

Project Summary: Baseball KG

Fandel Lin, Zihao Han



Intro (Reminder)

Building MLB knowledge graph for player and game prediction.

KG Construction

- Major League Baseball (MLB)

Player-performance & match-result prediction (Extrapolation)

- Whether an event would take place in the future.



Time

t_1

STL

win

CWS

SF

win

LA

t_2

STL

lose

SF

CWS

lose

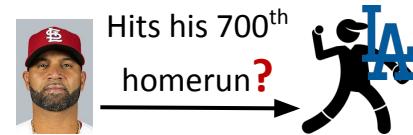
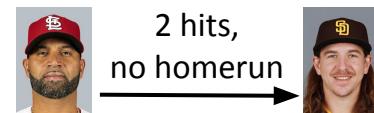
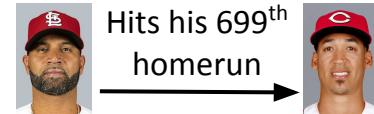
LA

t_3

STL

?

LA



Goal (Reminder)



Analyze relationships between players in baseball match

Discover new and interesting facts or patterns among player performance

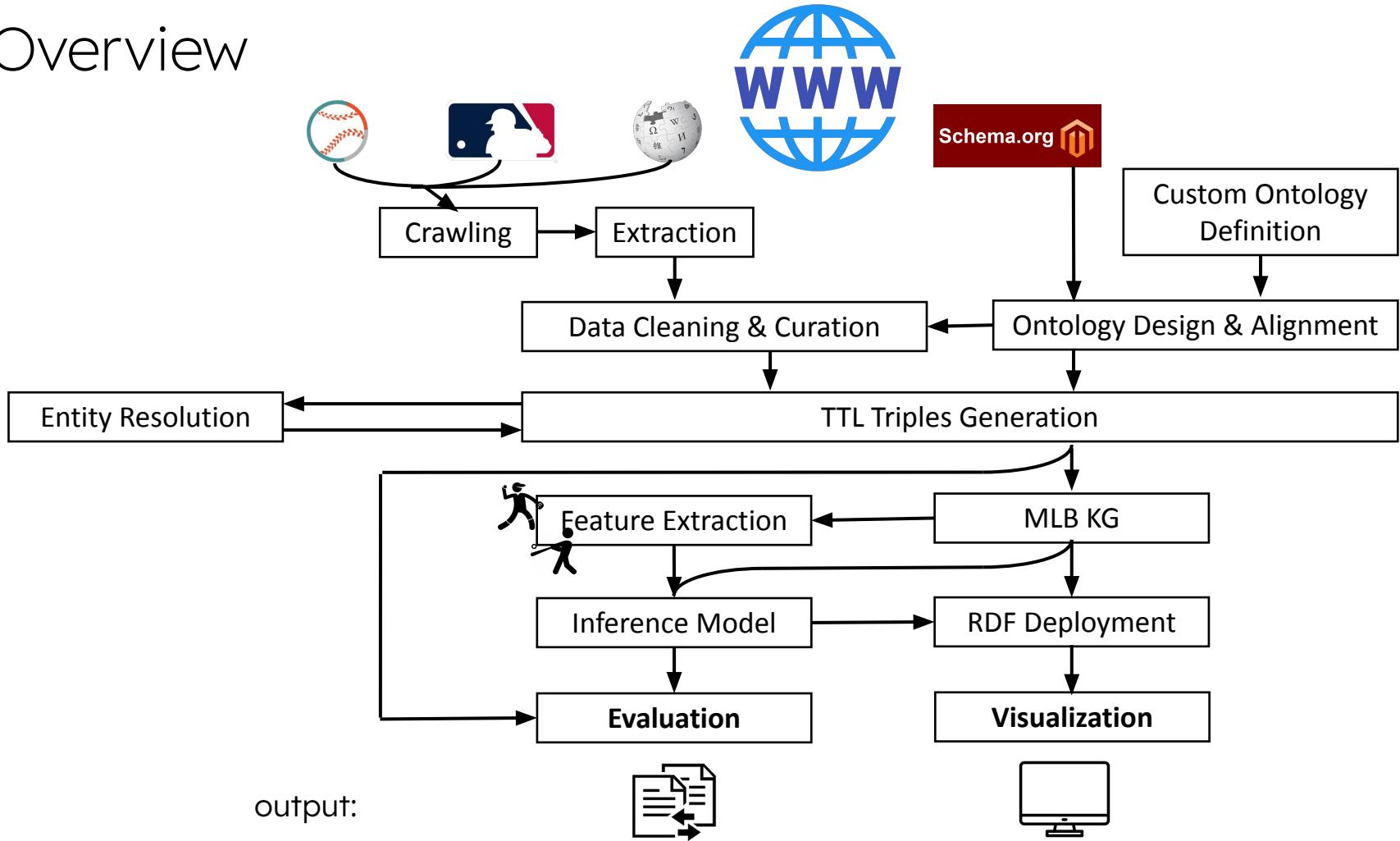
Expand the knowledge in baseball matches that was not aware of

Provide an easy-to-use user interface

Predict based on extracted information from the knowledge graph



Overview



Dataset

MLB.com

Season data for teams
(match schedule, player list, ...)

