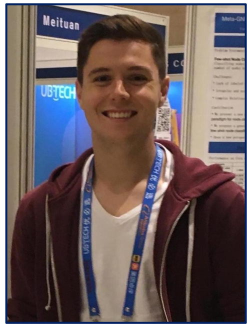
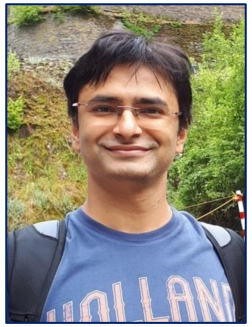
SMART Task 3.0

Entity and Relation Linking using CLOCQ

***Philipp Christmann**, Rishiraj Saha Roy, Gerhard Weikum*





Question Answering System

Who plays Viserys in GRRM's latest HBO series?

Paddy Considine

Question Answering System

Question Answering System

Who plays Viserys in GRRM's latest HBO series?

Harry Lloyd

Question Answering System

Entity Linking (EL)

"Viserys" ↦ `Viserys III Targaryen`
"GRRM" ↦ `George R. R. Martin`
"HBO" ↦ `HBO`

Link entity mentions in question to the KB entities
⇒ Fill SPARQL query slots
⇒ Initiate the search space (e.g. for graph processing)

Relation Linking (RL)

"plays" ↦ `plays for team`

Linking relations independently of EL results can be suboptimal!

Link relation mentions in question to KB relations
⇒ Fill SPARQL query slots
⇒ KB fact matching in search space

`Green linking`: correct
`Red linking`: wrong

Single error in EL/RL can lead to complete failure!

SMART Task 3.0

- ★ **Relation Linking Task**
- ★ **Entity Linking Task**
- ☆ **Answer Type Prediction Task**

Approach these tasks using CLOCQ

CLOCQ

- ★ **Unsupervised** linking tool
- ★ Parameters can be **tuned**
- ★ Publicly **available (**clocq.mpi-inf.mpg.de**)**

[Christmann et al., Beyond NED: Fast and Effective Search Space Reduction for Complex Question Answering over Knowledge Bases, WSDM 2022]

CLOCQ: Coherent Disambiguations

Who plays Viserys in GRRM's latest HBO series?

Approach both problems jointly

⇒ Mentions can lexically/semantically **match** with **many KB items**

"plays" ↦ `plays for team`, `instrument`, `number of plays`, `character role`, `time played`,…

⇒ By considering **question as a whole**, actual meaning becomes clear

⇒ Linked entities and relations in KB should be **coherent**

[Christmann et al., Beyond NED: Fast and Effective Search Space Reduction for Complex Question Answering over Knowledge Bases, WSDM 2022]

CLOCQ: Coherent Disambiguations

Who plays Viserys in GRRM's latest HBO series?

Retrieve candidate disambiguations **for all mentions** (phrases detected via TagME NER)

Score candidates based on

- **Semantic coherence:** semantic similarity among candidates
  - `HBO (network) - George R.R. Martin`
  - `Hollywood Bowl Orchestra (HBO) - George R.R. Martin`
- **KB connectivity:** KB distance among candidates
  - `HBO (network) - George R.R. Martin`
  - `Hollywood Bowl Orchestra (HBO) - George R.R. Martin`

global signals

- **Question relatedness:** candidate vs. question
  - `HBO (network) -` "who plays Viserys in GRRM's latest HBO series?"
  - `Hollywood Bowl Orchestra (HBO) -` "who plays Viserys in GRRM's latest HBO series?"
- **Lexical matching:** candidate vs. mention
  - `HBO (network) -` "HBO"
  - `Hollywood Bowl Orchestra (HBO) -` "HBO"

local signals

[Christmann et al., Beyond NED: Fast and Effective Search Space Reduction for Complex Question Answering over Knowledge Bases, WSDM 2022]

CLOCQ: Auto-k Setting

Who plays Viserys in GRRM's latest HBO series?

