Contextual Media Retrieval Using Natural Language Queries

IMPRS-CS PhD Application Talk

Sreyasi Nag Chowdhury

Master's Thesis Supervisors
Dr. Mario Fritz
Dr. Andreas Bulling

Adviser
M.Sc. Mateusz Malinowski

UNIVERSITÄT DES SAARLANDES
max planck institut informatik
CLUSTER OF EXCELLENCE
PERCEPTUAL USER INTERFACES
perceptual.mpi-inf.mpg.de
Scalable Learning Perception
scalable.mpi-inf.mpg.de

Outline

- Motivation and Overview
- Contextual Media Retrieval System
- Results and Conclusion

Motivation

"Collective Memory"
of media content

Spatio-temporal
exploration of media
on wearable devices

System Overview

Demonstration

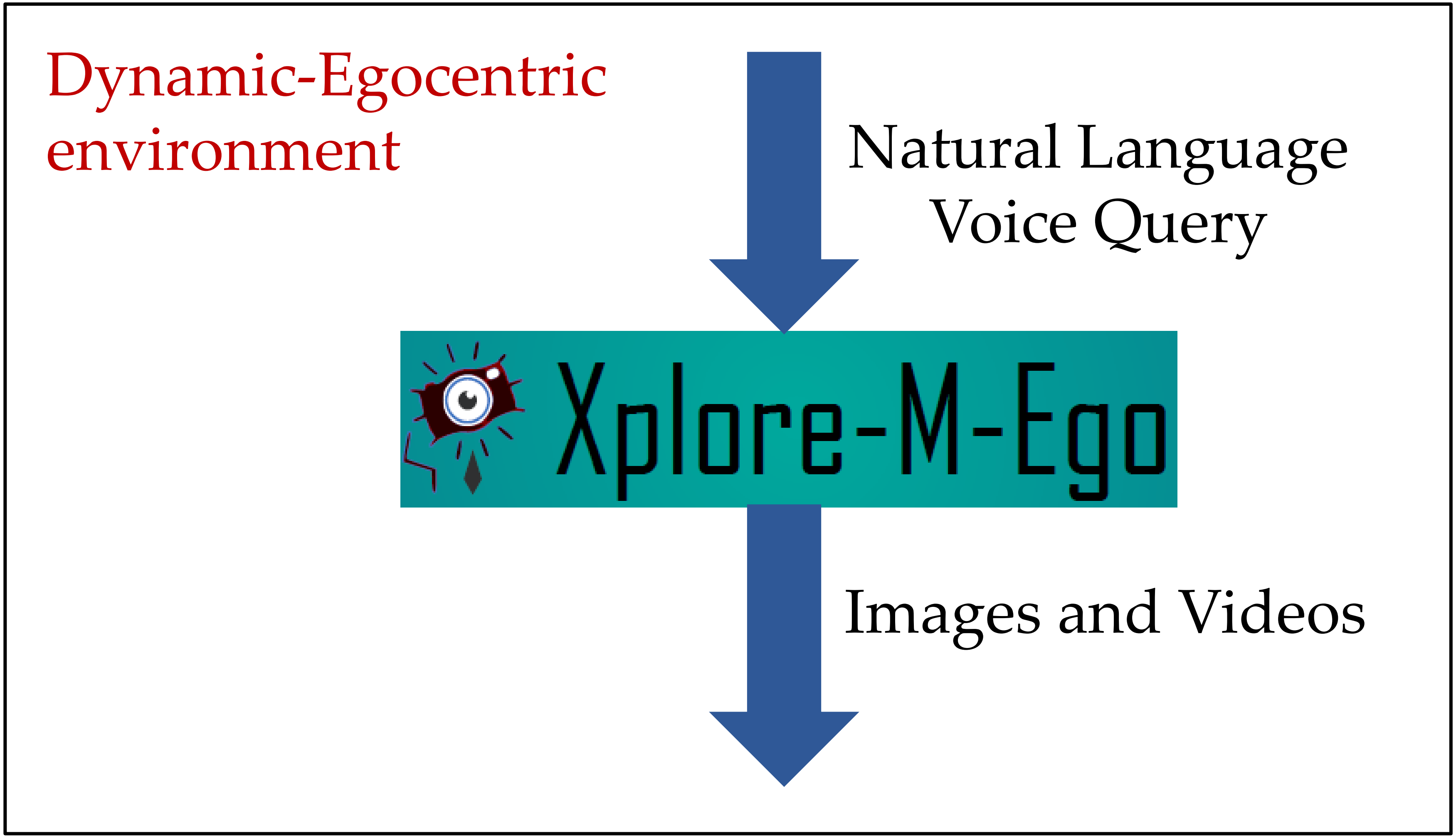
System Overview

Demonstration : Spatial Exploration

System Overview

Demonstration : Temporal Exploration

System Overview



Related Work

Category	Existing Functions	Our contribution

Related Work

Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval		

Related Work

Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval	Browsing media collections in a static allocentric setting; Click-based GUI	

Related Work

Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval	Browsing media collections in a static allocentric setting; Click-based GUI	Browsing media collections in a dynamic egocentric setting; hands-free GUI

Related Work

Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval	Browsing media collections in a static allocentric setting; Click-based GUI	Browsing media collections in a dynamic egocentric setting; hands-free GUI
Natural Language Query Processing		

Related Work

Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval	Browsing media collections in a static allocentric setting; Click-based GUI	Browsing media collections in a dynamic egocentric setting; hands-free GUI
Natural Language Query Processing	Question-answering w.r.t. a static world; Returning textual information	

Related Work

Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval	Browsing media collections in a static allocentric setting; Click-based GUI	Browsing media collections in a dynamic egocentric setting; hands-free GUI
Natural Language Query Processing	Question-answering w.r.t. a static world; Returning textual information	Question-answering w.r.t. a dynamic world; Returning media files

Related Work

Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval	Browsing media collections in a static allocentric setting; Click-based GUI	Browsing media collections in a dynamic egocentric setting; hands-free GUI
Natural Language Query Processing	Question-answering w.r.t. a static world; Returning textual information	Question-answering w.r.t. a dynamic world; Returning media files
Media Retrieval Using Natural Language Queries		

Related Work

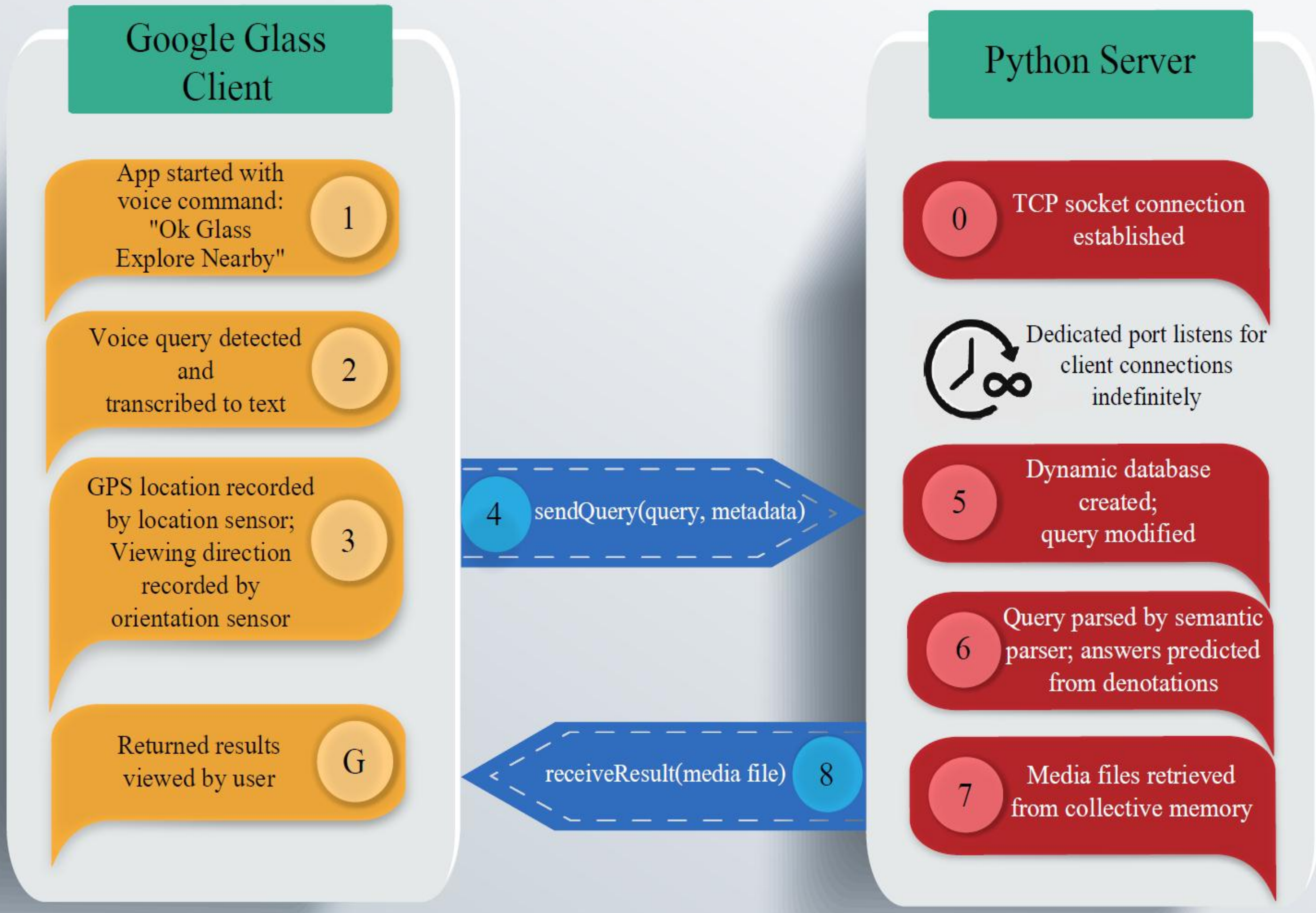
Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval	Browsing media collections in a static allocentric setting; Click-based GUI	Browsing media collections in a dynamic egocentric setting; hands-free GUI
Natural Language Query Processing	Question-answering w.r.t. a static world; Returning textual information	Question-answering w.r.t. a dynamic world; Returning media files
Media Retrieval Using Natural Language Queries	Retrieving media based on scene contents; Using short structured phrases as queries; Does not take into account user's context	

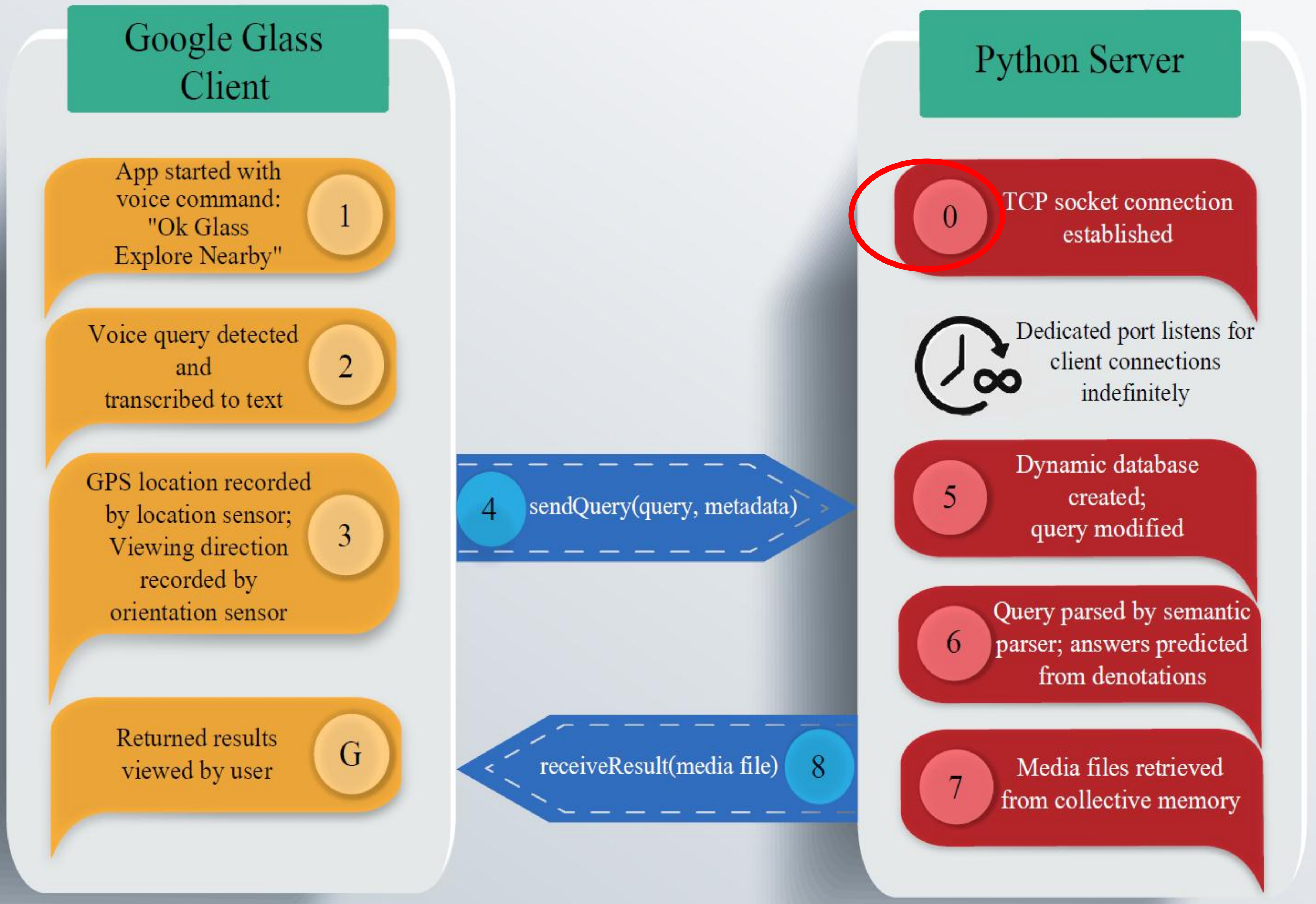
Related Work

Category	Existing Functions	Our contribution
Spatio-temporal Media Retrieval	Browsing media collections in a static allocentric setting; Click-based GUI	Browsing media collections in a dynamic egocentric setting; hands-free GUI
Natural Language Query Processing	Question-answering w.r.t. a static world; Returning textual information	Question-answering w.r.t. a dynamic world; Returning media files
Media Retrieval Using Natural Language Queries	Retrieving media based on scene contents; Using short structured phrases as queries; Does not take into account user's context	Retrieving media based on geographic location; Using rich complete natural language sentences as queries; Takes into account user's context

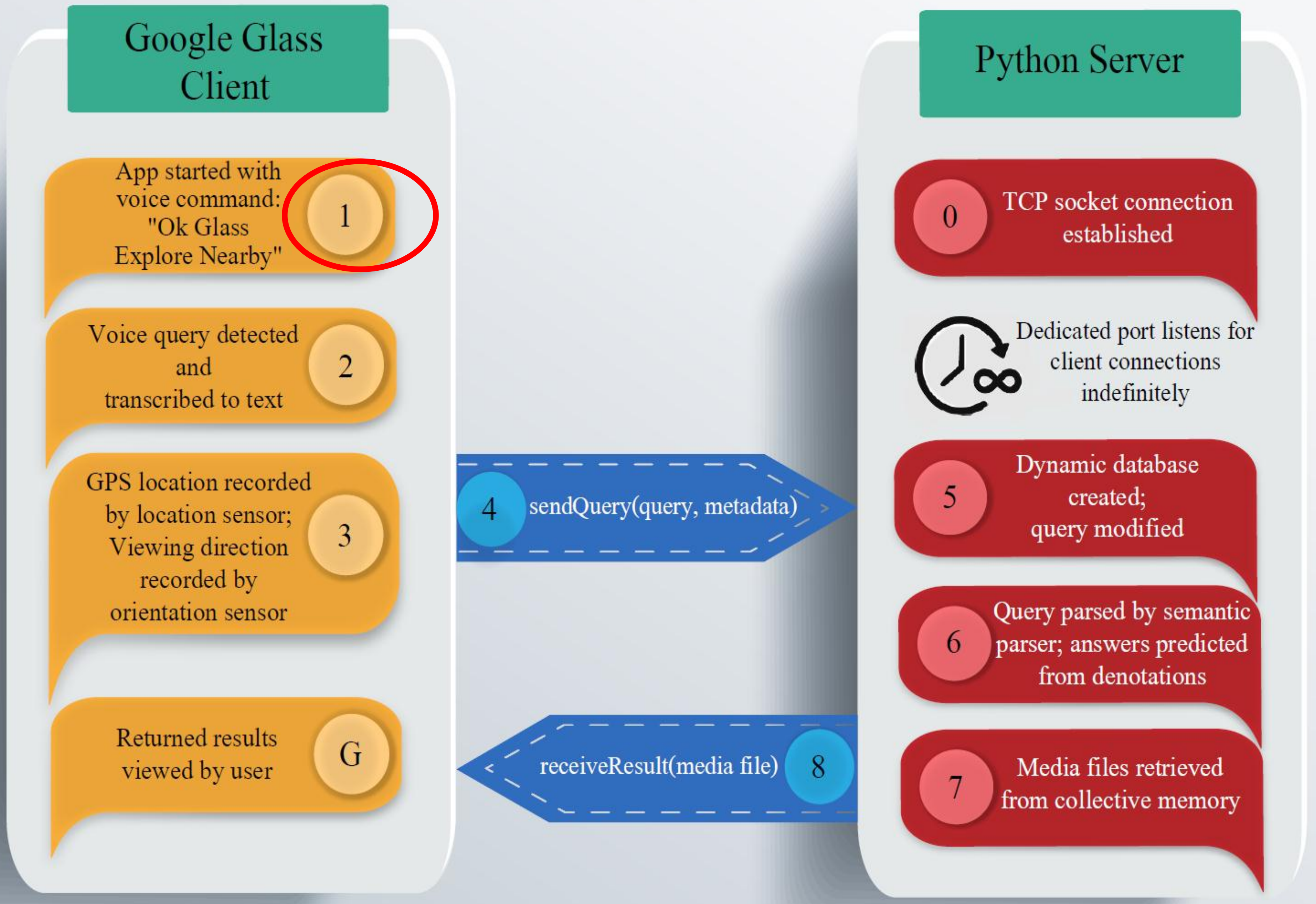
Outline

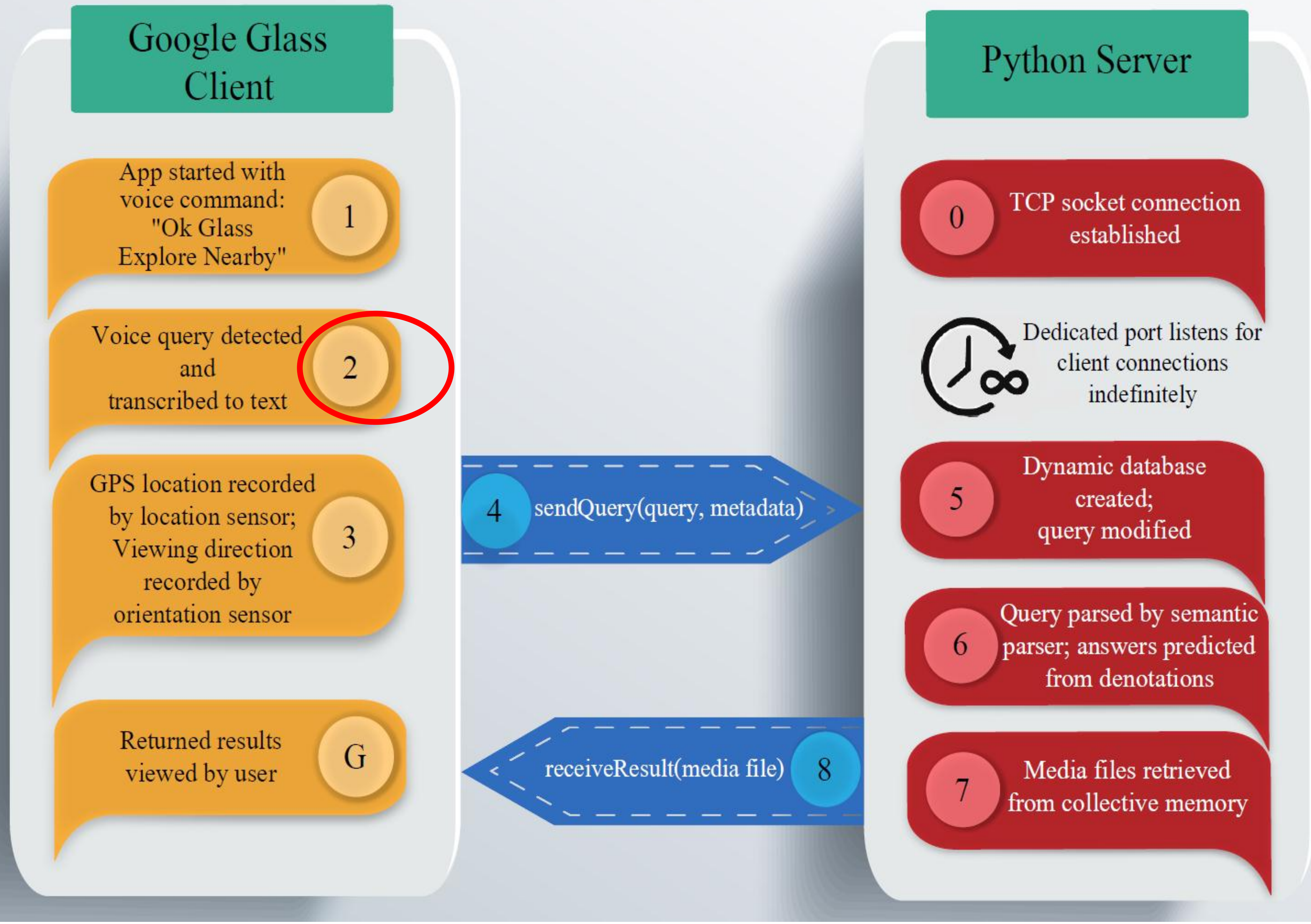
- Motivation and Overview
- Contextual Media Retrieval System
- Results and Conclusion

