLP to learn source descriptions (Afzal et al, 2009)
  - Extract type and function provided by service
  - Only provides high-level service type (ex: algorithm, application, data)
- Mining existing workflows (Belhajjame et al, 2008)
  - Connections in parameters of workflows use to infer semantic types
  - Limited semantic description of a web service

Outline

USC Viterbi School of Engineering

- Integrated Approach
  - Discovering related sources
  - Constructing syntactic models of the sources
  - Determining the semantic types of the data
  - Building semantic models of the sources
- Experimental Results
- Related Work
- Discussion

USC

Discussion

USC Viterbi School of Engineering

- Integrated approach to discovering and modeling online sources and services:
  - *Discover new sources*
  - *How to invoke a source*
  - *Discovering the template for the source*
  - *Finding the semantic types of the output*
  - *Learning a definition of what the service does*
- Provides an approach to generate services and data for the Semantic Web
  - Little motivation for providers to annotate services
  - Instead we can generate metadata automatically

Future Work

USC Viterbi School of Engineering

- Coverage, Precision, & Recall
  - Difficult to invoke sources with many inputs
    - *Hotel reservation sites*
  - Hard to learn sources that have many attributes
    - *Some weather sources could have 40 attributes*
- Learning beyond the domain model
  - Learn new semantic types
    - *Discover barometric pressure*
  - Learn new source attributes
    - *Learn about 6-day high and low temperatures*
  - Learn new source relations
    - *Learn conversion between Fahrenheit and Celsius*
  - Learn the domain and range of the sources
    - *Learn that a source provides world weather vs. US weather*
- Linking the Deep Web to the Linked Data Web
  - Use linked data ontologies as domain model
  - Perform entity linkage from web source URI to linked data URI

USC

Acknowledgements & Papers

- Sponsors
  - DARPA CALO Program, AFOSR, & NSF

- Papers
  - Integrated Approach
    - *[Ambite, Darbha, Goel, Knoblock, Lerman, Parundekar, Russ, ISWC 2009]*
  - Source discovery
    - *[Plangprasopchok and Lerman, WWW, 2009]*
  - Source extraction
    - *[Gazen, CMU Ph.d. thesis, 2008]*
  - Semantic typing
    - *[Lerman, Plangprasopchok, & Knoblock, IJSWIS, 2008]*
  - Source modeling
    - *[Carman & Knoblock, JAIR, 2007]*