Make up for potential errors

⇒ Mentions can be highly **ambiguous**

"Viserys" ↦ `Viserys III Targaryen`, `Viserys I Targaryen`, `Viserys II Targaryen`,…

⇒ Consider **top-k** results

⇒ **k** set **automatically** based on **ambiguity** of individual mention

- ⇒ Entropy of candidate distribution

[Christmann et al., Beyond NED: Fast and Effective Search Space Reduction for Complex Question Answering over Knowledge Bases, WSDM 2022]

CLOCQ: Potential Results

Who plays Viserys in GRRM's latest HBO series?

Paddy Considine

"Viserys"

	KB item
1	Viserys III Targaryen
2	Viserys I Targaryen
3	Viserys II Targaryen

k = 3

"GRRM"

	KB item
1	George R. R. Martin

k = 1

"HBO"

	KB item
1	HBO (network)
2	HBO (company)

k = 2

"plays"

	KB item
1	character role
2	characters
3	instrument

k = 3

[Christmann et al., Beyond NED: Fast and Effective Search Space Reduction for Complex Question Answering over Knowledge Bases, WSDM 2022]

Post-hoc Pruning Module

Who plays Viserys in GRRM's latest HBO series?

CLOQC links ALL mentions

$\Rightarrow$ This helps sometimes, but is not always required

"latest" $\mapsto$	`latest start date, end time, latest release,…`
"series" $\mapsto$	`TV series, part of the series, series (rank),…`

Should not be linked for EL (or RL) task

$\Rightarrow$ Lead to noise

$\Rightarrow$ Decreased precision

$\Rightarrow$ **Idea:** Prune mentions that should not be disambiguated

$\Rightarrow$ Train **seq2seq** model to generate mentions in question

$\Rightarrow$ Consider only **entities linked** to **such mentions** (relaxed substring matching)

Post-hoc Pruning Module

Who plays Viserys in GRRM’s latest HBO series?

How to obtain training data for pruning module?

⇒ **Run CLOCQ on train set:** obtain (mention, entity) pairs for each instance

CLOCQ outputs

“Viserys”	↦ Viserys III Targaryen, Viserys I Targaryen,…
“GRRM”	↦ George R. R. Martin
“latest”	↦ latest start date, end time, latest release,…
“HBO”	↦ HBO (network), HBO (company),…
“series”	↦ TV series, part of the series, series (rank),…

Gold entities

Viserys I Targaryen

George R. R. Martin

HBO (network)

⇒ **Training instance**

Input: Who plays Viserys in GRRM‘s latest HBO series?

Output: Viserys|GRRM|HBO

Experimental Setup

Knowledge Base

$\Rightarrow$ **Wikidata –** 94M entities, 3k predicates

Metrics

$\Rightarrow$ **Precision –** how many predictions are correct?

$\Rightarrow$ **Recall –** how many of the relevant entities/relations are found?

$\Rightarrow$ **F1 –** harmonic mean between precision and recall

SMART 3.0: Entity Linking Task

Training instances	Test instances
49,987	13,794

DC-3 is operated by which **airline**?

`Douglas DC-3` `airline`

Methods

★ CLOCQ (k=1)
★ CLOCQ (k=AUTO)
★ CLOCQ (k=1) + Pruning
★ CLOCQ (k=AUTO) + Pruning

SMART 3.0: Relation Linking Task

Training instances	Test instances
24,112	6,029

Where was the **actress** from Frasier **born**?