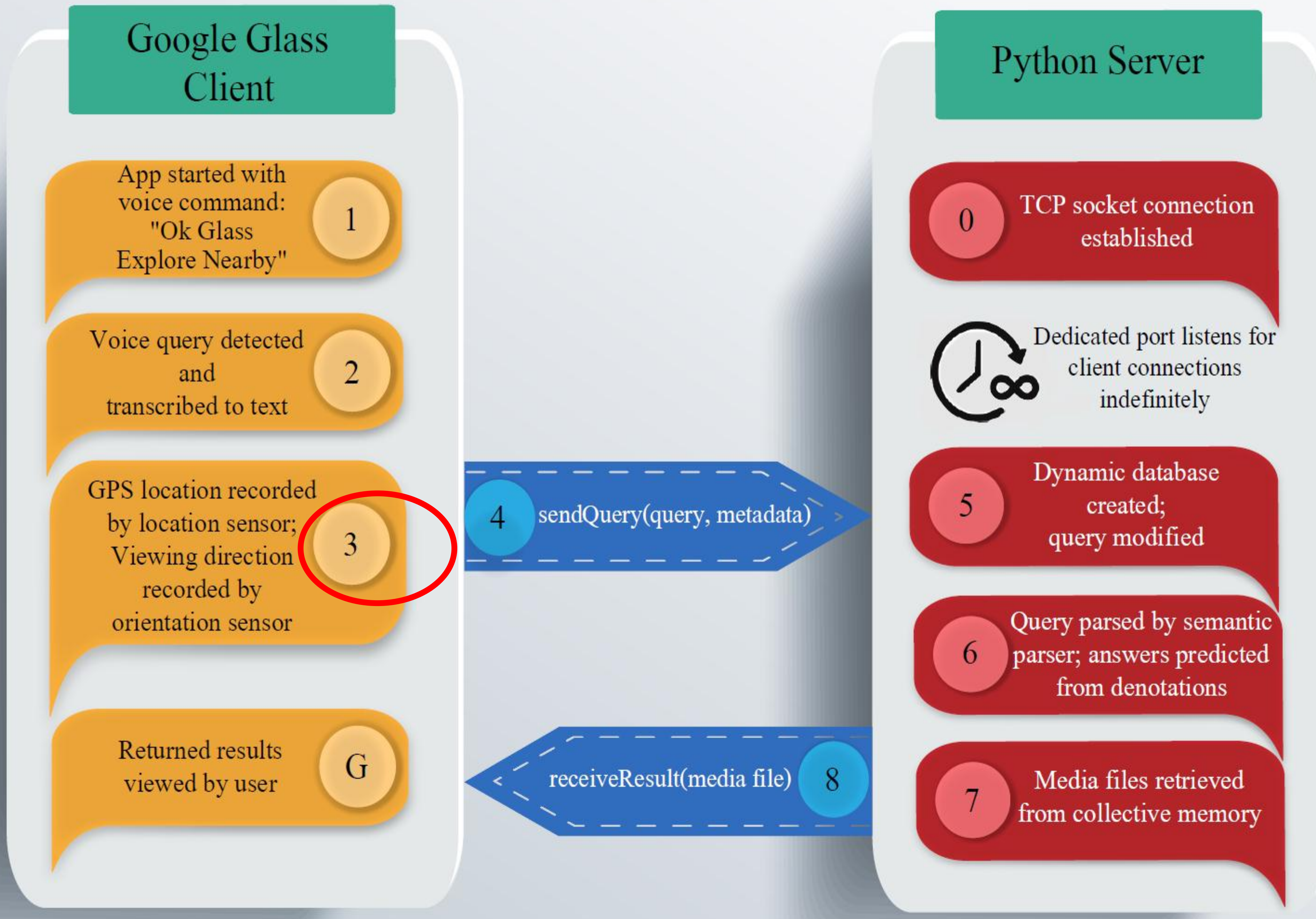

Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory

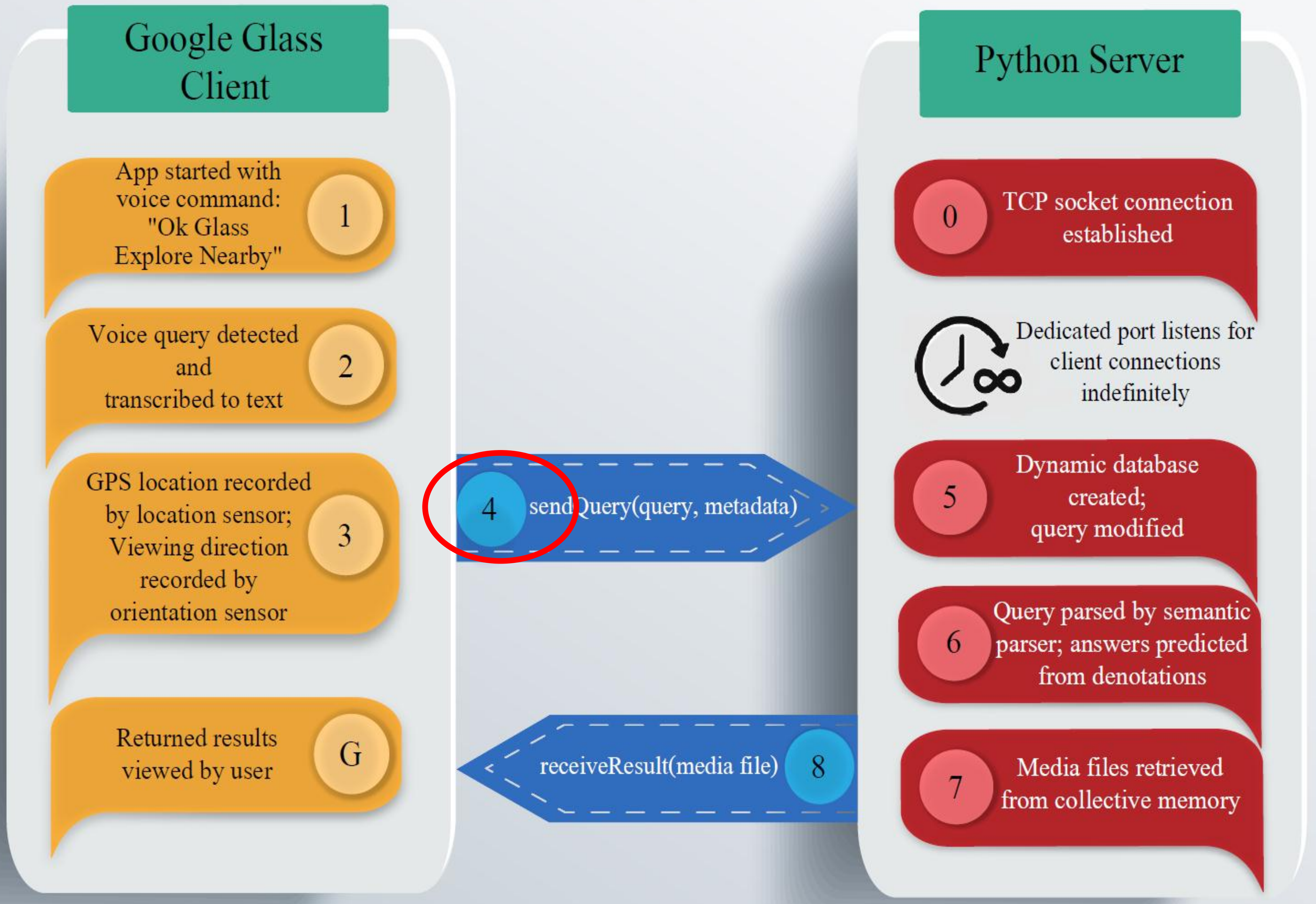


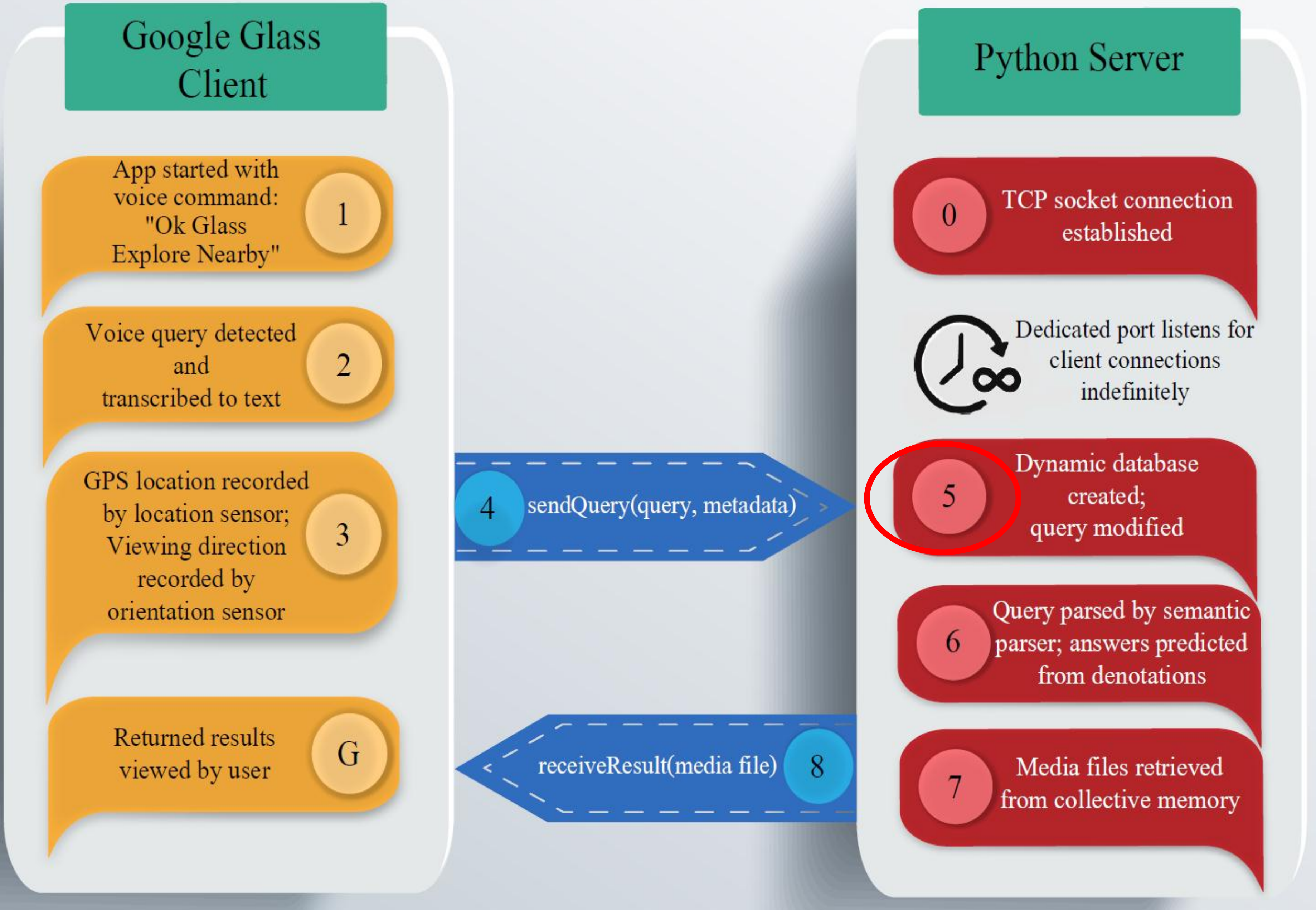
Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory

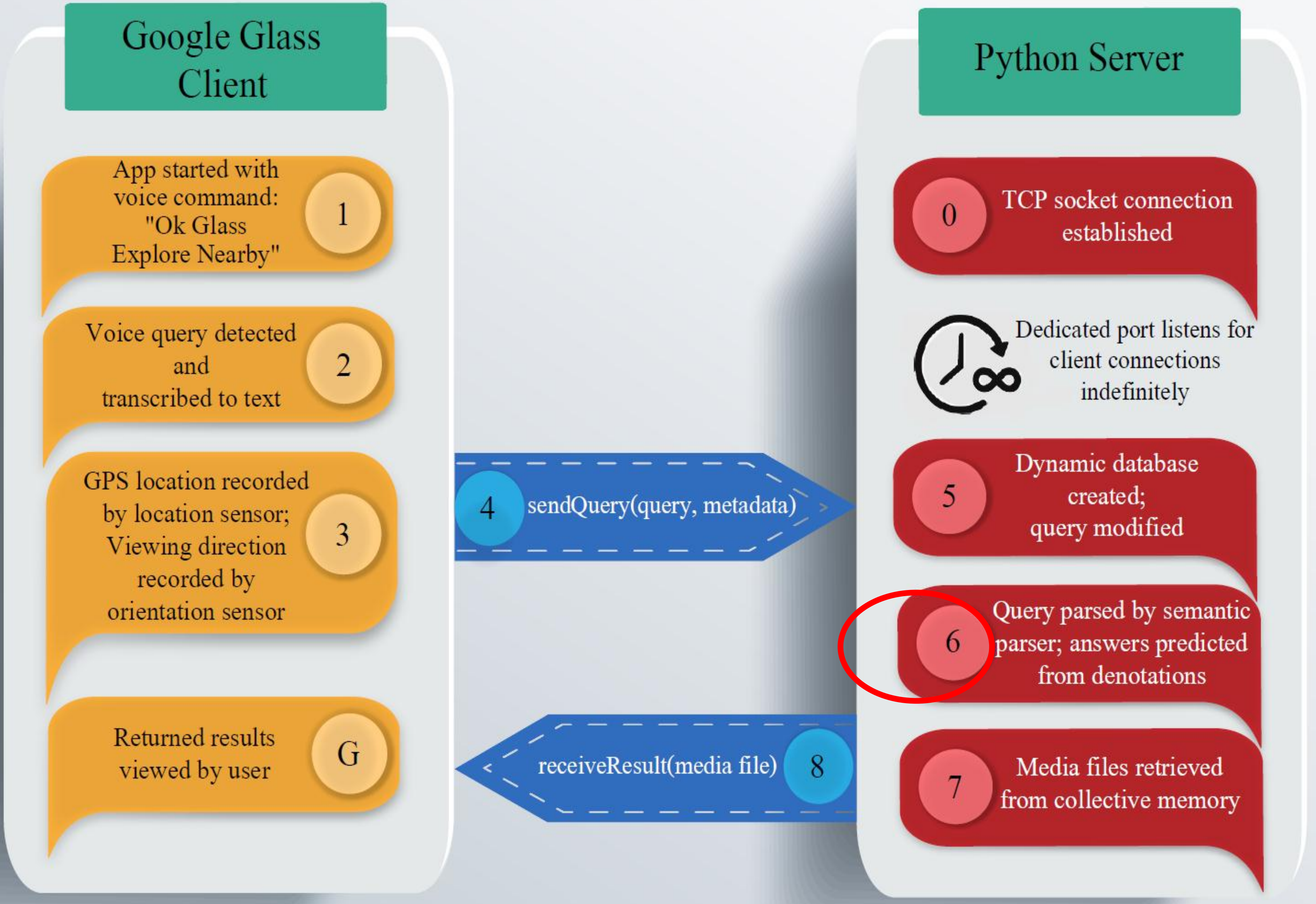

Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory

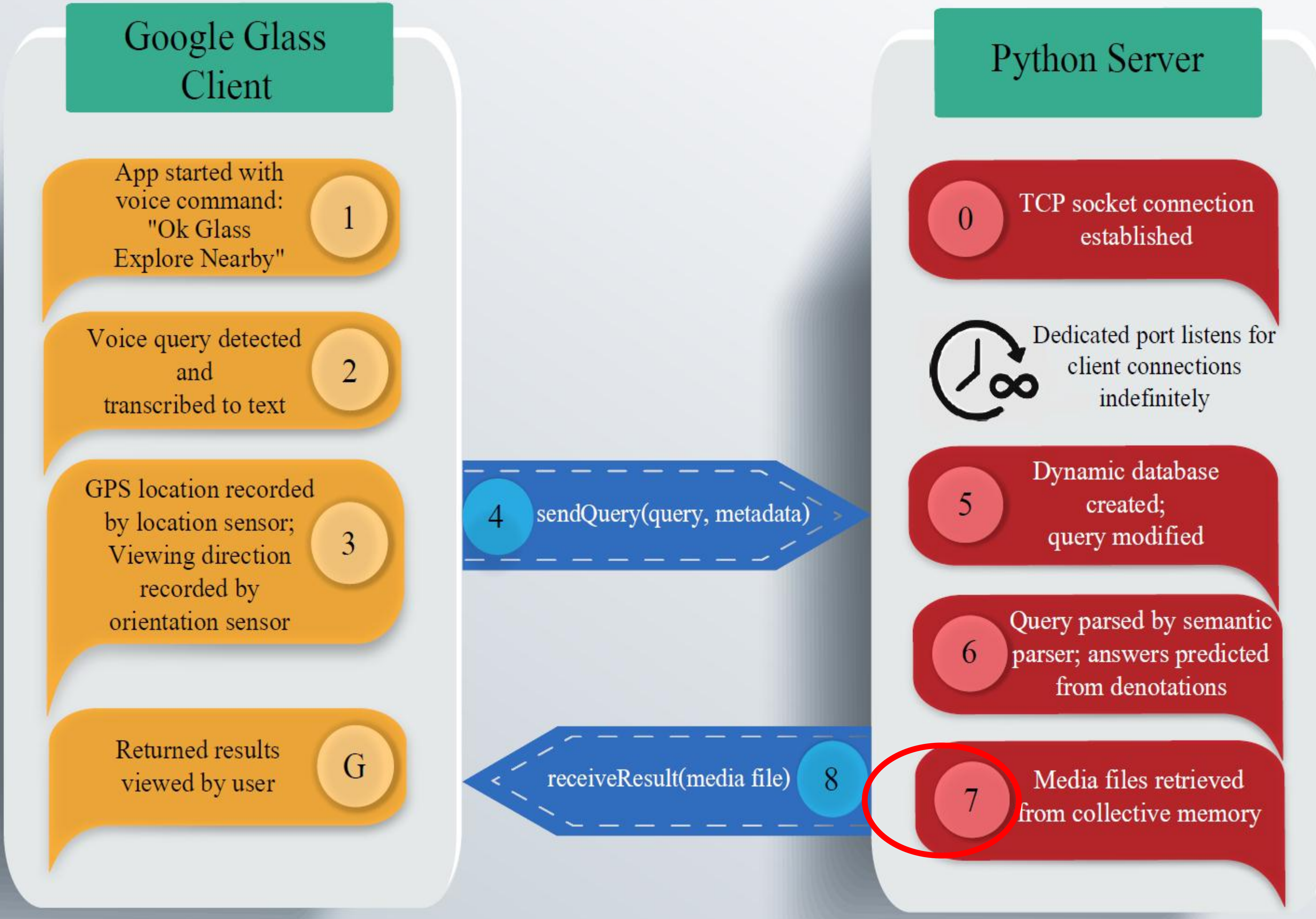

Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory

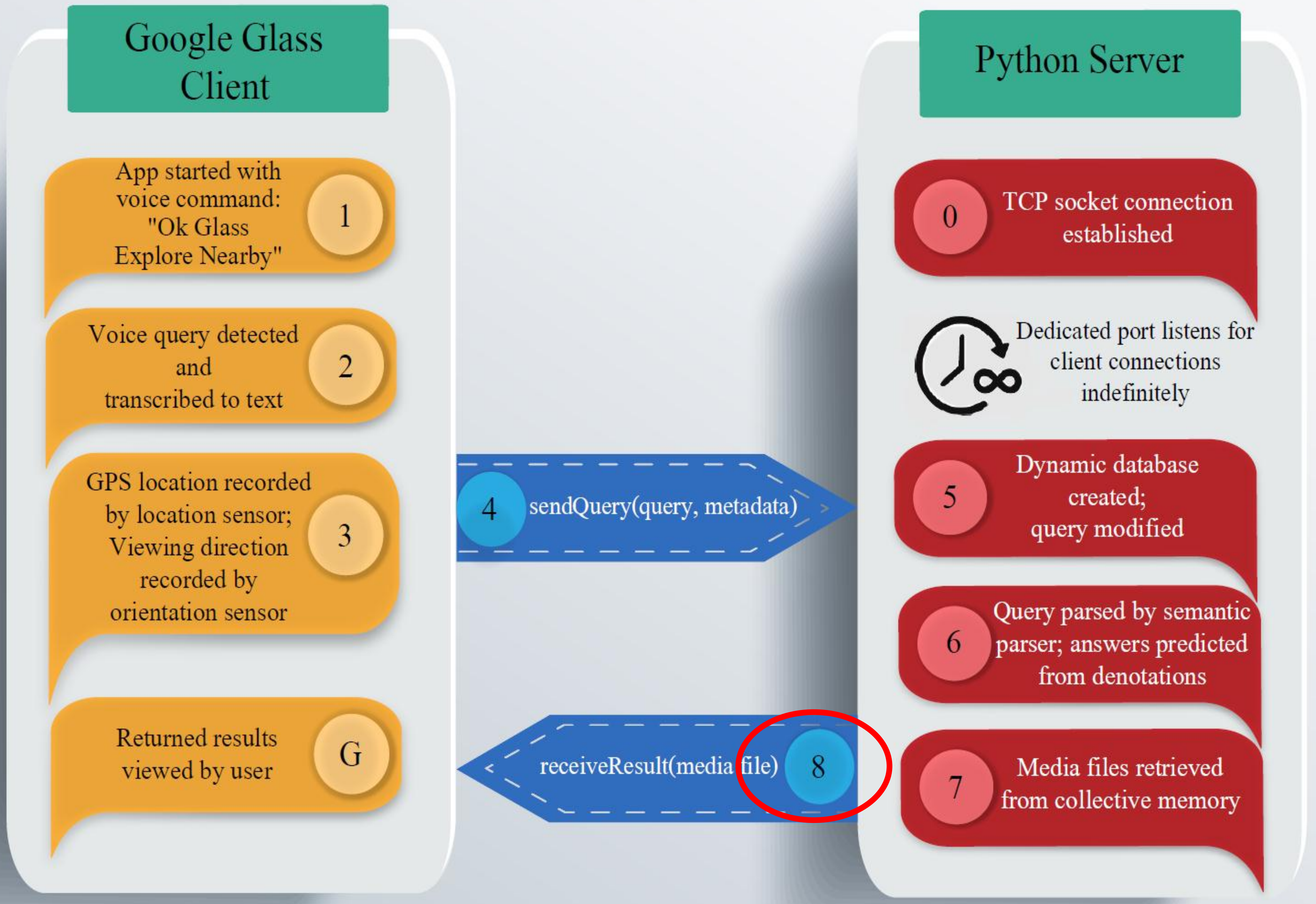

Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory

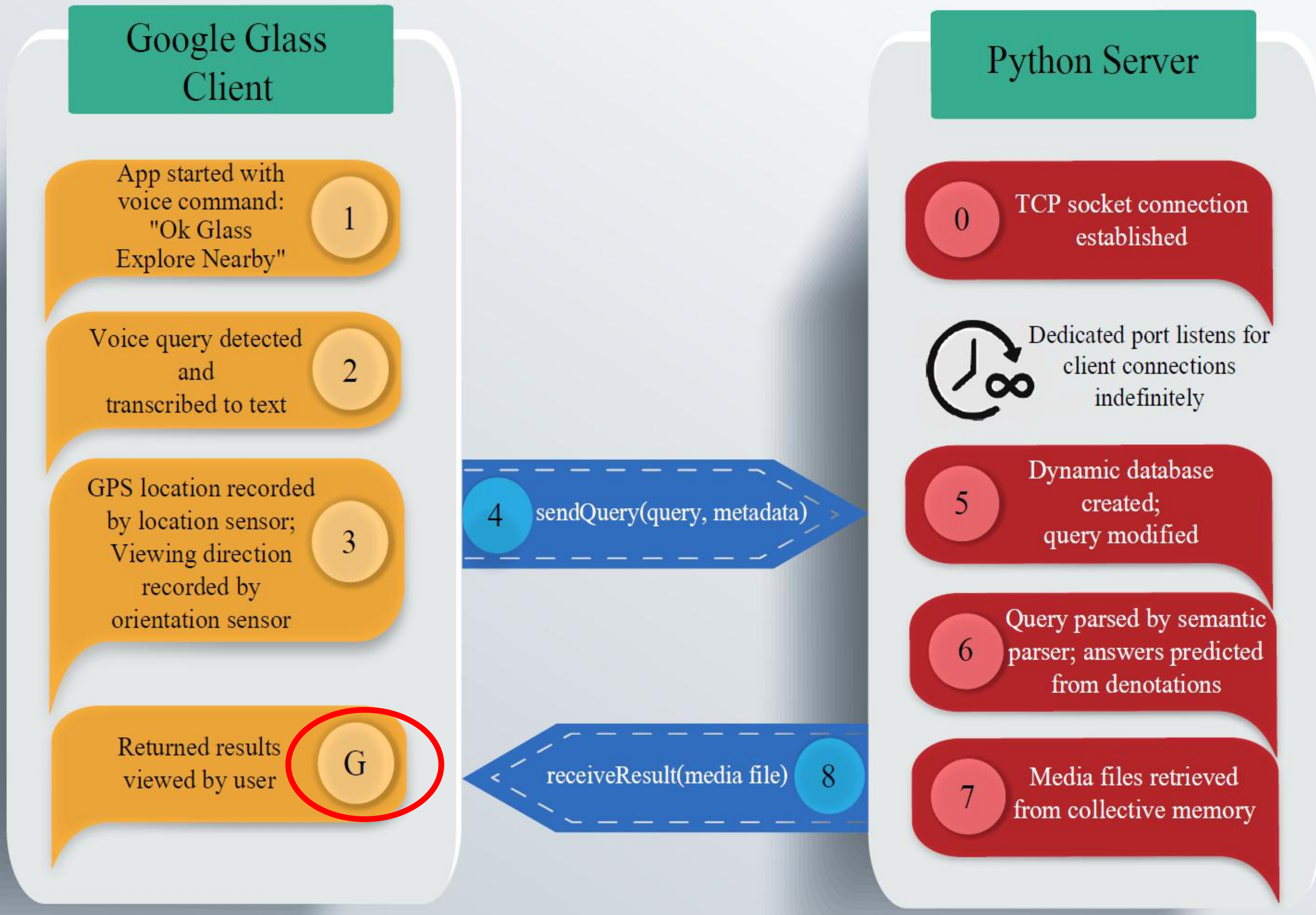

Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory


Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory

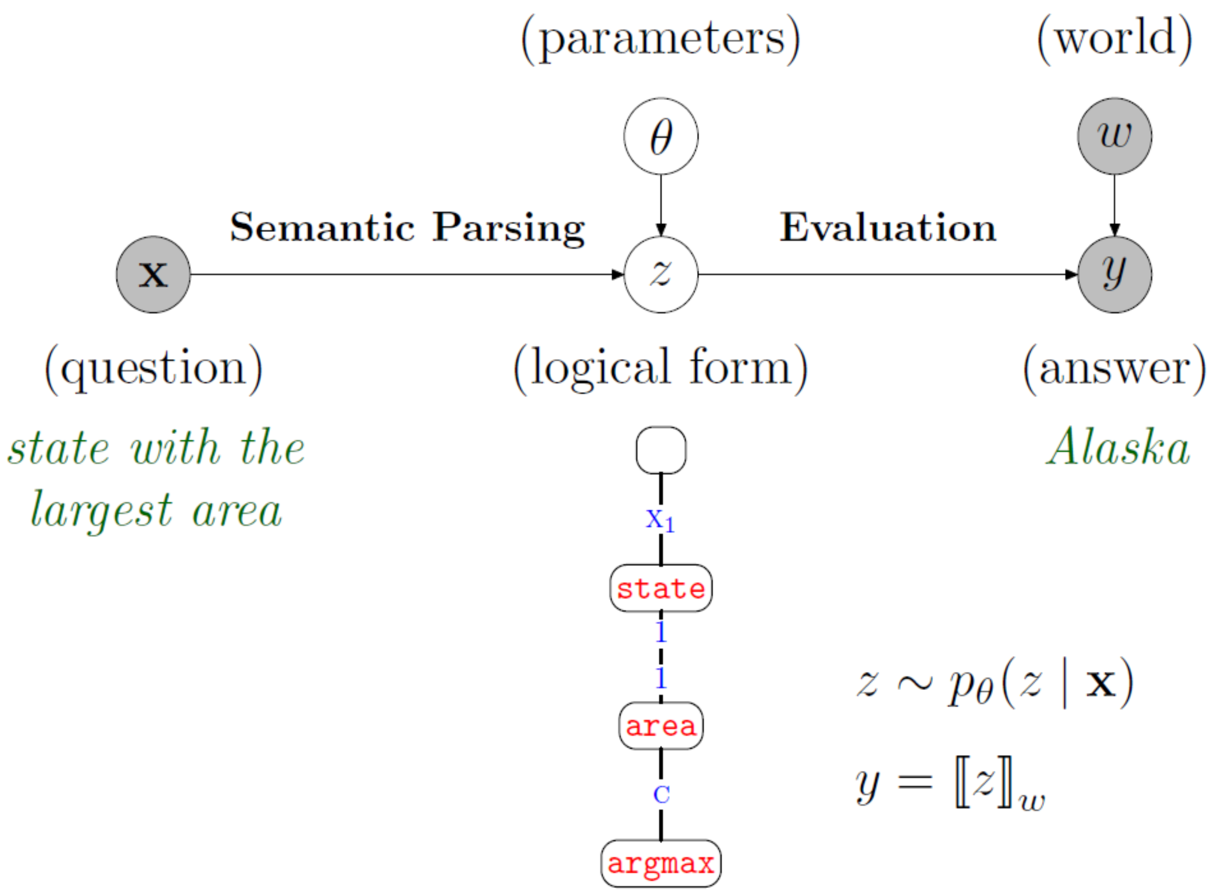

Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory


Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory


Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory


Google Glass Client
App started with voice command: "Ok Glass Explore Nearby"
1
Voice query detected and transcribed to text
2
GPS location recorded by location sensor; Viewing direction recorded by orientation sensor
3
Returned results viewed by user
G
4
sendQuery(query, metadata)
receiveResult(media file)
8
Python Server
0
TCP socket connection established
Dedicated port listens for client connections indefinitely
5
Dynamic database created; query modified
6
Query parsed by semantic parser; answers predicted from denotations
7
Media files retrieved from collective memory

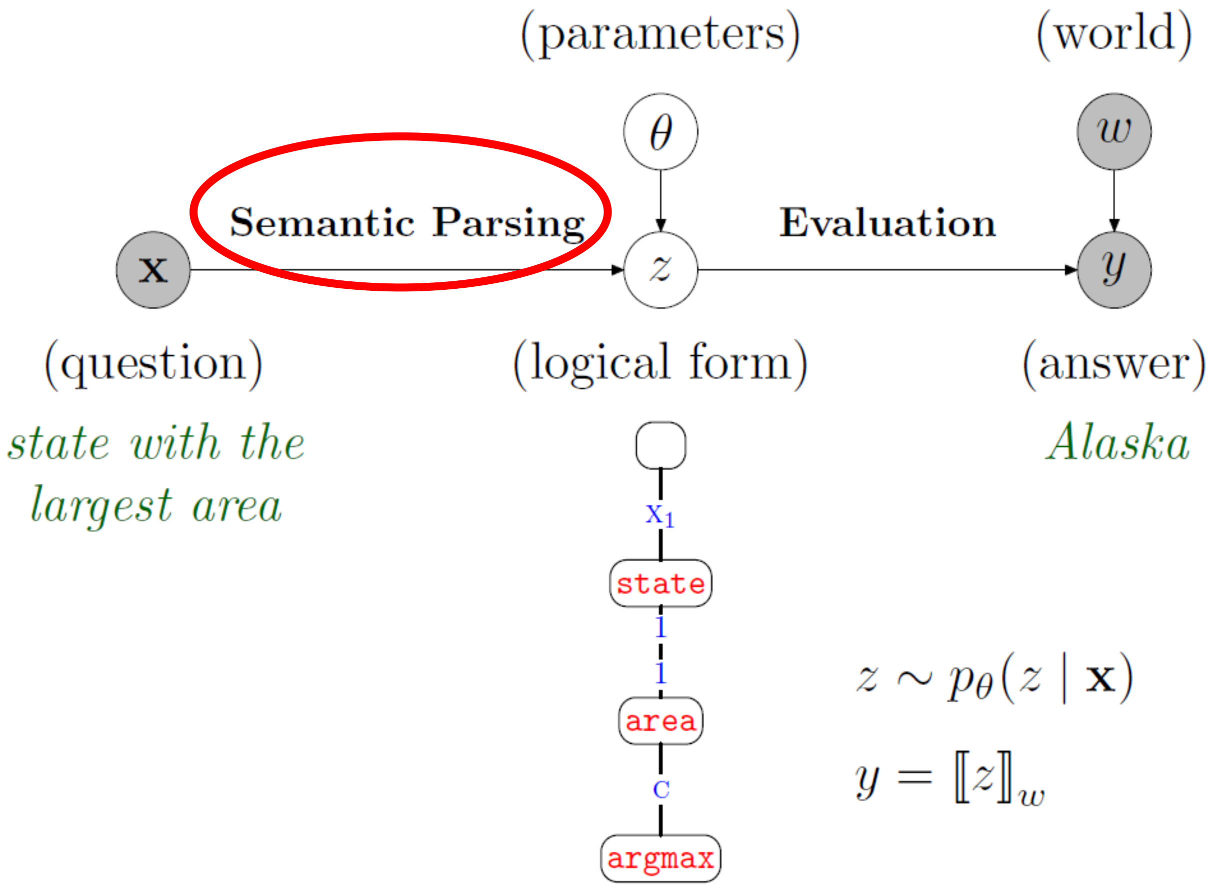
Question – Answering



Q&A Model of Percy Liang

("Learning Dependency-based Compositional Semantics", Liang et al.)

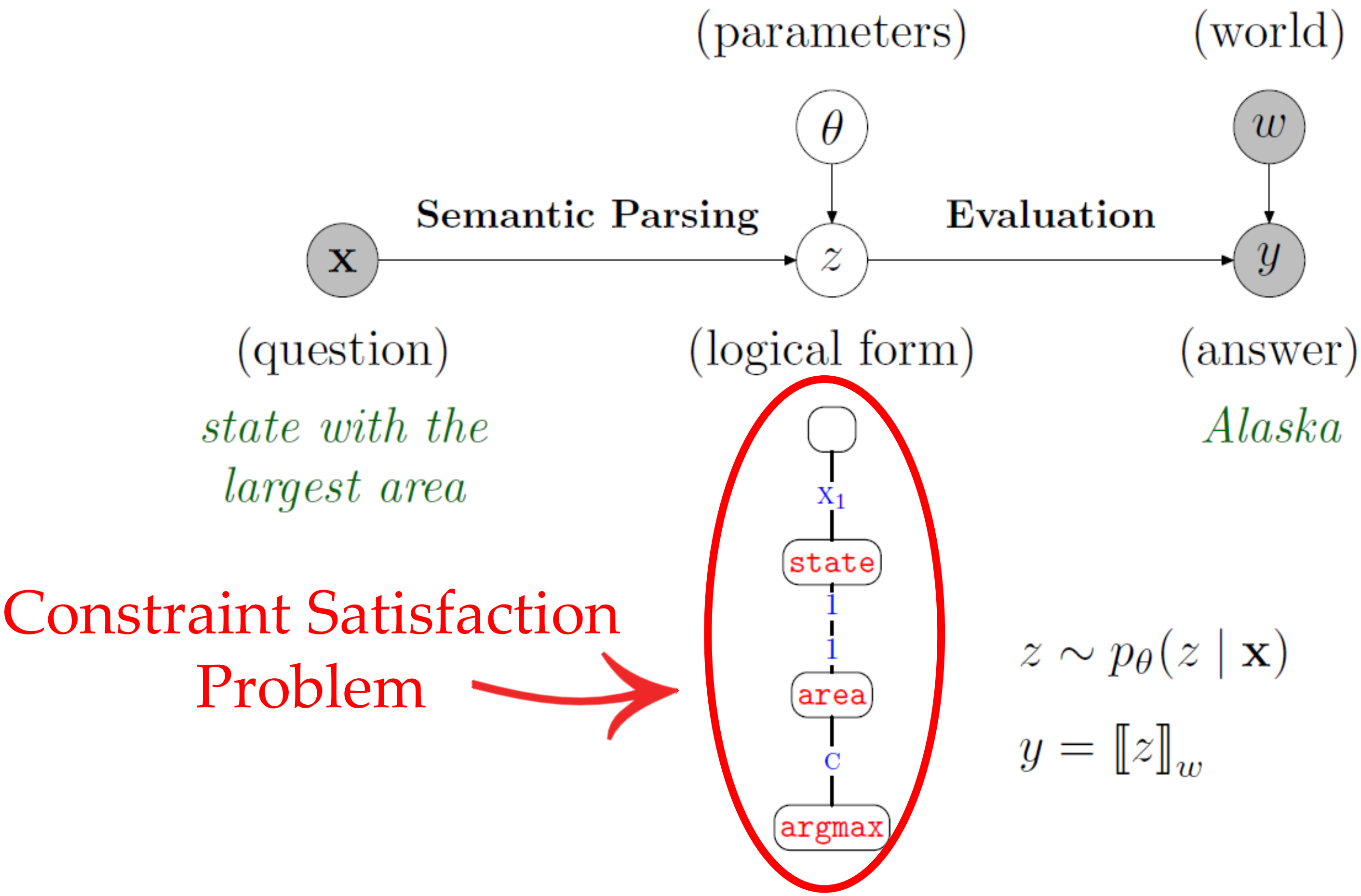
Question – Answering



Q&A Model of Percy Liang

("Learning Dependency-based Compositional Semantics", Liang et al.)