The screenshot shows the MLB.com website interface. It includes:

- Schedule:** A grid view of games for the St. Louis Cardinals in September 2022. One game is highlighted: Sep 16 vs. Cincinnati Reds (W, 6 - 5).
- Roster & Staff:** A table showing the active roster, depth chart, coaches, and transactions for the Cardinals.
- Albert Pujols Profile:** A detailed player profile for Albert Pujols, including his stats, career timeline, and a large image of him in a Cardinals uniform.
- Albert Pujols Player Card:** A summary card for Albert Pujols, showing his name, number (5), and a small image.
- Albert Pujols Batting Statistics:** A table showing his seasonal batting statistics from 2001 to 2004 for the St. Louis Cardinals.

Baseballsavant.mlb.com

Statistics for players
(pitch types, plate discipline, ...)

The screenshot shows the Baseballsavant.mlb.com website interface. It includes:

- Player Profile:** A detailed profile for Albert Pujols, showing his career statistics, a radar chart of his swing plane, and a video highlight.
- Salad Ball Profile:** A table showing Pujols' seasonal performance across various metrics like HR/FB, BB/K, and BABIP.
- Ran Value by Pitch Type:** A table showing the run value per 100 pitches for different pitch types in 2022.
- Advanced MLB Batting Statistics:** A table showing seasonal statistics for Albert Pujols from 2001 to 2004, including TPs, BB, K, and HR.

Wikipedia.org

Wikipedia infobox for players

The screenshot shows the Wikipedia infobox for Albert Pujols. It includes:

- Basic Information:** Name, birth date, death date, gender, and birth place.
- Biography:** Summary of his life, including his career highlights and achievements.
- Playing career:** Details of his professional baseball career, including teams, seasons, and statistics.
- Personal life:** Family information, including his wife and children.
- Awards and honors:** Recognition he received during his career.
- External links:** Links to other Wikipedia pages and external websites.



Crawling Challenges

Problem: Dynamic web pages and web links are difficult to parse for MLB.com

Solution:

Using Selenium

Selenium can simulate mouse clicks to click on the link of the next game to crawl all game schedules and dates.

The screenshot shows a baseball game page from MLB.com. At the top, there's a header with the date '11/05/2022', the game 'Phillies vs Astros', and the score 'Astros 7 - Phillies 5'. Below the header is a 'WORLD SERIES' section. The main content features a large scoreboard with the final score: 'Astros 1 - Phillies 4'. Below the scoreboard, there are detailed player stats for both teams. On the right side, there's a sidebar with a calendar and other game information. A developer tools window is overlaid on the page, showing the DOM structure. The 'Elements' tab is selected, and the 'gameday-index-scoreboard' element is highlighted. This element contains various game details like the date, time, and stadium. The developer tools also show the CSS and JS files being used for this specific component.

Using MLB official gamelog

Link composition is complex with many variables. The official gamelog obtains the game id, game type and game stage in the form of json files. Combined with the name of the home team opposing team, the corresponding web link can be completed.



Extraction Challenges

Problem: too many tables to store,
Image team name instead text

Solution:

Set each player as a directory and each table as a individual file to read. To have a more uniform format on subsequent reads. But it will slow down when reading (total over 7k tables)

Extract team name first in place image as needed

Stats													
		PA	AB	R	H	HR	RBI	SB	Avg	OBP	SLG	OPS	
2022		696	570	133	177	62	131	16	.311	.425	.686	1.111	
7 Seasons		3161	2638	535	748	220	497	40	.284	.394	.583	.977	

Statcast Outs Above Average													
Year	Team	Pos	OAA	In	Lateral toward 3B	Lateral toward 1B	Back	RHB	LHB	Attempts	Success Rate	Estimated Success Rate	Success Rate Added
2016	NY	RF	1	0	0	1	0	0	0	40	88%	84%	3%
2017	NY	RF	11	2	3	3	0	0	3	310	91%	87%	4%
2018	NY	ALL	1	3	0	-2	0	3	-2	204	89%	85%	0%
2018	NY	RF	1	3	0	-2	0	3	-2	203	89%	88%	0%
2018	NY	CF	0	0	0	0	0	0	0	1	100%	99%	1%
2019	NY	RF	8	0	2	4	3	5	3	212	91%	87%	4%
2020	NY	RF	1	1	-1	0	0	1	0	58	90%	88%	2%
2021	NY	ALL	0	2	-1	-1	0	0	0	282	87%	87%	0%
2021	NY	RF	0	2	-1	-2	1	0	0	237	86%	86%	0%
2021	NY	CF	0	0	-1	1	0	0	0	45	89%	89%	0%
2022	NY	ALL	3	-2	2	1	2	1	2	329	90%	89%	1%
2022	NY	RF	2	0	0	1	1	2	0	150	89%	87%	1%
2022	NY	CF	1	-2	2	0	1	-1	2	179	91%	90%	1%
Career	MLB	ALL	26	6	5	6	10	20	6	1435	89%	87%	2%