`cast member`

`place of birth`

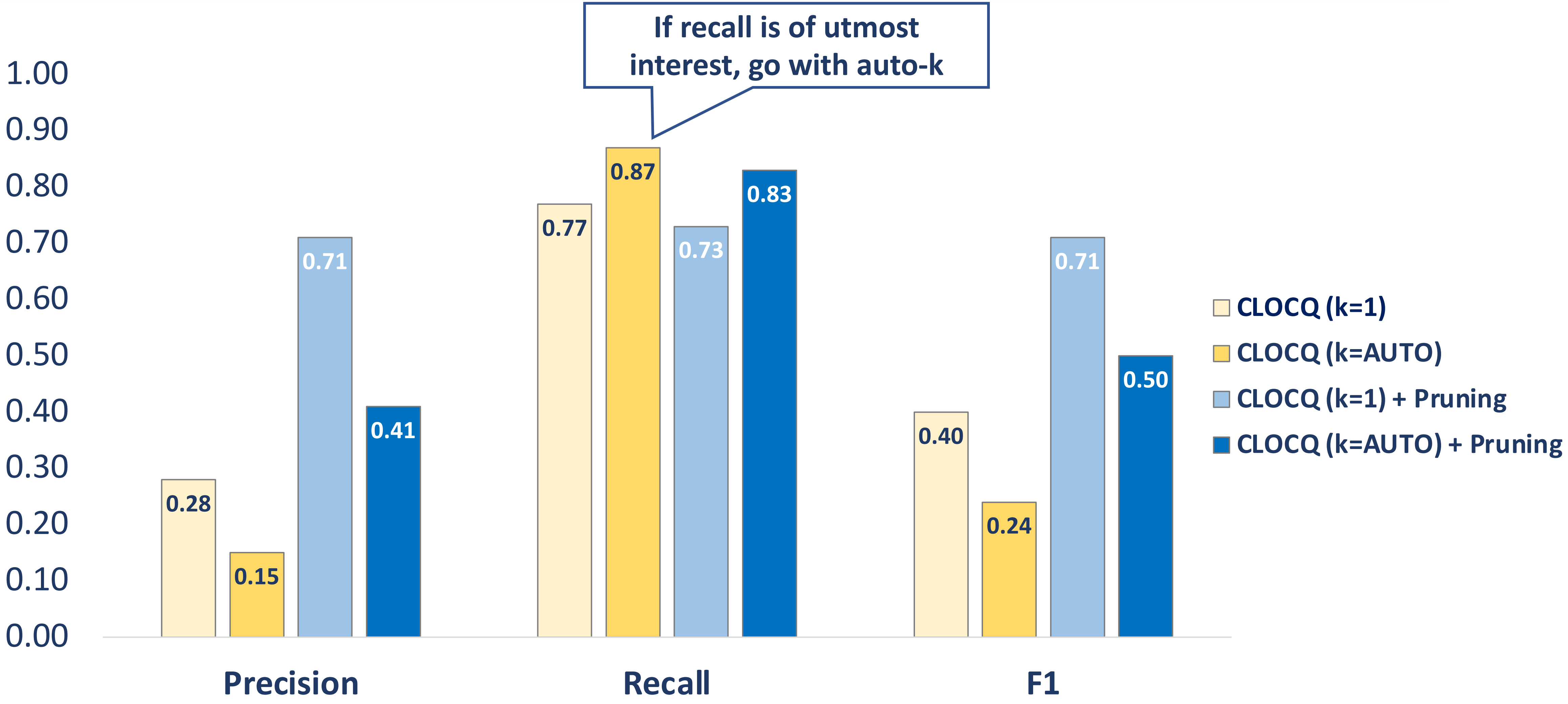
Methods

★ CLOCQ – top-1 relation per mention

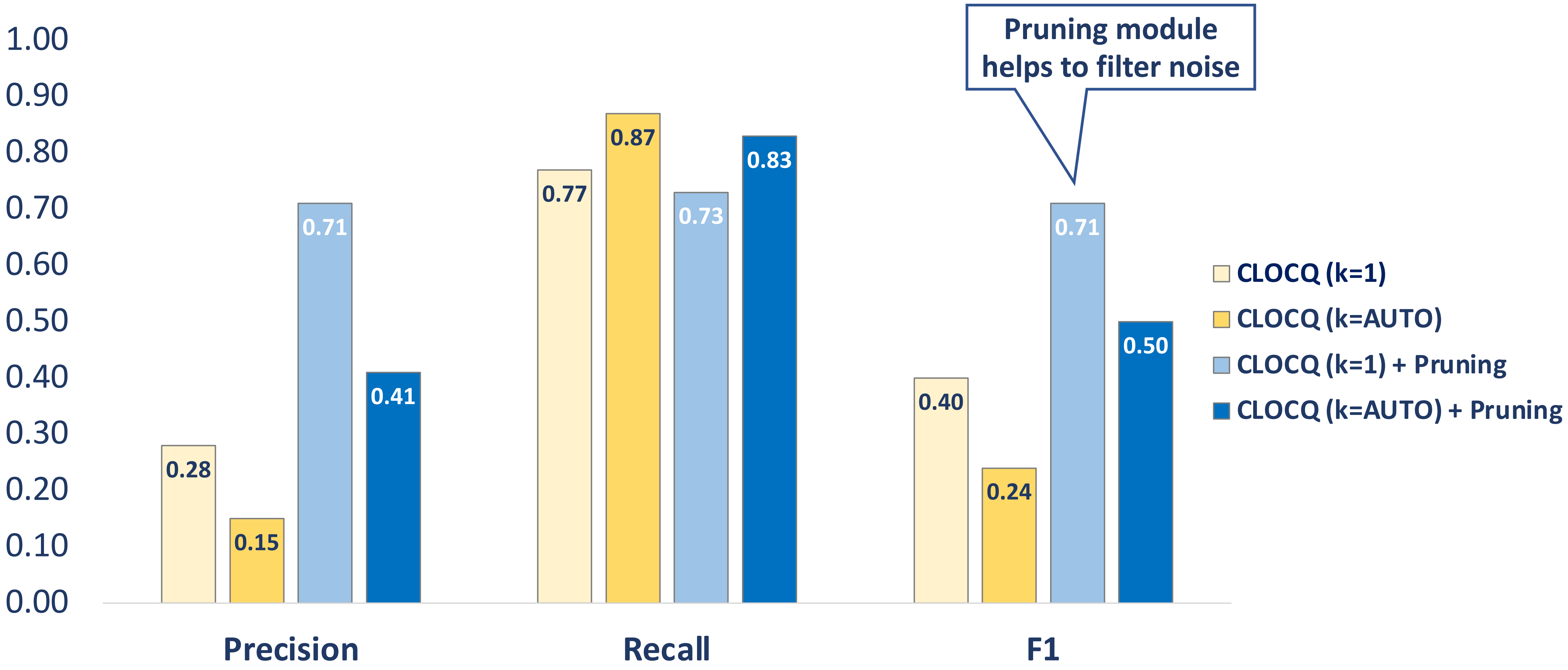
★ CLOCQ – all relations per mention

k=40

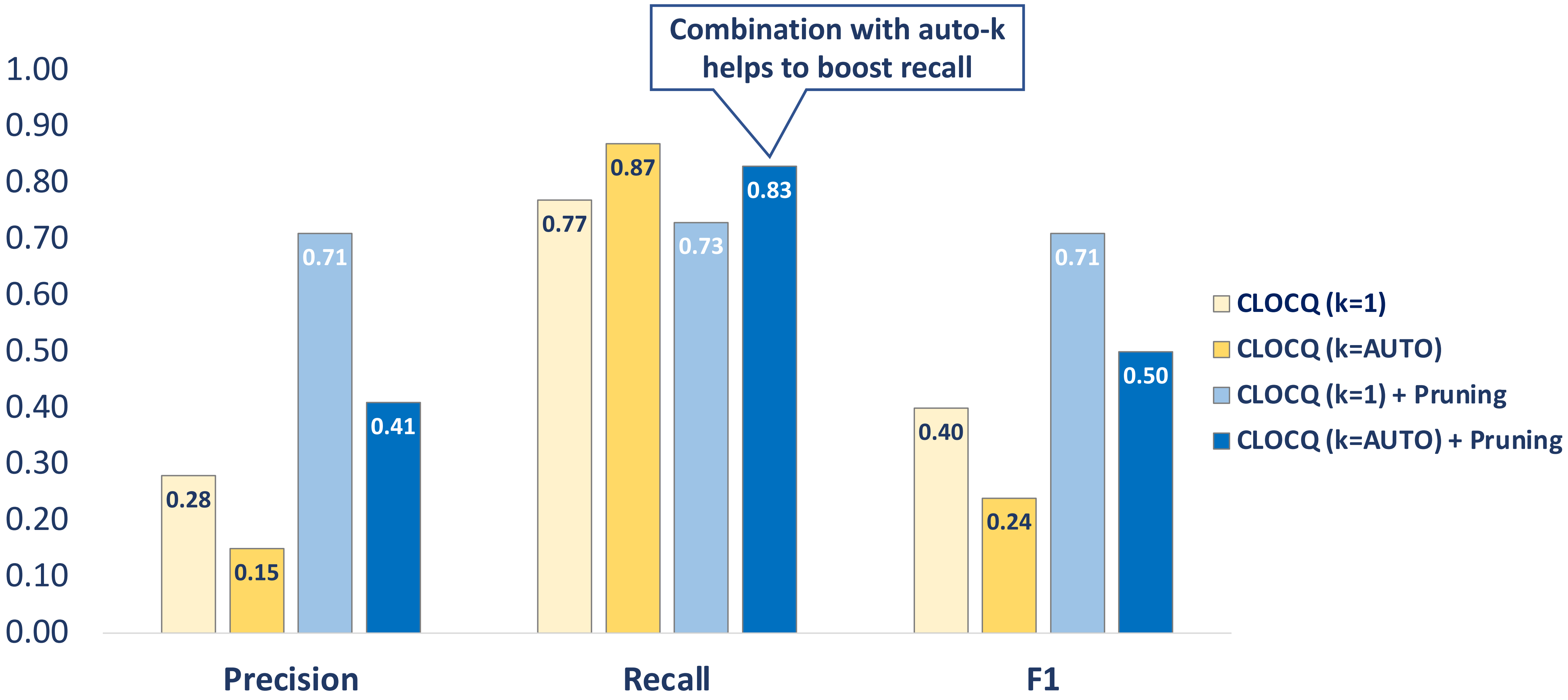
Entity Linking



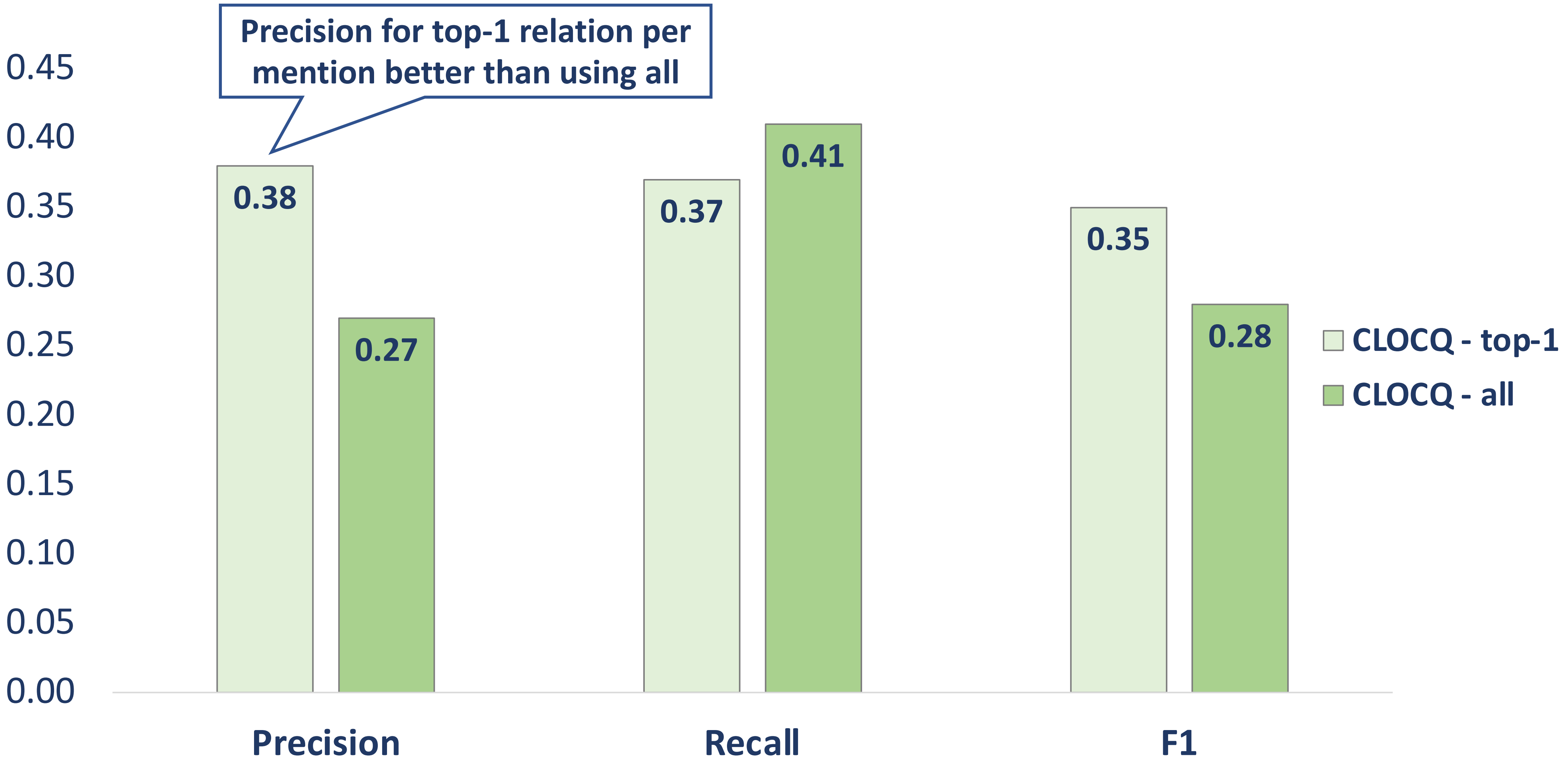
Entity Linking



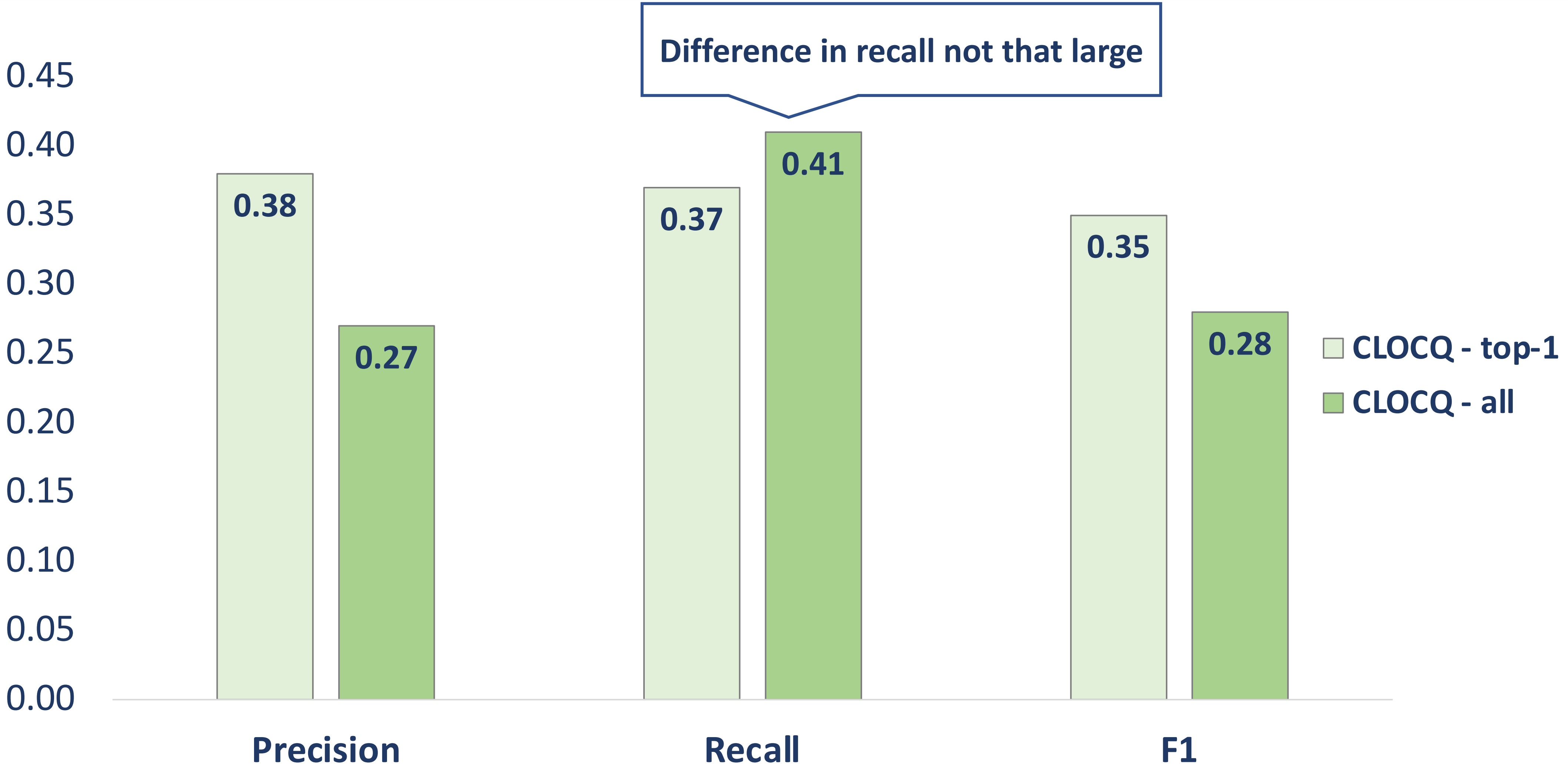
Entity Linking



Relation Linking



Relation Linking



EL – Anecdotal Examples