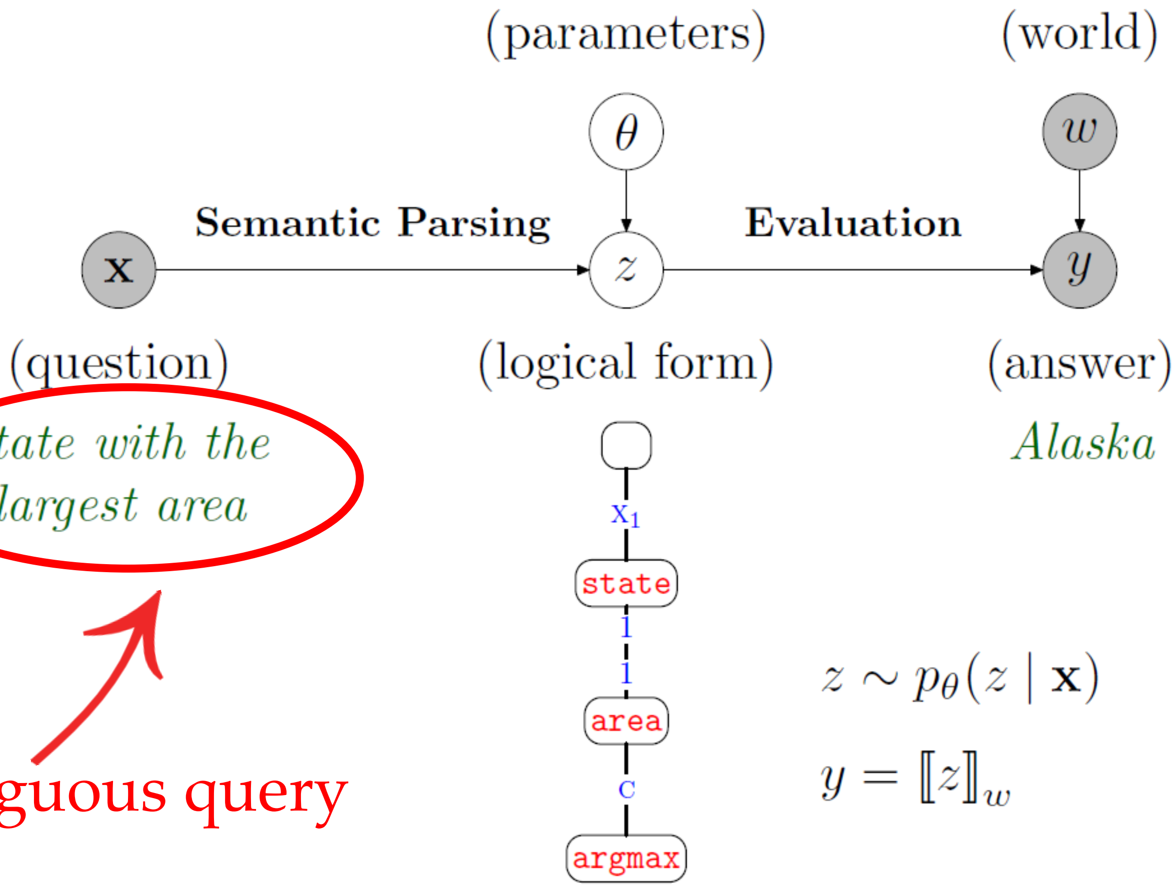
Question – Answering



Q&A Model of Percy Liang

("Learning Dependency-based Compositional Semantics", Liang et al.)

Question – Answering



$$z \sim p_\theta(z \mid \mathbf{x})$$

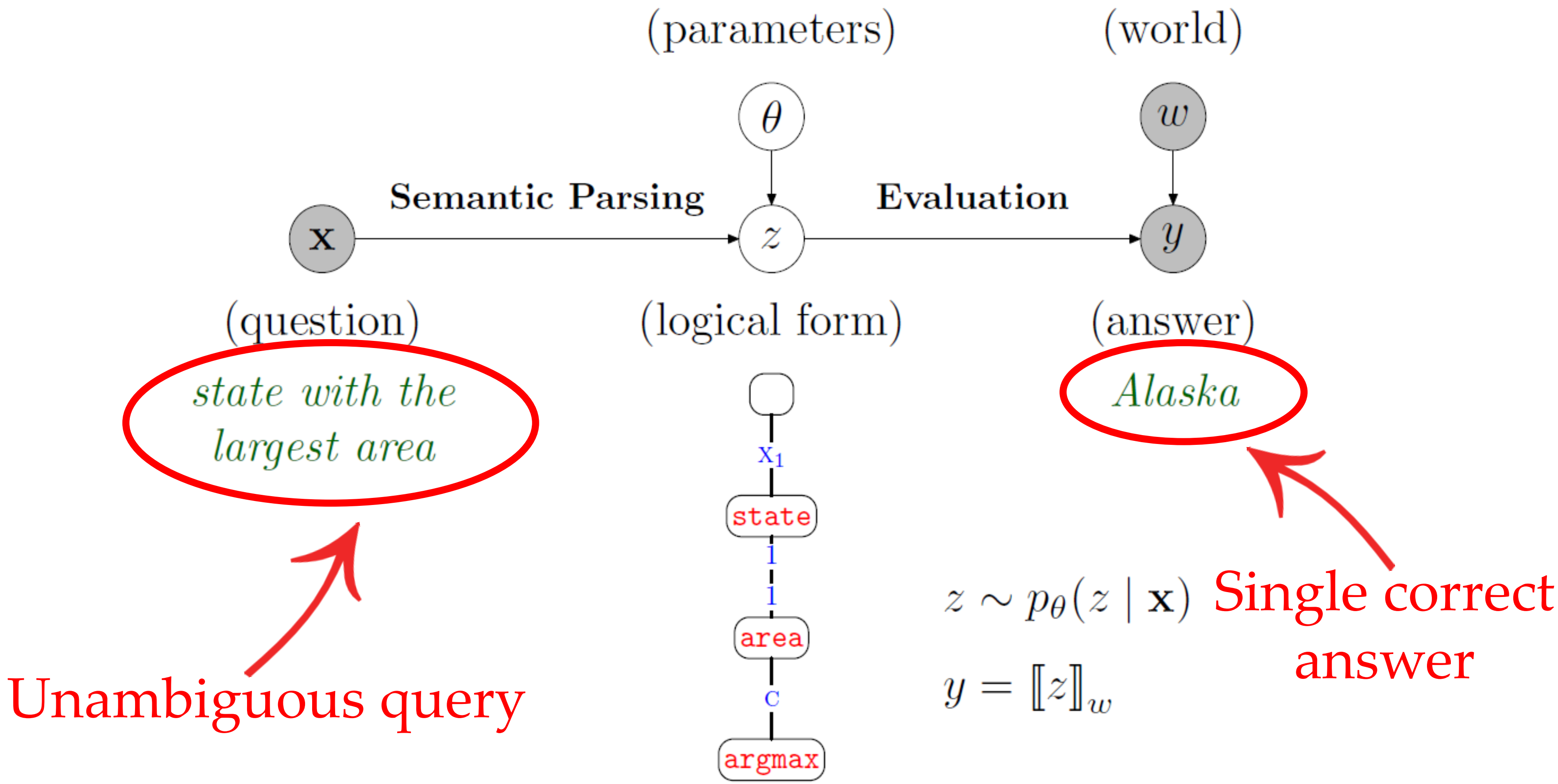
$$y = [\![z]\!]_w$$

Unambiguous query

Q&A Model of Percy Liang

(“Learning Dependency-based Compositional Semantics”, Liang et al.)

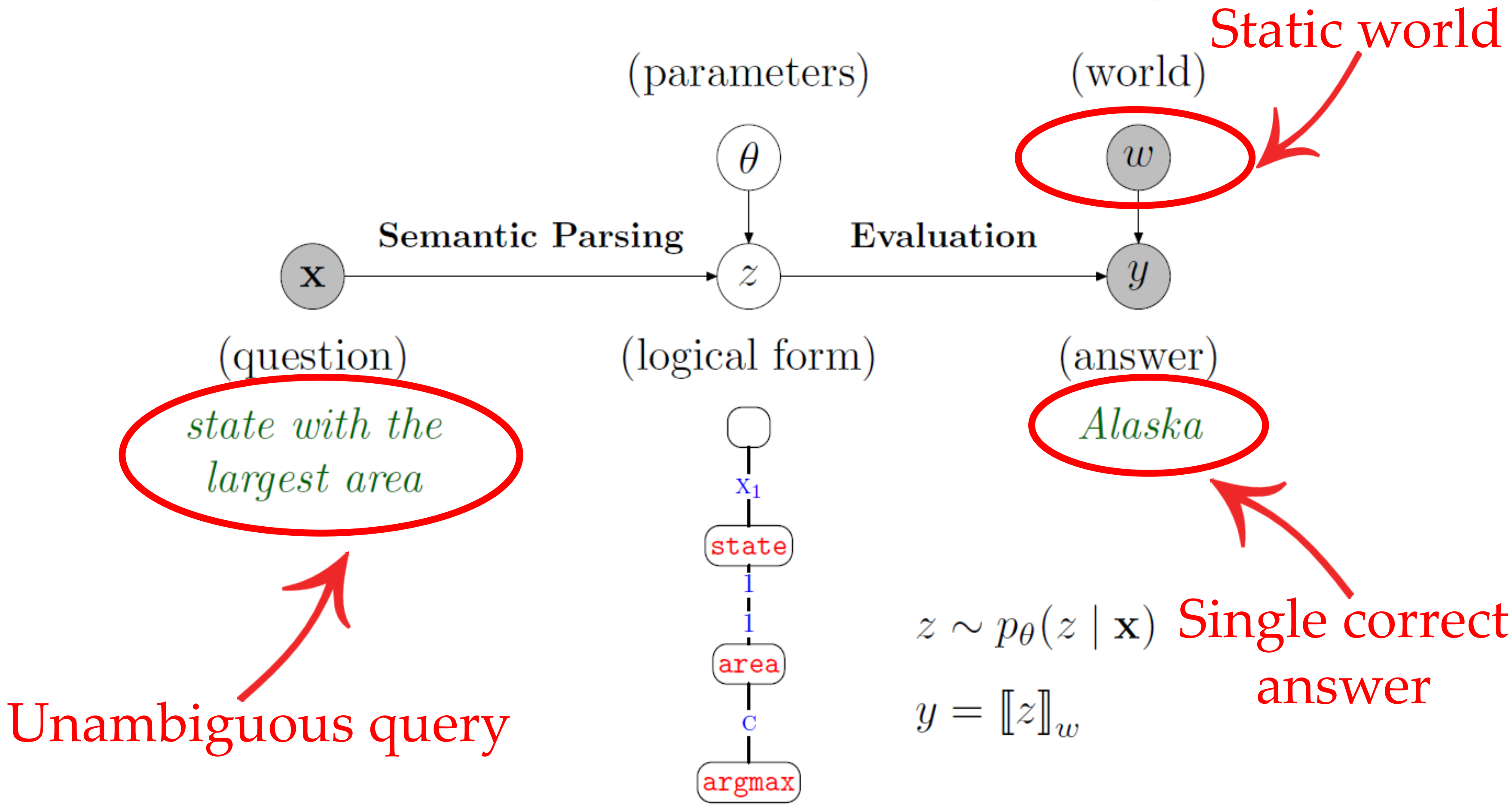
Question – Answering



Q&A Model of Percy Liang

("Learning Dependency-based Compositional Semantics", Liang et al.)

Question – Answering



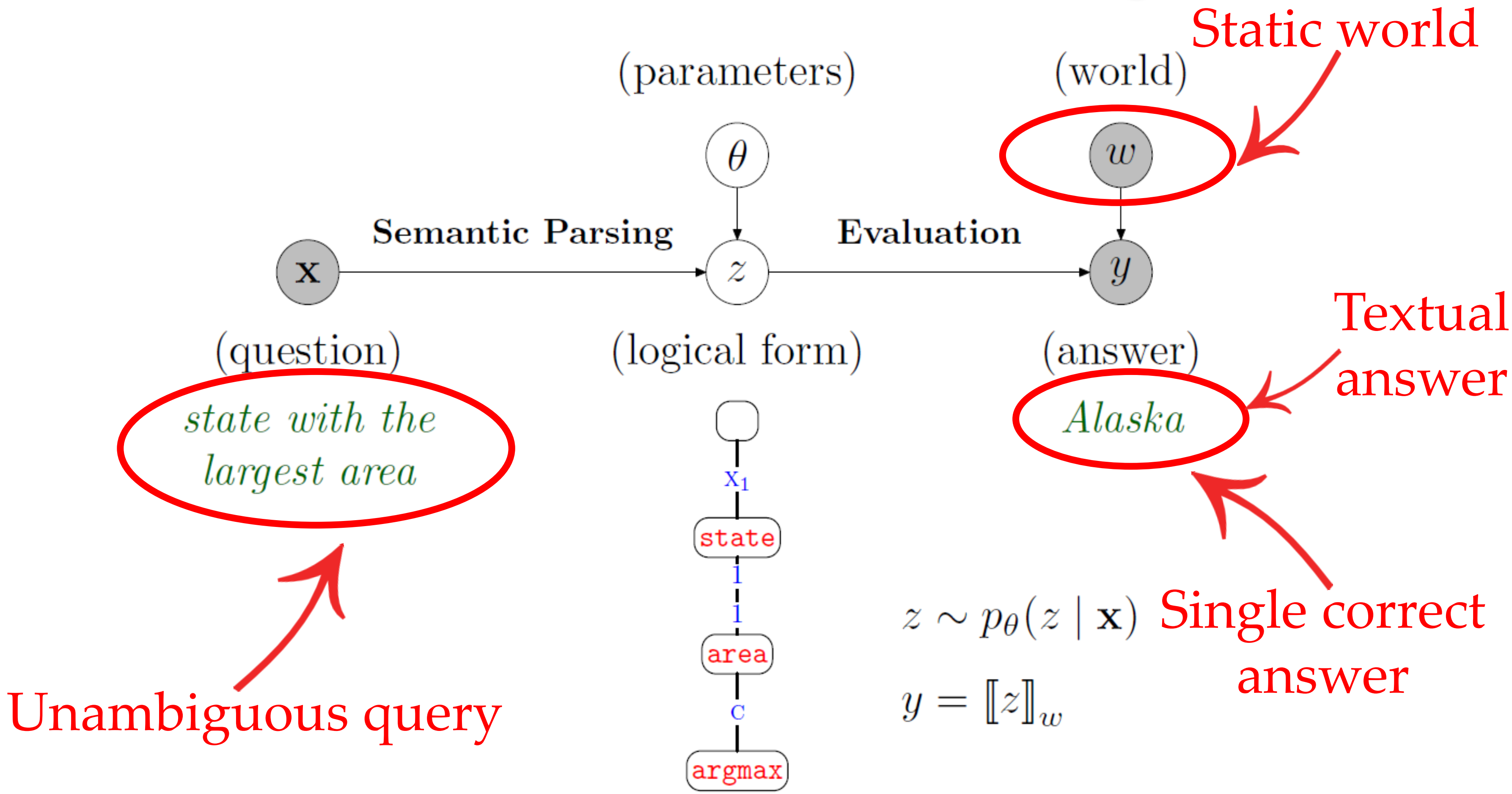
$$z \sim p_\theta(z \mid \mathbf{x})$$

$$y = [\![z]\!]_w$$

Q&A Model of Percy Liang

("Learning Dependency-based Compositional Semantics", Liang et al.)

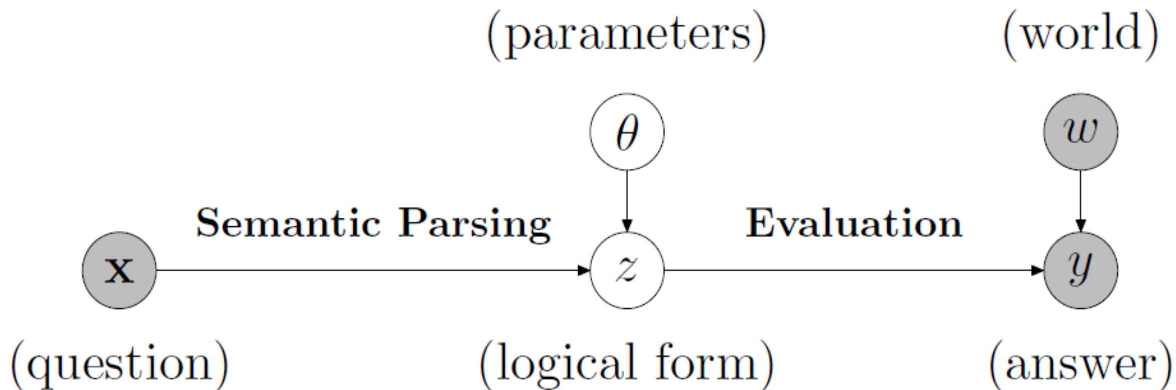
Question – Answering



Q&A Model of Percy Liang

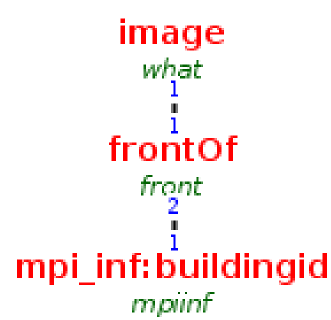
("Learning Dependency-based Compositional Semantics", Liang et al.)

Question – Answering



what is there in front of MPI-INF?

Ambiguous query

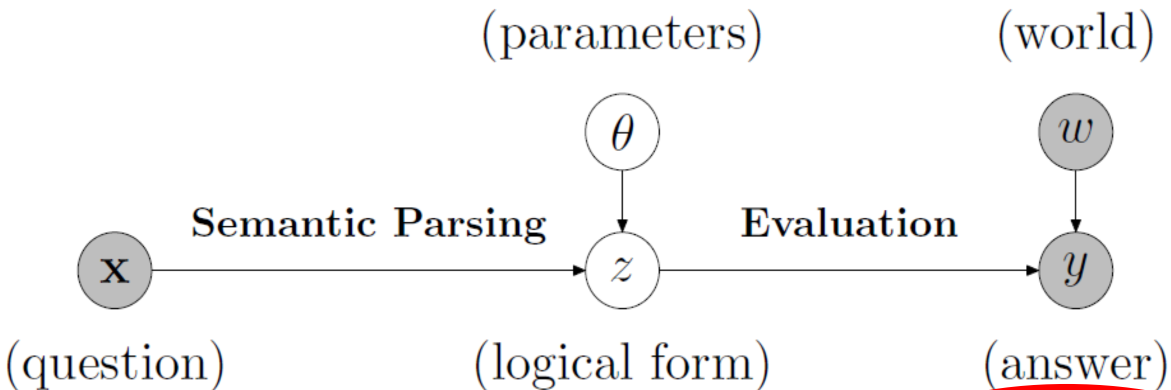


$$z \sim p_\theta(z \mid \mathbf{x})$$

$$y = [\![z]\!]_w$$

Our Q&A Model

Question – Answering



what is there in front of MPI-INF?

image
what
1
1
frontOf
front
2
1
mpi_inf:buildingid
mpiinf

$$z \sim p_\theta(z \mid \mathbf{x})$$

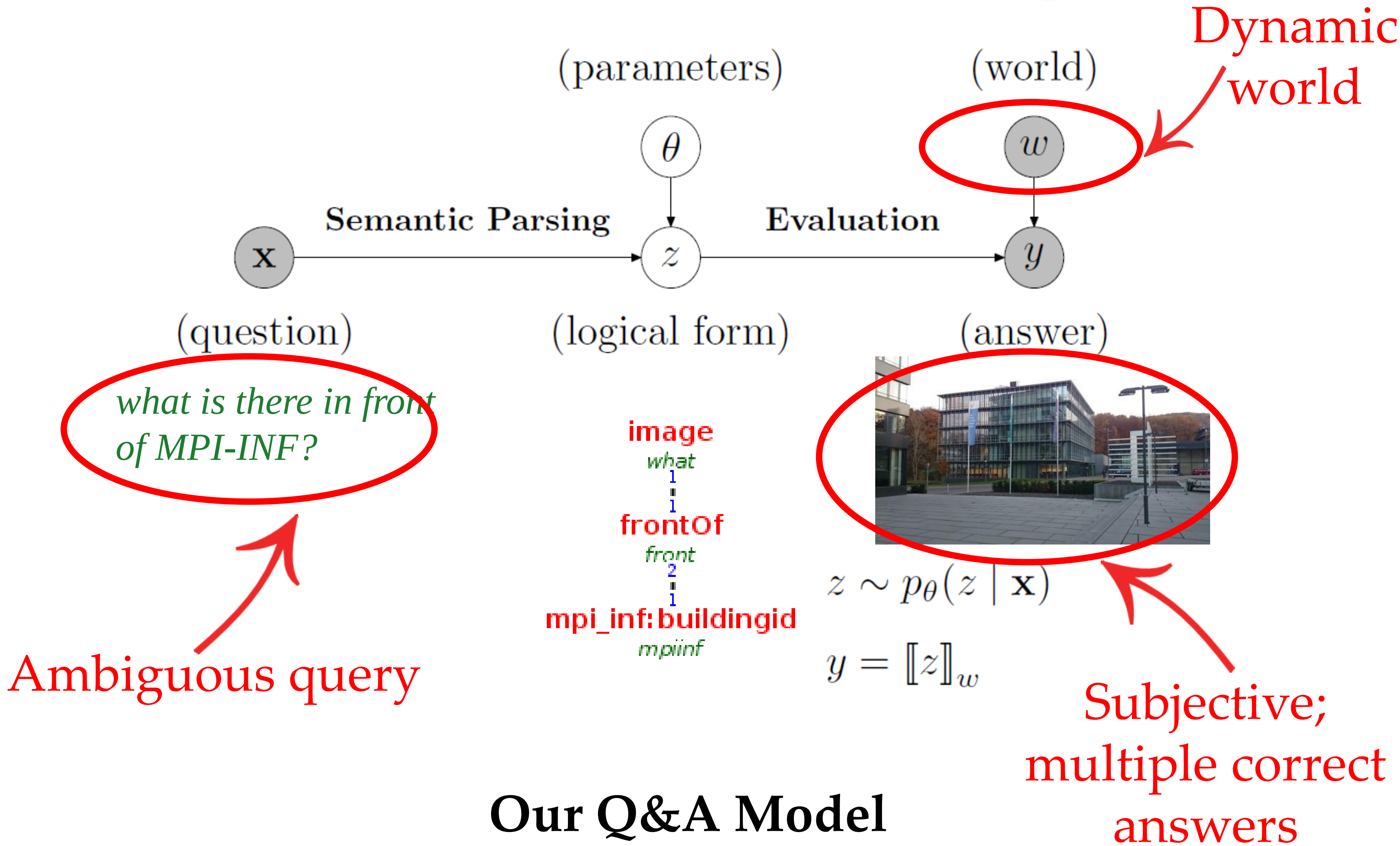
$$y = [\![z]\!]_w$$

Ambiguous query

Subjective; multiple correct answers

Our Q&A Model

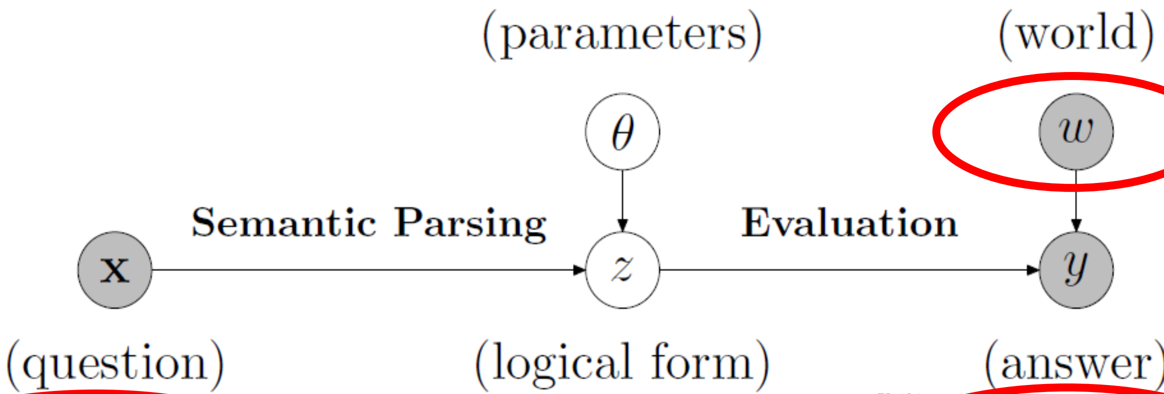
Question – Answering



Our Q&A Model

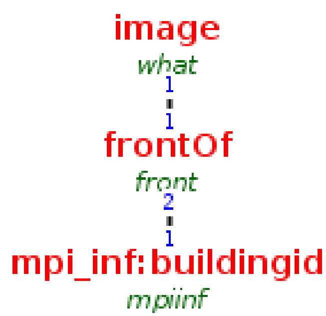
Question – Answering

Dynamic world



Media as answer

what is there in front of MPI-INF?



$$z \sim p_\theta(z \mid \mathbf{x})$$

$$y = [\![z]\!]_w$$

Ambiguous query

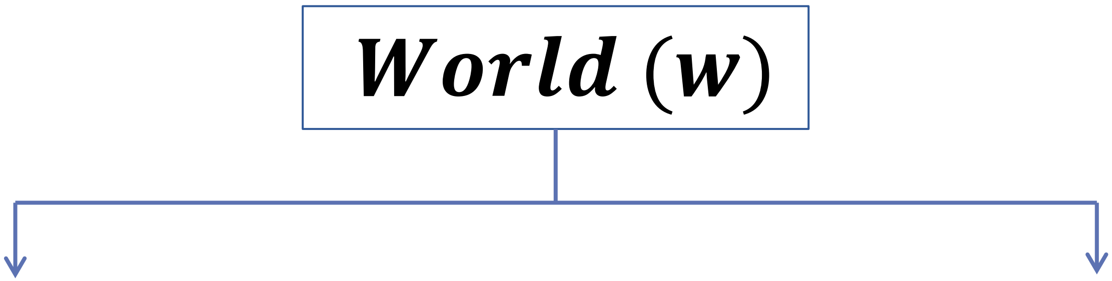
Subjective; multiple correct answers

Our Q&A Model

Dynamic-Egocentric Extension

$$\boxed{\boldsymbol{World\ (w)}}$$

Dynamic-Egocentric Extension



Dynamic-Egocentric Extension

$\boldsymbol{World\ (w)}$

$\boldsymbol{Static\ World\ (w_s)}$

cafe('mensa',49.2560,7.0454).
building('mpi_inf',49.2578,7.0460).

Dynamic-Egocentric Extension

$\boldsymbol{World\ (w)}$

$\boldsymbol{Static\ World\ (w_s)}$

cafe('mensa',49.2560,7.0454).
building('mpi_inf',49.2578,7.0460).

$\boldsymbol{Dynamic\ World\ (w_d)}$

Dynamic-Egocentric Extension

$\boldsymbol{World\ (w)}$

$\boldsymbol{Static\ World\ (w_s)}$

cafe('mensa',49.2560,7.0454).
building('mpi_inf',49.2578,7.0460).

$\boldsymbol{Dynamic\ World\ (w_d)}$

Dynamic-Egocentric Extension

World (w)

Static World (w_s)

cafe('mensa',49.2560,7.0454).
building('mpi_inf',49.2578,7.0460).

Dynamic World (w_d)

User Metadata (w_{du})

person(49.2578,7.0460,'n').
day(20150220).

Dynamic-Egocentric Extension

World (w)

Static World (w_s)

cafe('mensa',49.2560,7.0454).
building('mpi_inf',49.2578,7.0460).

Dynamic World (w_d)

User Metadata (w_{du})

person(49.2578,7.0460,'n').
day(20150220).

Collective Memory (w_{dm})

image(`img_20141111_165828',20141111,49.2566,7.0442,`november').

video(`vid_20141121_120149',20141121,49.2569, 7.0456,`november').

Qualitative Results

Qualitative Results

"What building is to the left of MPI-SWS?"

Qualitative Results

"What building is to the left of MPI-SWS?"

Qualitative Results

"What building is to the left of MPI-SWS?"

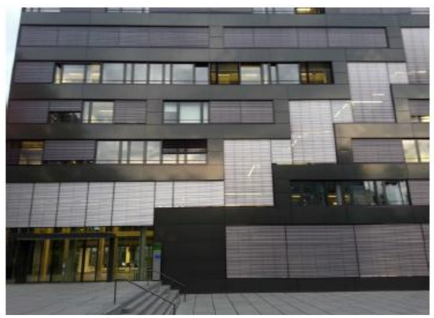
Qualitative Results

"What building is to the left of MPI-SWS?"

Qualitative Results

"What building is to the left of MPI-SWS?"

"What is near MPI-INF?"



Qualitative Results

"What building is to the left of MPI-SWS?"

"What is near MPI-INF?"

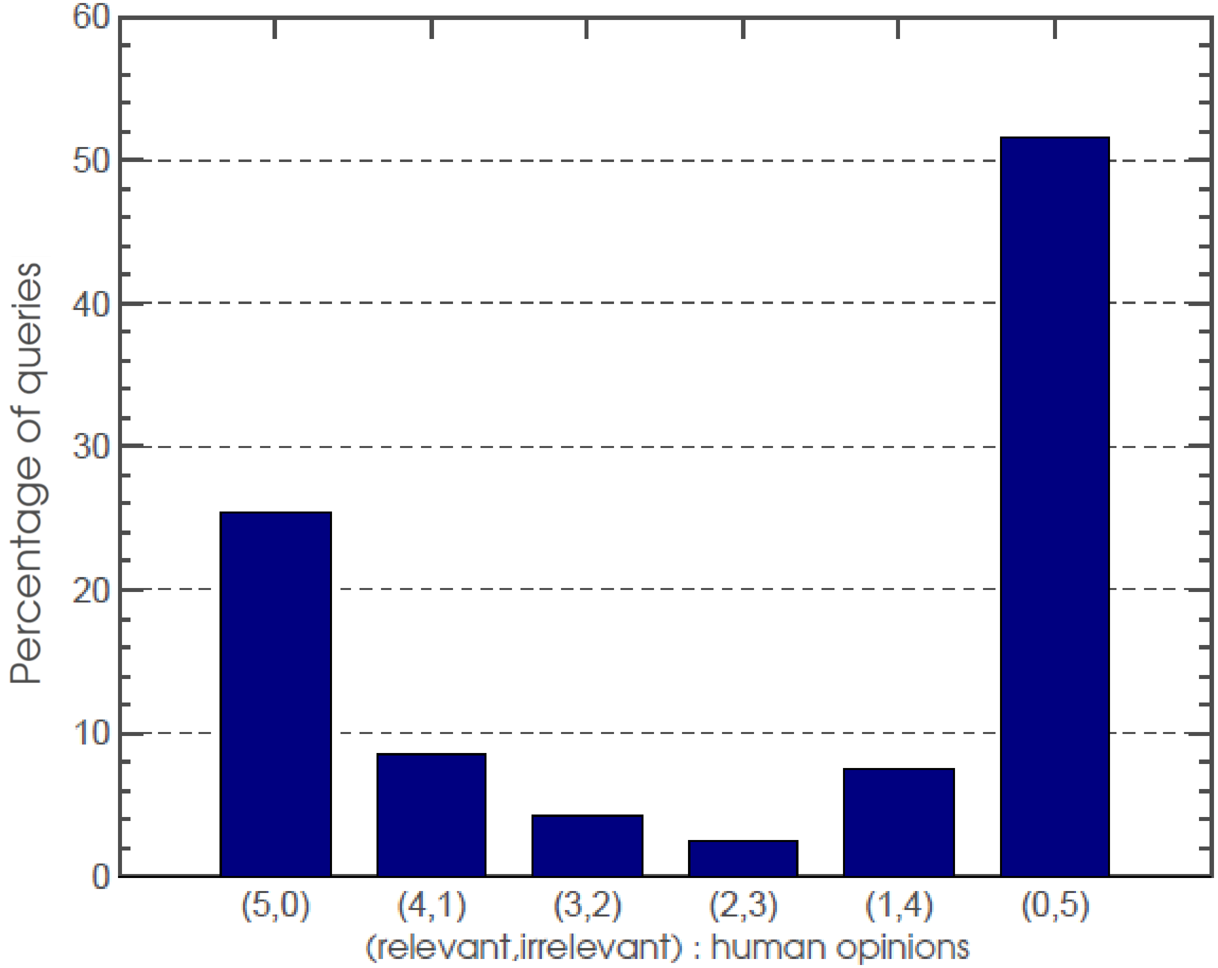
"What did this place (MPI-INF) look like in December?"

Outline

- Motivation and Overview
- Contextual Media Retrieval System
- Results and Conclusion

Evaluation

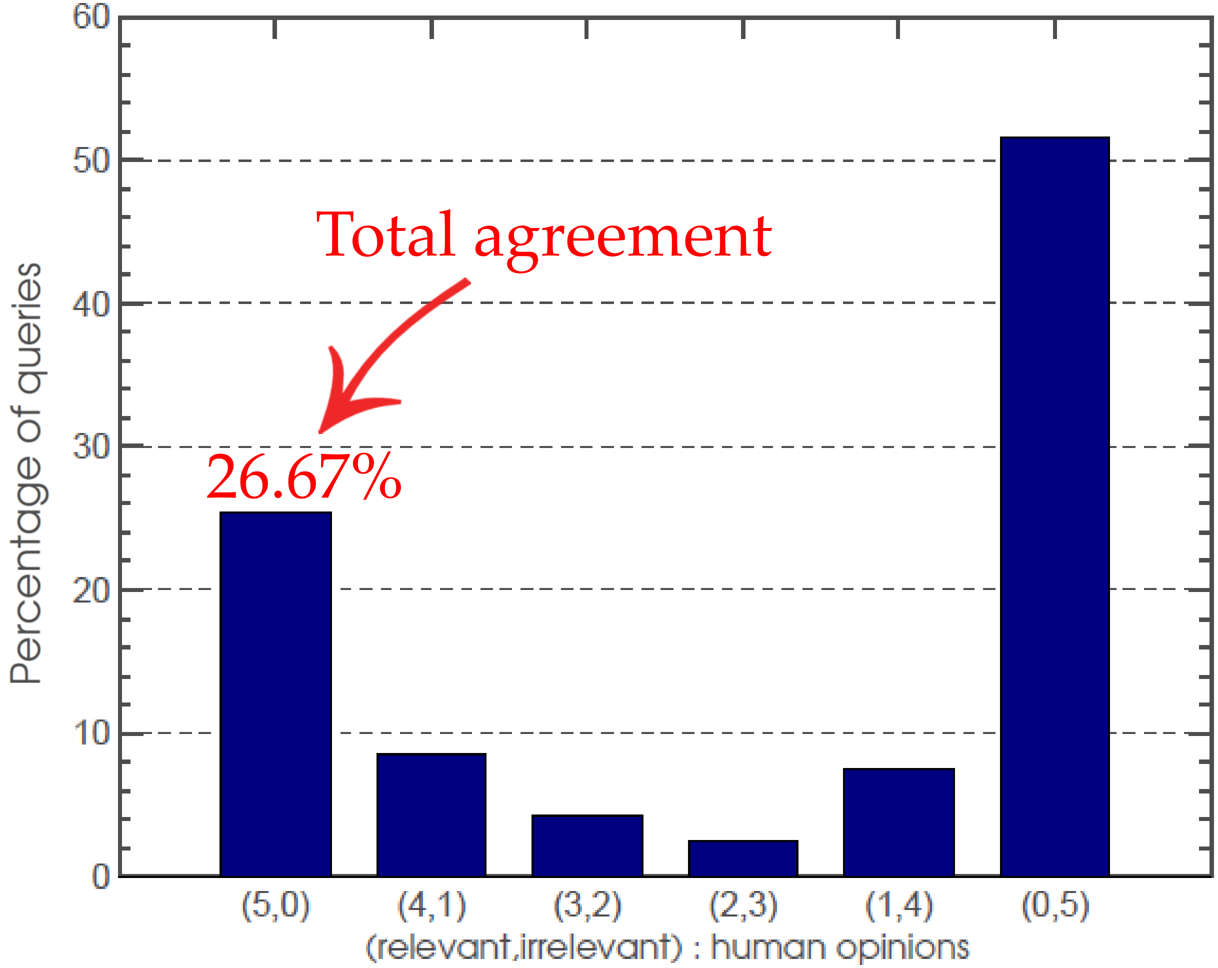
Agreement and Disagreement between users



* Model tested on 500 test queries

Evaluation

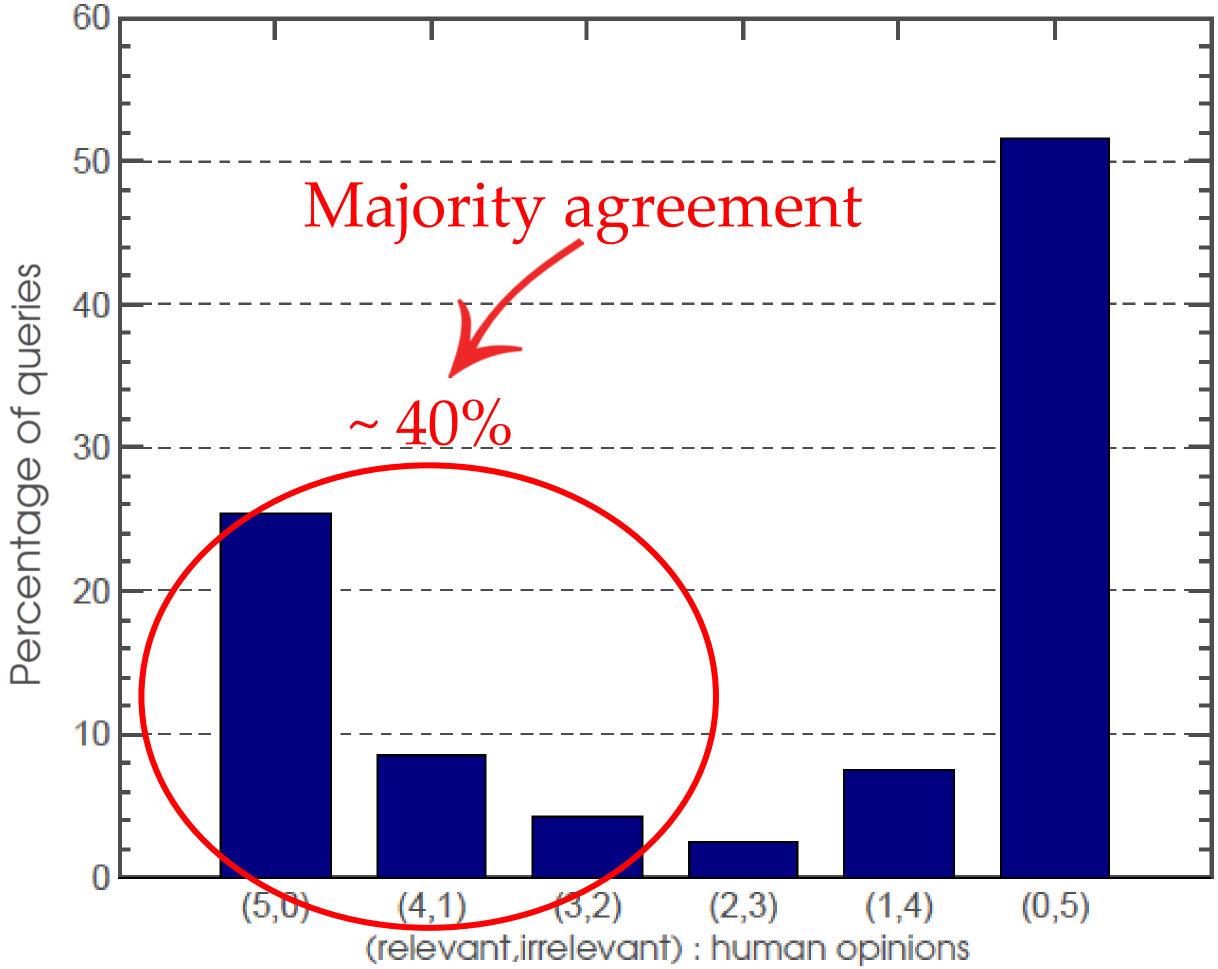
Agreement and Disagreement between users