What was **Toby Wright**'s profession?

"Toby Wright" ↦ `Toby Wright (football player)`

CLOCQ (k=1)

"Toby Wright" ↦ `Toby Wright (football player)`
`Toby Wright (record producer)`

CLOCQ (k=AUTO)

which footballer was born in **middlesbrough**?

"middlesbrough" ↦ `Middlesbrough F.C. (football club)`

CLOCQ (k=1)

"middlesbrough" ↦ `Middlesbrough F.C. (football club)`
`Middlesbrough (borough)`
`Middlesbrough (town)`

CLOCQ (k=AUTO)

⇒ **Top-k** helps to **resolve ambiguity**

⇒ **Preferrable** if remaining QA system can deal with multiple linkings

RL – Anecdotal Examples

Which **child** of John Adams **died on** February 23, 1848?

`child`
`date of death`

"child" ↦ `child`	
"died on" ↦ `date of death`	

CLOCQ – top-1

"child" ↦ `child`	
"died on" ↦ `date of death` `people who died here`	

CLOCQ – all

What is the **point in time** that Nicolaus Cusanus **was made cardinal** by the Holy Roman Church?

`start time`
`position held`

"point in time" ↦ `point in time`

CLOCQ – top-1

"point in time" ↦ `point in time` `located in time zone`

CLOCQ – all

⇒ CLOCQ can **identify multiple** relations in a question

⇒ Some questions may require **deeper understanding**

Conclusion

CLOCQ

★ **Top-k** linkings for **all mentions** (entities, relation, types, concepts)
★ **k** set automatically for each **individual** mention
★ **Neural pruning** reduces noise in **post-hoc** manner

SMART Task 3.0

★ **Recall** of CLOCQ for **entity linking task well above 0.8**
★ **Trade-off** between recall and precision: best setting highly **depends on QA system**
★ **CLOCQ** is a **suitable** system for both tasks
☆ Integrating **neural components** in CLOCQ algorithm could **improve** results

⇒ SMART tasks can help to **identify, understand,** and **eliminate failure cases** of QA modules

clocq.mpi-inf.mpg.de

Thank you!