* Model tested on 500 test queries

Evaluation

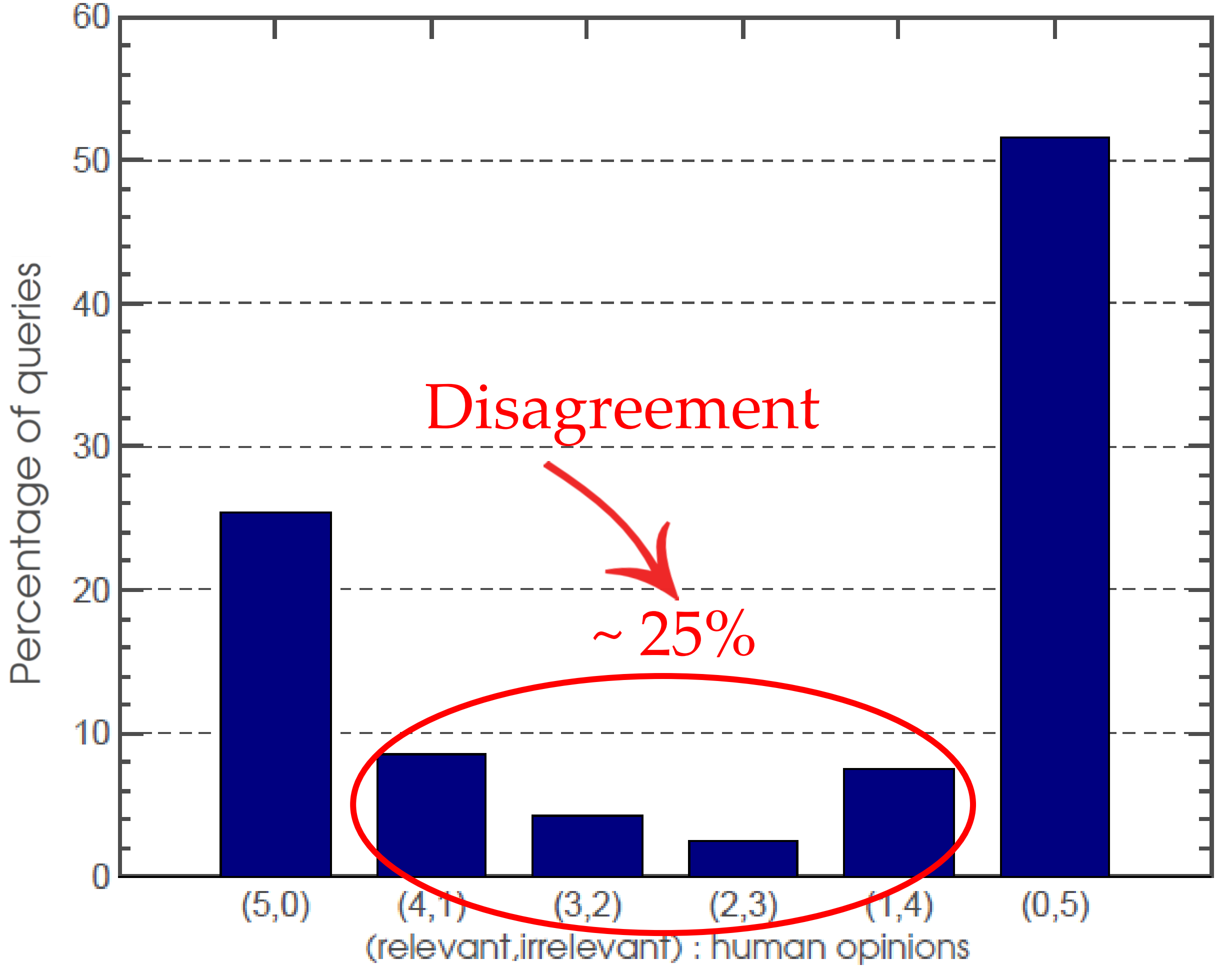
Agreement and Disagreement between users



* Model tested on 500 test queries

Evaluation

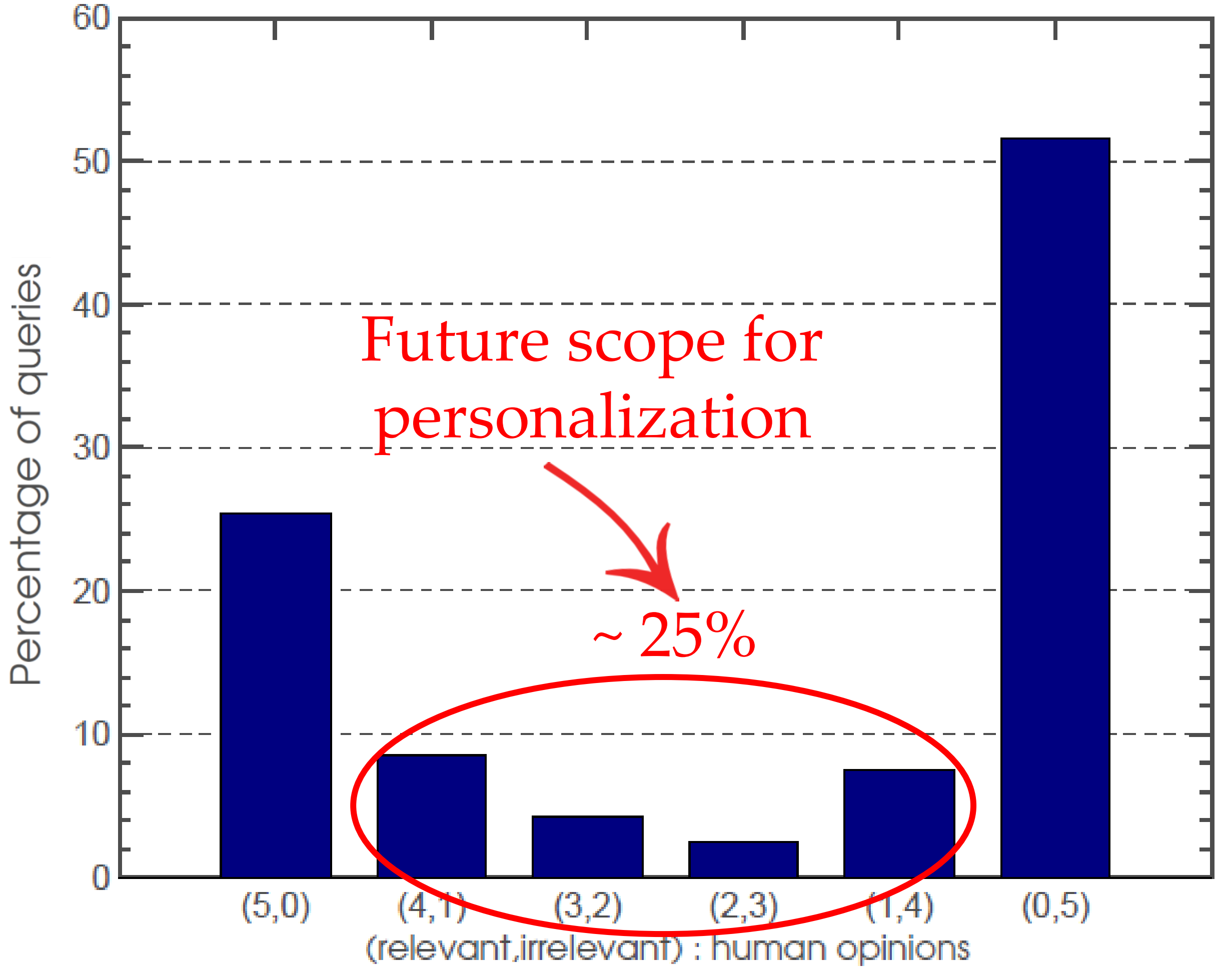
Agreement and Disagreement between users



* Model tested on 500 test queries

Evaluation

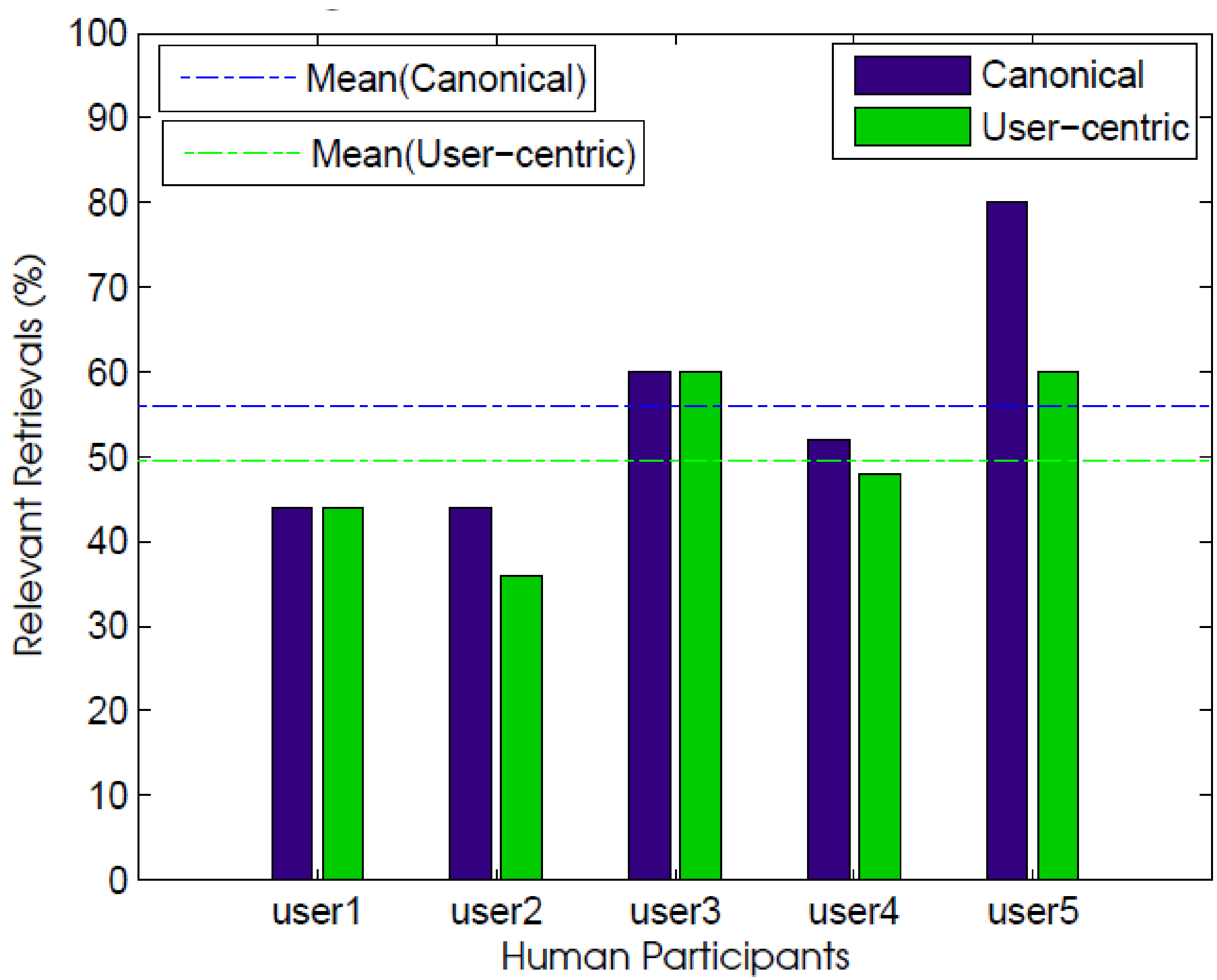
Agreement and Disagreement between users



* Model tested on 500 test queries

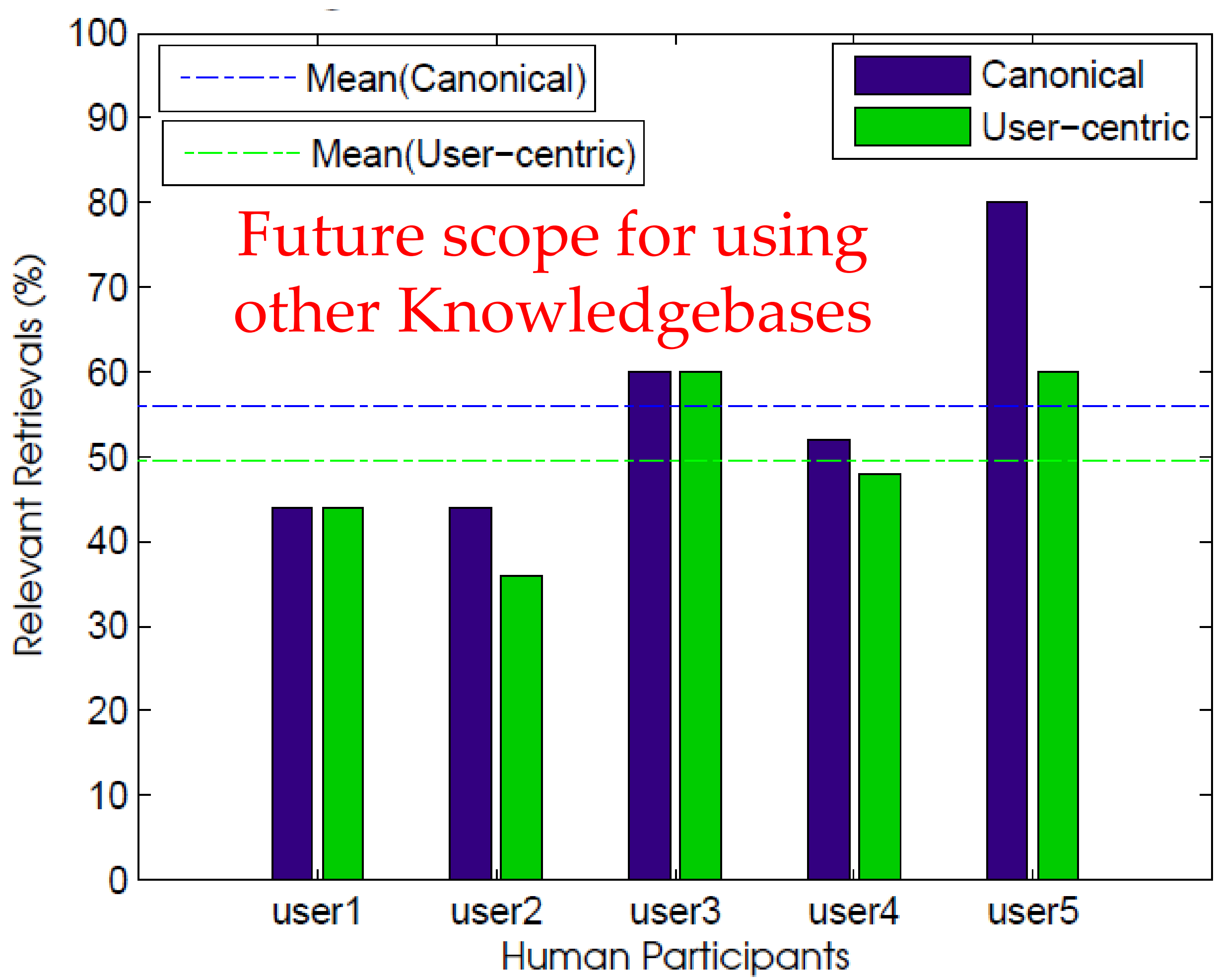
Evaluation

Study of human reference frame resolution



Evaluation

Study of human reference frame resolution



Summary

We have:

- Instantiated a ***"Collective Memory"*** of media content
- Developed a novel architecture for media retrieval with natural language voice queries in a dynamic setting - ***Xplore-M-Ego***
- Integrated ***'egocentrism'*** to media retrieval

Summary

We have:

- Instantiated a ***"Collective Memory"*** of media content
- Developed a novel architecture for media retrieval with natural language voice queries in a dynamic setting - ***Xplore-M-Ego***
- Integrated ***'egocentrism'*** to media retrieval

Thank You

References

- Photo Tourism: Exploring Photo Collections in 3D
  Noah Snavely, Steven M. Seitz, Richard Szeliski

- Video Collections in Panoramic Contexts
  J.Tompkin, F.Pece, R.Shah, S.Izadi, J.Kautz, C.Theobalt

- Videoscapes: Exploring Sparse, Unstructures Video Collections
  J.Tompkin, K. In Kim, J.Kautz, C.Theobalt

- PhotoScope:Visualizing Spatiotemporal Coverage of Photos for Construction Management
  F.Wu, M.Tory

References

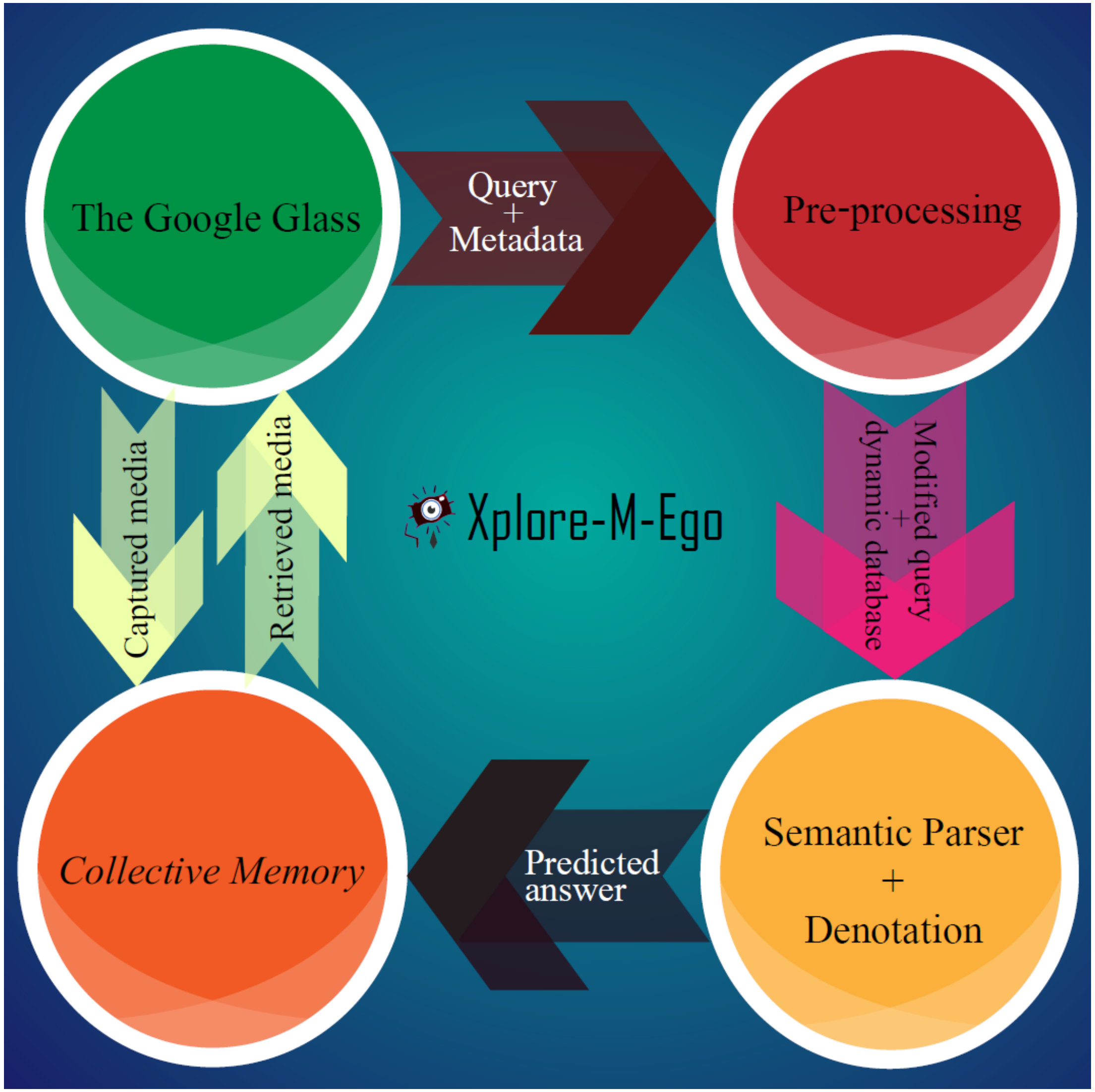
- Learning Dependency-Based Compositional Semantics
  Percy Liang, Michael I. Jordan, Dan Klein
- A multi-world approach to question answering about real-world scenes based on uncertain input
  M. Malinowski, M. Fritz
- Image Retrieval with Structured Object Queries Using Latent Ranking SVM
  T.Lan, W.Yang, Y.Wang, G.Mori
- Interpretation of Spatial Language in a Map Navigation Task
  M. Levit, D. Roy

Extra Material

Contribution

- Instantiation of a ***"Collective Memory"*** of media files
- Extension of **question-answering** to a **dynamic** setting
- Extension of **spatio-temporal exploration** of media to a dynamic setting
- Incorporation of **'egocentrism'** to media retrieval
- Use of **natural language voice queries** for media retrieval

System Overview



<u>**Modules of *Xplore-M-Ego***</u>

- **The Google Glass:**
  User Interface
- **Pre-processing :**
  Modification of query,
  Mapping of a dynamic
  environment to a static
  environment
- **Semantic Parser + Denotation :**
  Semantic parsing and
  prediction of answer
- **Collective Memory :**
  Store of media files

Related Work

- Spatio-temporal Media Retrieval

Paper	Author(s)	Overview
Photo tourism: exploring photo collections in 3D	N. Snavely, S. M. Seitz, and R. Szeliski	Exploration of popular world sites by browsing through images
Video collections in panoramic contexts	J. Tompkin, F. Pece, S. Rajvi, I. Shahram, K. Jan, and C. Theobalt	Spatio-temporal exploration of videos embedded on a panoramic context

Related Work

- Natural Language Question-Answering

Paper	Author(s)	Overview
Learning Dependency-based compositional Semantics	P. Liang, M. I. Jordan, and D. Klein	Training of a semantic parser with question-answer pairs; single static world approach
A multi-world approach to question answering about real-world scenes based on uncertain input	M. Malinowski and M. Fritz	Question-answering task based on real world indoor images; static multi-world approach

Related Work

- Media Retrieval with Natural Language Queries

Paper	Author(s)	Overview
Towards surveillance video search by natural language query	S. Tellex and D. Roy	Retrieval of video frames from surveillance videos with spatial relations “across” and “along”
Image retrieval with structured object queries using latent ranking SVM	T. Lan, W. Yang, Y. Wang, and G. Mori	Retrieval of images based on scene contents using short structured phrases as queries

Data Collection

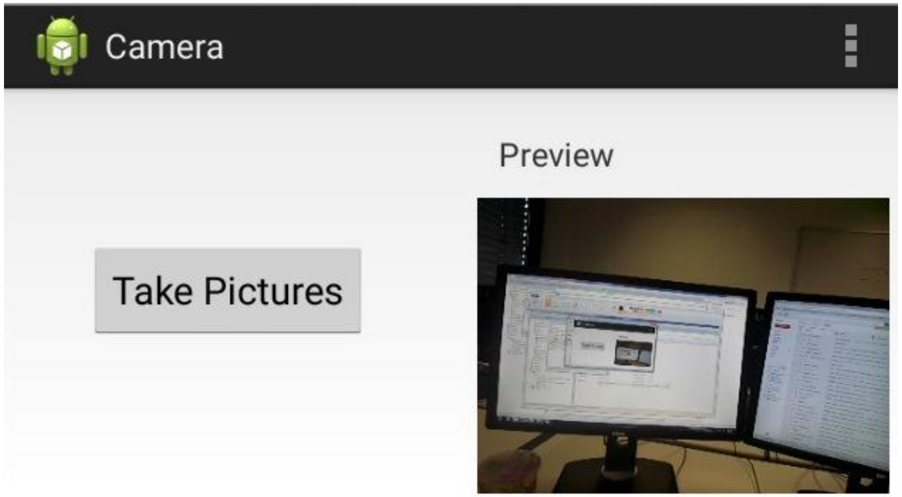
1. Map information : OpenStreetMap

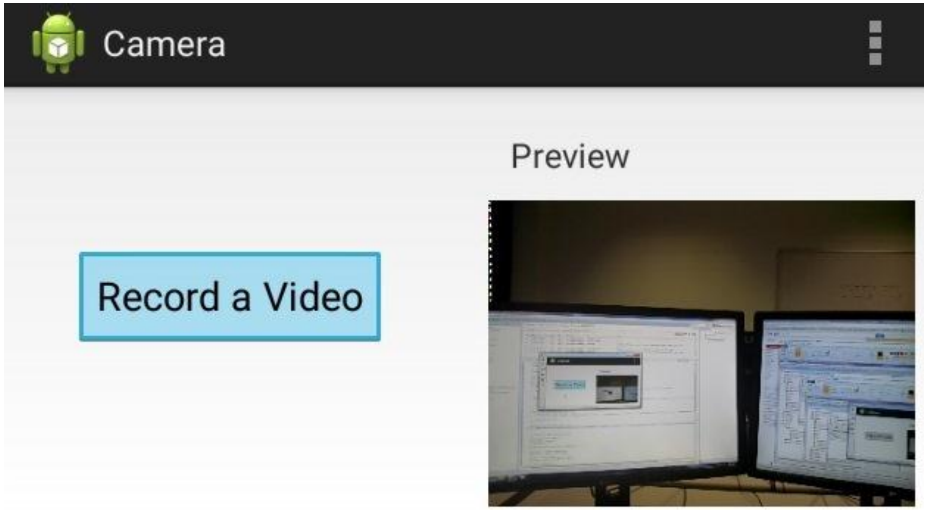
Contains –

- Type of the entity
- GPS coordinates
- Name
- Address

Data Collection

2. Collection of media files : *Collective Memory*





** Media files were captured with smart phones

Data Collection

3. Training and Test data

- ❑ Synthetically-generated Data

(“What is there in front of MPI-INF?”, answer(A, (frontOf(A, ‘mpi inf’))))
(“What is there behind MPI-INF?”, answer(A, (behind(A, ‘mpi inf’))))
(“What is there on the right of MPI-INF?”, answer(A, (rightOf(A, ‘mpi inf’))))
(“What is there on the left of MPI-INF?”, answer(A, (leftOf(A, ‘mpi inf’))))

- ❑ Real-world Data

(“What is there on the left of MPI-INF?”, ‘img_20141102_123406’)
(“What is on the left of MPI-INF?”, ‘img_20141113_160930’)
(“What is to the left of MPI-INF?”, ‘img_20141109_134914’)
(“What is on the left side of MPI-INF?”, ‘img_20141115_100705’)

Data Collection

“What is there beside MPI-INF?”

“What is on the left of E 1.3?”

“What is in front of the campus center?”

“How does the campus bus stop look?”

“What is there on the right side of the university campus?”

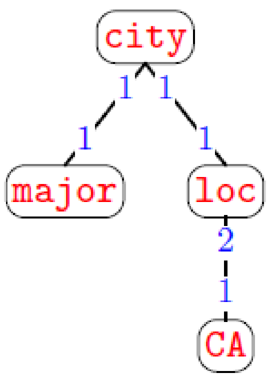
“What is in front of the university bus terminal?”

Semantic Parser

Dependency-based Compositional Semantics (DCS) by Percy Liang

Example: *major city in California*

$z = \langle \texttt{city}; \tfrac{1}{1}{:}\langle \texttt{major} \rangle ; \tfrac{1}{1}{:}\langle \texttt{loc}; \tfrac{2}{1}{:}\langle \texttt{CA} \rangle\rangle\rangle$



$$\begin{aligned}&\lambda c\, \exists m\, \exists \ell\, \exists s\,.\\ &\quad \texttt{city}(c) \wedge \texttt{major}(m) \wedge\\ &\quad \texttt{loc}(\ell) \wedge \texttt{CA}(s) \wedge\\ &\quad c_1 = m_1 \wedge c_1 = \ell_1 \wedge \ell_2 = s_1\end{aligned}$$

(a) DCS tree (b) Lambda calculus formula

(c) Denotation: $[\![z]\!]_w = \{\texttt{SF}, \texttt{LA}, \ldots\}$

- DCS tree defines relations between predicates
- Denotation are solutions satisfying the relations
- *city, major, loc, CA* are predicates

UNIVERSITÄT DES SAARLANDES max planck institut informatik CLUSTER OF EXCELLENCE PERCEPTUAL USER INTERFACES perceptual.mpi-inf.mpg.de Scalable Learning Perception scalable.mpi-inf.mpg.de

Semantic Parser

World(w):

state('california','ca', 'sacramento', 23.67e+6, 158.0e+3,31, 'los angeles', 'san diego', 'san francisco', 'san jose').

city('alabama','al','birmingham',284413).

river('arkansas',2333,['colorado','kansas', 'oklahoma','arkansas']).

mountain('alaska','ak','mckinley',6194).

road('86',['massachusetts','connecticut']).

country('usa',307890000,9826675).

Example Questions

"What is the highest point in Florida?"

"Which State has the shortest river?"

"What is the capital of Maine?"

"What are the populations of states through which the Mississippi river run?"

"Name all the lakes of US?"

**slide courtesy: Percy Liang

Semantic Parser

Learning in DCS

Objective:

$$\max_{\theta} \sum_{z} \underbrace{p(y \mid z, w)}_{\textbf{Interpretation}} \; \underbrace{p(z \mid x, \theta)}_{\textbf{Semantic parsing}}$$

EM-like Algorithm:

parameters θ

$(0.3, -1.4, \ldots, 0.6)$ $\xrightarrow{\text{enumerate/score DCS trees}}$ k-best list

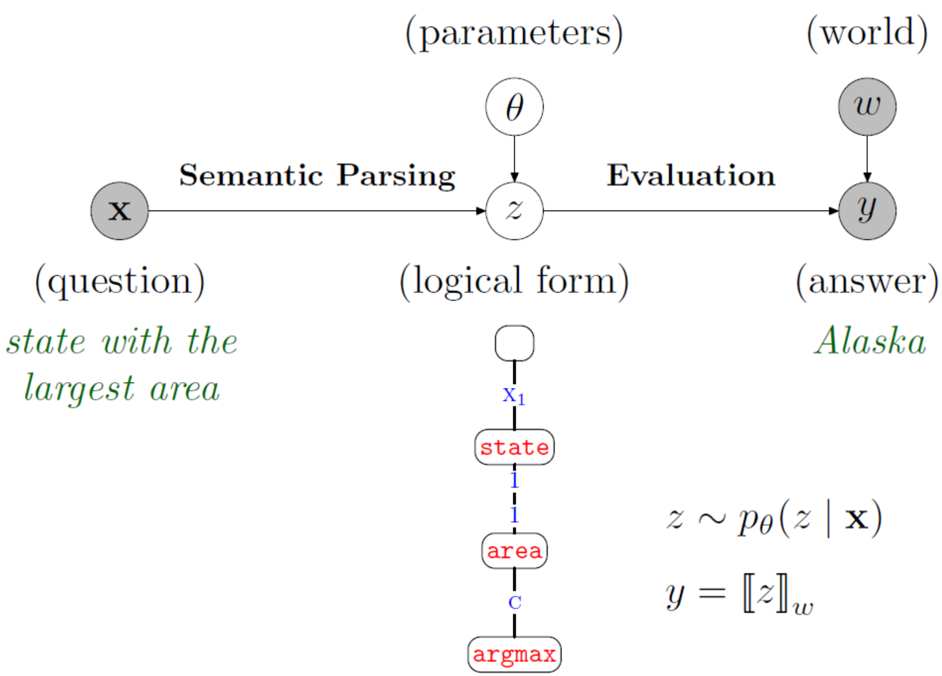
k-best list $\xrightarrow{\text{numerical optimization (L-BFGS)}}$ $(0.3, -1.4, \ldots, 0.6)$

k-best list

- `tree3` ✓
- `tree8` ✓
- `tree2` ✗
- `tree4` ✗
- `tree9` ✗

Semantic Parser

- Induction of logical forms



$$z \sim p_\theta(z \mid \mathbf{x})$$

$$y = [\![z]\!]_w$$

- Logical forms (DCS trees) induced as latent variables according to a probability distribution parametrized with θ
- Answer y evaluated with respect to world w

Semantic Parser

- Induction of logical forms

Requirements –

A set of rules/predicates:

```
city(cityid(City,St)) :- state(State,St,_ ,_ ,_ ,_ ,City,_ ,_ ,_ ).
loc(cityid(City,St),stateid(State)) :- state(State,St,_,_ ,_ ,_ ,City,_ ,_ ,_ ).
river(riverid(R)) :- river(R,_ ,_ ).
loc(cityid(City,St),stateid(State)) :- city(State,St,City, ).
traverse(riverid(R),stateid(S)) :- river(R, ,States), member(S,States).
area(stateid(X),squared mile(Area)) :- state(X,_ ,_ ,_ ,Area,_ ,_ ,_ ,_ ,_ ).
population(countryid(X),Pop) :- country(X,Pop,_).
major(X) :- city(X), population(X,moreThan(150000)).
```

Semantic Parser

- Induction of logical forms

Requirements –

A set of lexical triggers(*L*):

<(*function words*; predicate)>

(*most*, size).
(*total*, sum).
(*called*, nameObj).

<([*POS tags*]; [predicates])>

(*WRB*; loc)
([*NN;NNS*]; [city,state,country,lake,mountain,river,place)
([*NN;NNS*]; [person,capital,population])
([*NN;NNS; JJ*]; [len,negLen,size,negSize,elevation)
([*NN;NNS; JJ*]; [negElevation,density,negDensity,area,negArea])
(*JJ*; major)

Augmented Lexicon(*L+*):

(*long*, len). (*large*, size).
(*small*, negSize). (*high*, elevation).

Media Retrieval from Denotations

World(w):

image(`img_20141111_165828',20141111,49.2566,7.0442,`november').

video(`vid_20141121_120149',20141121,49.2569, 7.0456,`november').

cafe('mensa',49.2560,7.0454).

building('mpi_inf',49.2578,7.0460).

bank('postbank',49.2556,7.0449).

Example Questions

"What is there on the right of MPI-INF?"

"What is there in front of postbank?"

"What is there on the left of Mensa?"

"What is there near Science Park?"

"What happened here one day ago?"

"What does this place look like in December?"

Dynamic-Egocentric Extension

Lexical triggers:

Basic lexicon *L*	**Augmented lexicon *L+***
([*WP,WDT*], [image,video]). (*NN*, [atm,building,cafe,highway,parking,research_institution, restaurant,shop,sport,tourism,university]). (*JJS*, [nearest]). ([*NN,NNS,VB*], [view]). (*VBD*, [view]). Prediction accuracy: 17.9%	(*front*, frontOf). (*behind*, behind). (*right*, rightOf). (*left*, leftOf). Prediction accuracy: 47%

Dynamic-Egocentric Extension

Static Database of Geographic Facts w_s

```
atm('postbank_atm',49.2573855,7.0430358,49.2574,7.0430).
bank('bank1saar',49.2545957,7.0401859,49.2546,7.0402).
bar('canossa',49.2572934,7.0429204,49.2573,7.0429).
building('department_of_culture',49.25343,7.0414877,49.2534,7.0415).
cafe('icoffee',49.2574952,7.0453556,49.2575,7.0454).
highway('campus',49.25573,7.0389795,49.2557,7.0390).
library('state_library',49.253353,7.038327,49.2534,7.0383).
parking('uni_nord',49.25751,7.041421,49.2575,7.0414).
research_institution('dfki',49.25717,7.041499,49.2572,7.0415).
```

Dynamic-Egocentric Extension

Dynamic Database of User Metadata w_{du}

```
person(49.2578,7.0454,'n').
day(20141104).
```

Dynamic Database of Media Content w_{dm}

```
image('img_20141111_165828',20141111,49.2566,7.0442,'november').
image('img_20141112_092045',20141112,49.2554,7.0396,'november').
video('vid_20141121_120149',20141121,49.2569,7.0456,'november').
video('vid_20141123_165241',20141123,49.2530,7.0338,'november').
```

POS tags from Penn Treebank

- WRB : Wh-adverb
- NN : Noun, singular or mass
- NNS : Noun, plural
- JJ : Adjective
- WP : Wh-pronoun
- WDT : Wh-determiner
- NN : Noun, singular or mass
- JJS : Adjective, superlative
- NNS : Noun, plural
- VB : Verb
- VBD : Verb, past tense

Reason behind hard-coding spatial relations

- What is there left/VBN of MPI?
- What is there on the left/NN of MPI?
- What is there in front/NN of MPI?
- What is there behind/IN MPI?
- What is there right/RB of MPI?
- What is there on the right/NN of MPI?

Predicates used in *Xplore-M-Ego*

Table 4.1: Definitions of predicates in our DCS

	Predicates	Definitions	Example Query
Spatial	`frontOf(A,B)`	lat(B)>lat(A), lon(A)=lon(B)	"what is in front of A?"
	`behind(A,B)`	lat(B)<lat(A), lon(A)=lon(B)	"what is behind A?"
	`rightOf(A,B)`	lon(B)>lon(A), lat(A)=lat(B)	"what is on the right of A?"
	`leftOf(A,B)`	lon(B)<lon(A), lat(A)=lat(B)	"what is on the left of A?"
Temporal	`view2(M,B)`	month(B)=M, lat(B)=user's lat, lon(B)=user's lon	"how did this place look in M?"
	`view1(B)`	timestamp(B)=user's timestamp, lat(B)=user's lat, lon(B)=user's lon	"what happened here 5 days ago?"

Here, B is a media file. A is a geographical entity (e.g. 'MPI') and M is a month (e.g. 'May') uttered as part of the query; 'lat' and 'lon' stand for GPS latitude and longitude; `day` and `person` are predicates in w_{du}

UNIVERSITÄT DES SAARLANDES

max planck institut informatik

CLUSTER OF EXCELLENCE

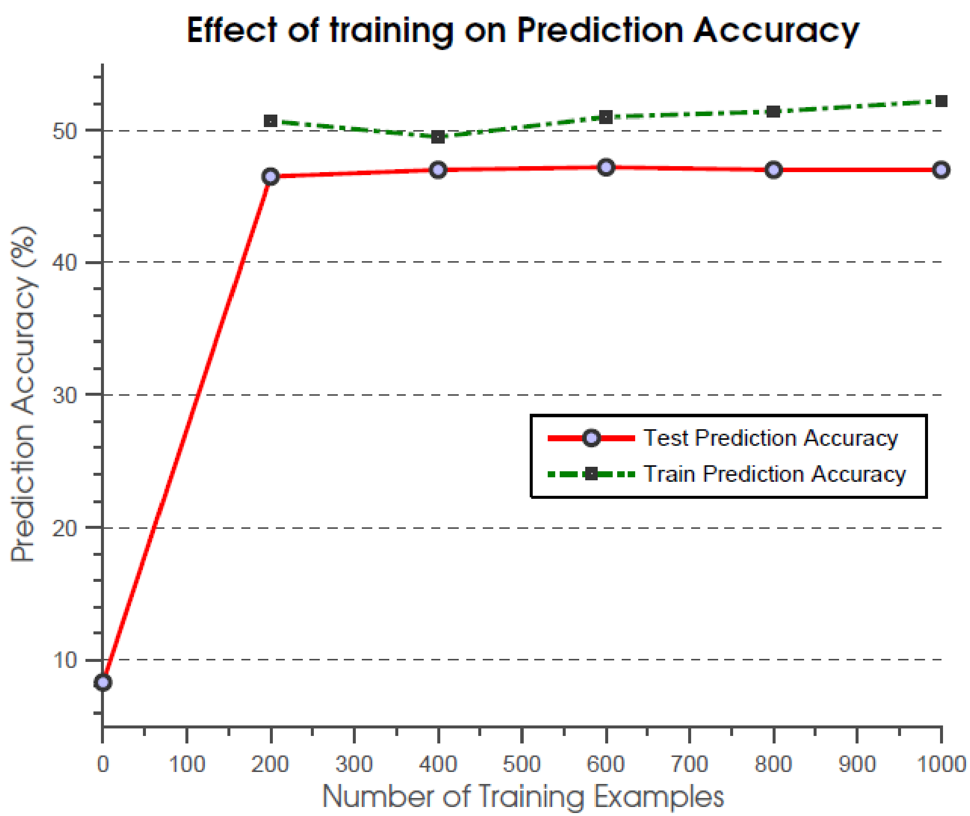
PERCEPTUAL USER INTERFACES

perceptual.mpi-inf.mpg.de

Scalable Learning Perception

scalable.mpi-inf.mpg.de

Results and Evaluation



- Synthetically generated question-answer pairs used for training and testing
- Maximum prediction accuracy – 47%

Results and Evaluation

Performance Measures:

- $q_m = number\ of\ queries\ with\ media\ retrievals$
- $q_r = number\ of\ queries\ with\ relevant\ retrievals\ among\ qm$
- $q_t = number\ of\ queries\ with\ textual\ retrievals\ and\ no\ retrievals$
- $average\ precision\ = \frac{q_r}{q_m}$
- $average\ recall\ = \frac{q_r}{q_m + q_t}$

Results and Evaluation

- "human-in-the-loop" training of the model
  - Five different models were trained
  - Training accuracies ranged from 42.6% to 48.8%
  - The best model based on training accuracy was used for further evaluations

Results and Evaluation

- “human-in-the-loop” training of the model

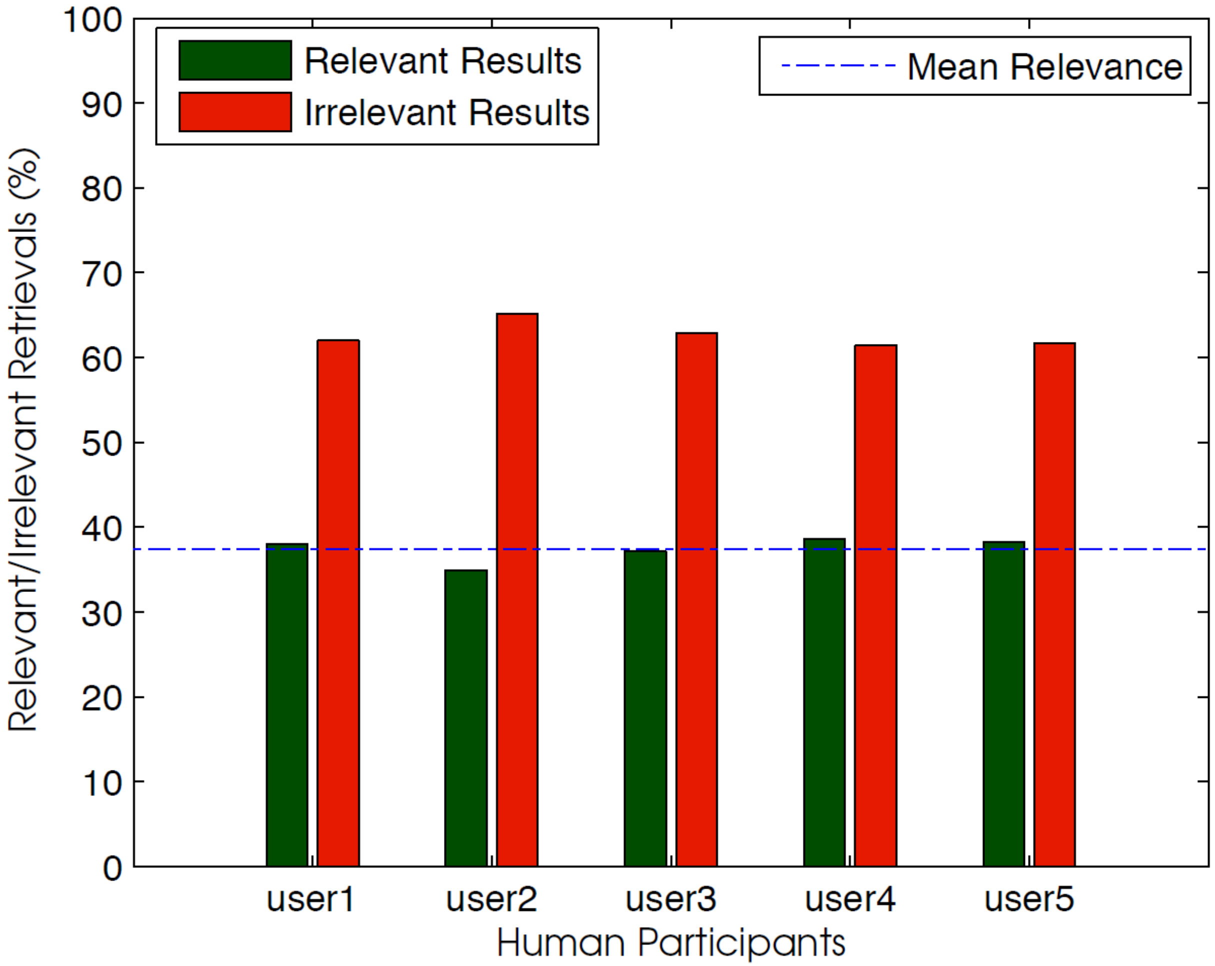
- It is a method of training the semantic parser by human users through relevance feedback
- “Correct”/“Wrong” decisions are made solely based on the predicted answers
- The models are trained with real questions from human users

Results and Evaluation

- “human-in-the-loop” training of the model
  - Automatic training of the semantic parser with the real data was not possible because –
    - GPS coordinates of media files showing a particular entity does not match that of the map data
    - Humans are inconsistent with regards to reference frames
    - Question-answer pairs didn’t follow any pattern
    - Denotations (often more than one answer) never matched with true answers, hence EM-like algorithm failed to learn

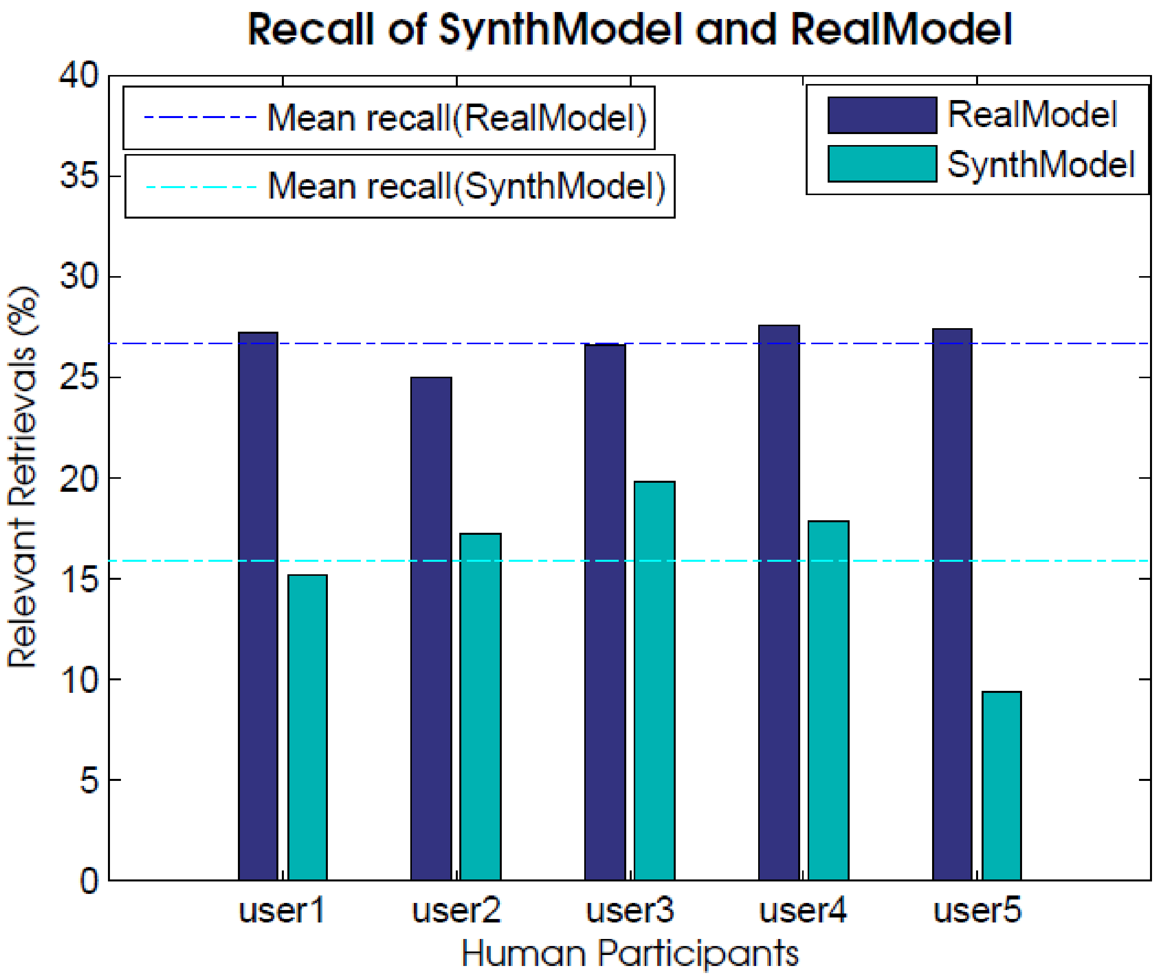
Results and Evaluation

Human evaluation of model trained with real-world data



- RealModel -model trained with real-world data
- Relevance feedback collected from five users
- Overall percentage of relevant retrievals = 26.67%

Results and Evaluation



- Recall of SynthModel = 15.88%
- Recall of RealModel = 26.67%

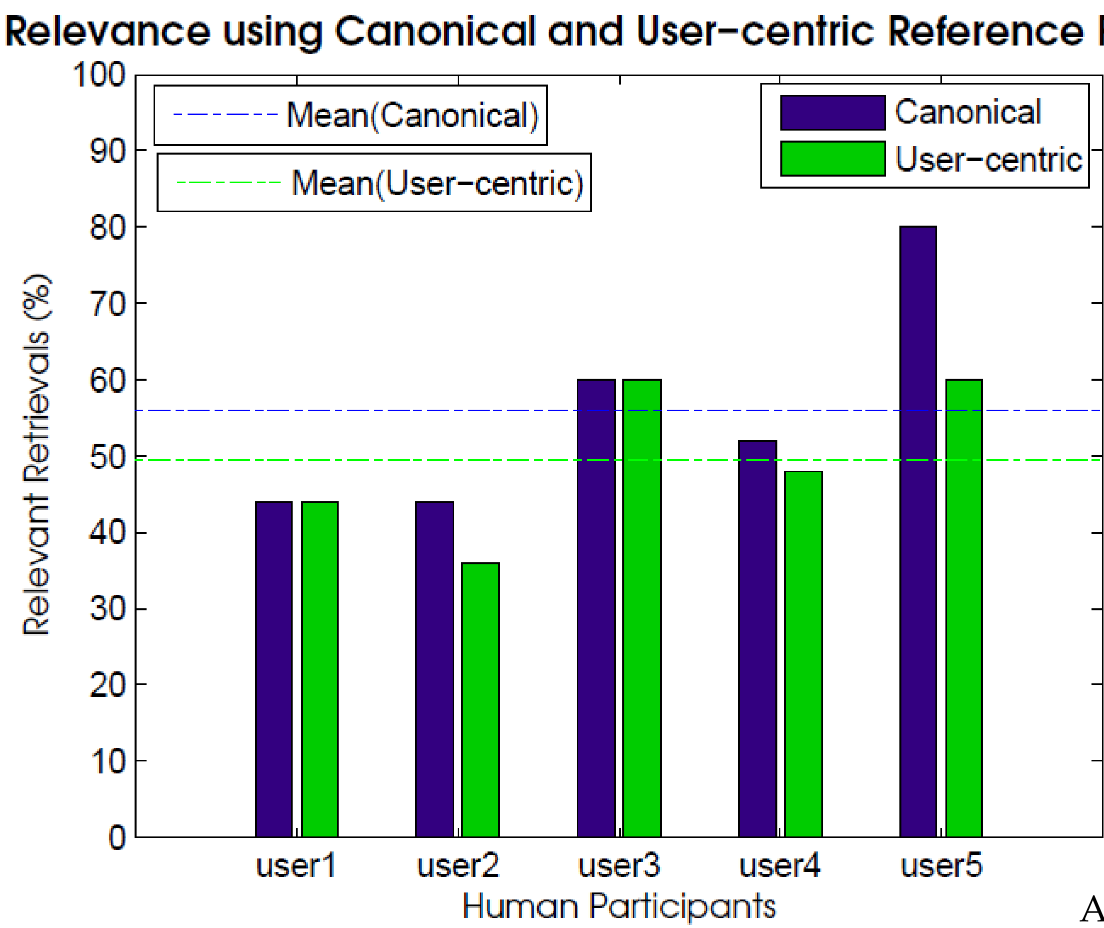
Evaluation

Human evaluation of temporal and contextual Q&A

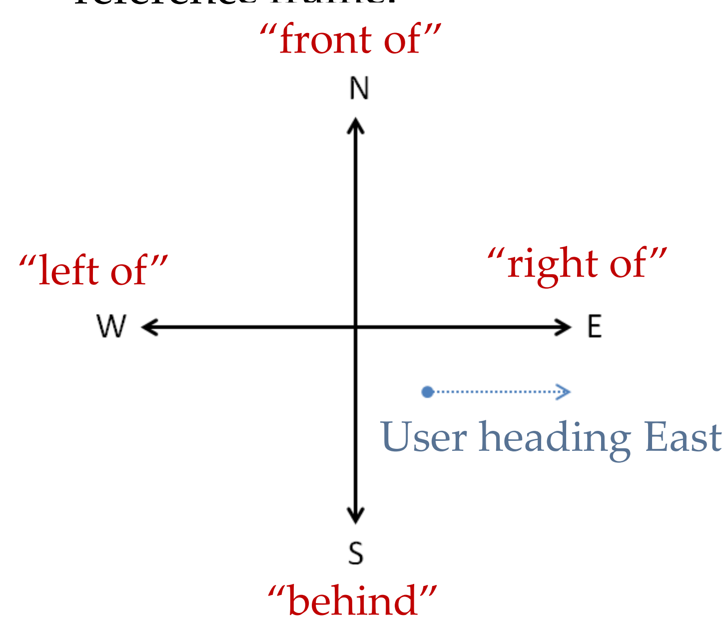
- Five hypothetical locations and viewing directions provided to users
- Relevance feedback collected for retrievals following a canonical reference frame and a user-centric reference frame

Evaluation

Human evaluation of temporal and contextual Q&A



- Canonical and User-centric reference frame:

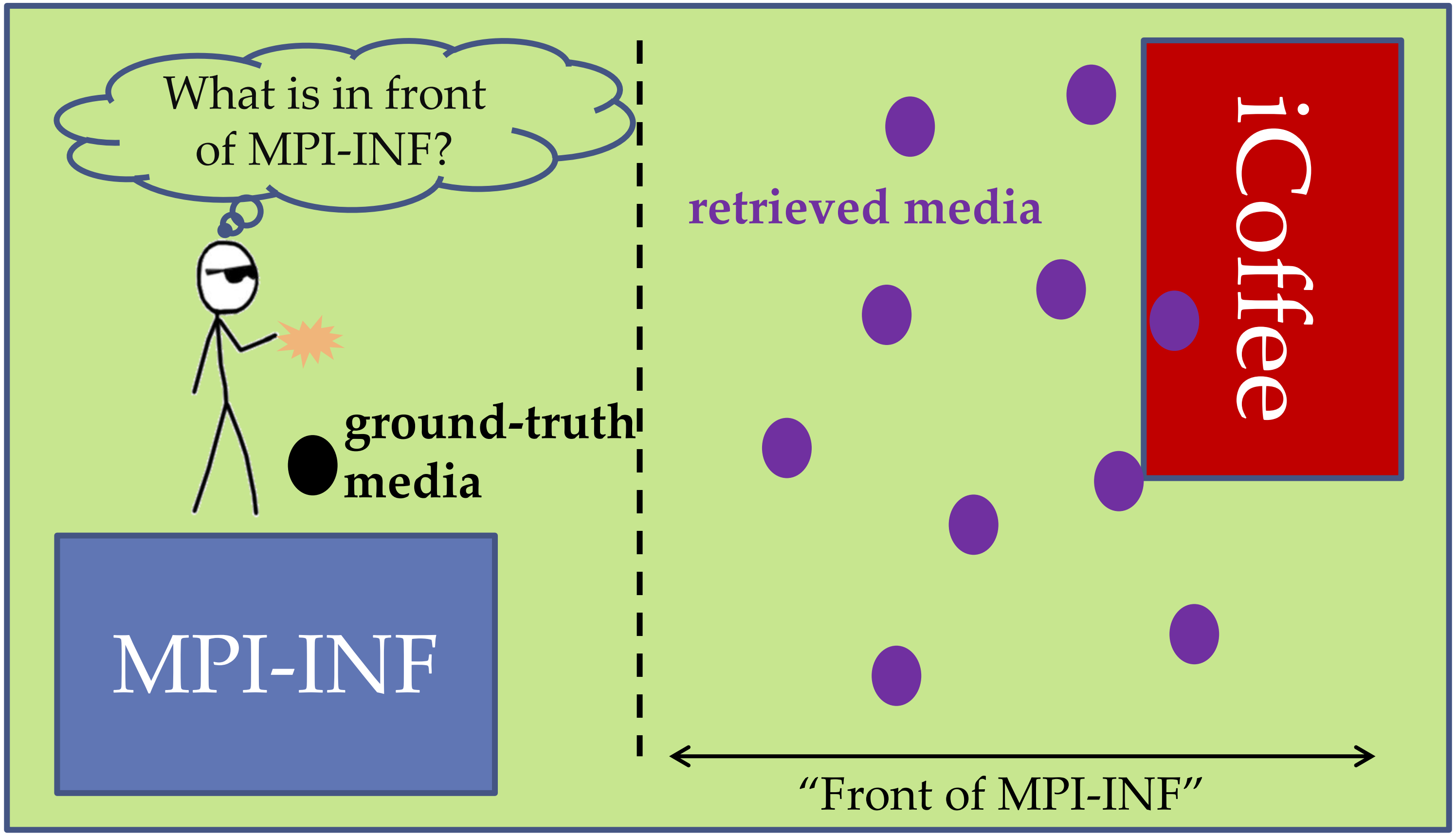


Original: What is there <u>in front of</u> MPI-INF?

Altered: What is there <u>on the right of</u> MPI-INF?

Discussion

Problem with matching GPS coordinates



Discussion

Challenges	Limitations
Converting a dynamic world to a static world	Spatial and temporal references not identified
Integrating ‘egocentrism’	Words tagged with incorrect POS tags
Handling temporal queries	Arguments not identified from sentences
Collection of data	Scalability
Increasing the coverage of the static database	Reference resolution is not handled

Discussion

Accuracy of Performance

- Matching the exact GPS coordinates for retrievals proved to be a failure

- It was handled by rough localization by rounding the GPS coordinates to the first 6 significant digits (49.2578401 -> 49.2578)

Failure case:

Query	Retrieved Images		
"What does MPI-SWS look like?"			

Discussion

Future Work

- Integration of image processing and computer vision methods for scene understanding (similar to Malinowski et al.)
- Development of a better semantic parser in light of our discussions about its limitations
- Development of more robust location sensors in devices used for capturing media
- Generation of a consensus about reference frames for applications involving the use of spatial relations

Summary of Quantitative Results

Table 6.2: Use of Lexicons L and $L+$

	Untrained Model	Trained Model
Basic Lexicon L	6%	17.9%
Augmented Lexicon $L+$	11.23%	47%

Table 6.3: Average Precison and Average Recall of semantic parser models

	Average Precision	Average Recall
SynthModel	50.2%	16%
RealModel	37.38%	28%

Table 6.4: Relevance feedback using different reference frames

	Canonical	User-centric
Mean	56%	49.6%
Standard Deviation	15%	10.4%