Statcast Detailed Outfield Outs Above Average													
Season	Outs Above Average	Back Left	Back	Back Right	Back All	In Left	In	In Right	In All	Expected Catch Percentage (%)	Actual Catch Percentage (%)	Catch Percentage Added (%)	
2016	1	0	0	1	1	0	0	0	1	84	88	4	
2017	11	1	3	-1	3	2	2	3	8	87	91	4	
2018	1	1	0	2	2	-1	3	-3	-1	88	89	1	
2019	8	0	3	1	4	2	0	3	5	87	91	4	
2020	1	0	0	0	0	-1	1	0	1	88	90	2	
2021	0	0	0	0	0	-1	2	-1	0	87	87	0	
2022	3	0	2	0	3	1	-2	1	0	89	90	1	



Data Cleaning Challenges

Problem: encoding issue and unstructured data

Solution:

Re-encoding to uniform encoding utf-8.

		Expected Home runs by Park																											
Year	HR	A	B	A ²	S	T	✓	✗	T _B	B	N	C	KC	D	F	S	A	A ²	P	D	G	A	M	H	W	P	✓	✗	C
2016	4	4	4	4	3	4	3	4	3	3	4	3	3	3	3	4	3	4	3	3	5	3	3	4	4	4	4	4	4
2017	56	52	62	54	53	55	54	55	56	51	58	56	46	41	53	58	54	59	46	55	63	48	51	56	56	54	56	48	50
2018	30	33	29	28	27	32	26	29	26	24	31	27	25	21	26	29	24	32	21	28	31	28	27	30	31	32	30	25	25
2019	28	28	24	24	26	25	25	28	25	19	26	24	18	19	18	27	19	28	17	23	25	24	24	23	25	28	26	22	23
2020	12	8	11	9	11	11	0	11	11	11	11	9	8	8	9	11	11	11	10	11	10	9	10	11	10	10	9	8	11
2021	39	34	37	30	36	41	33	36	36	30	34	36	28	27	31	39	37	33	31	32	34	32	35	36	33	34	37	32	33
2022	64	67	64	65	67	68	63	60	66	61	66	58	58	51	59	67	68	69	61	63	73	64	66	62	67	65	69	60	63
Player	233	226	231	214	223	236	204	223	223	199	230	213	186	170	199	235	216	236	189	215	241	208	216	222	226	227	231	199	209

[†]Note: This tells how many of this player's home runs would have been out of other stadiums. This accounts for different wall heights, distances and environmental effects. For these purposes, the 2020 Blue Jays home field was still Rogers Centre, not Sahlen Field.

+ View Complete Expected Home Runs Leaderboard

Year to Year Changes

Metric	2015	2016	+/-	2017	+/-	2018	+/-	2019	+/-	2020	+/-	2021	+/-	2022	+/-						
Exit Velocity	95.1			95	-0.1	1	94.7	-0.3	1	96	1.3	1	92.2	-3.8	1	95.8	3.6	1	95.9	0.1	
Launch Angle	20.6			15.8	-4.8	1	12.4	-3.4	1	11.2	-1.2	1	15.7	4.5	1	11.6	-4.1	1	15	3.4	1
Barrel %	11.6			25.7	14.1	1	16.2	-9.5	1	19.7	3.5	1	11.6	-8.1	1	17.6	6	1	26.5	8.9	1
Hard Hit %	58.1			54.7	-3.4	1	54.1	-0.6	1	58.4	4.3	1	40.6	-17.8	1	58.4	17.8	1	61.8	3.4	1
Sweet Spot %	32.6			38.2	5.6	1	35.7	-2.5	1	39.1	3.4	1	36.2	-2.9	1	38.5	2.3	1	39	0.5	1
xBA	.184			.287	.0103	1	.273	-0.014	1	.274	.0001	1	.265	-0.009	1	.300	.0043	1	.305	-0.003	1
BA	.179			.284	.0105	1	.278	-0.006	1	.272	-0.009	1	.257	-0.015	1	.287	.003	1	.311	.004	1
xwOBA	.281			.45	.169	1	.393	-0.057	1	.398	.0005	1	.378	-0.02	1	.418	.004	1	.463	.0045	1
wOBA	.267			.43	.163	1	.391	-0.039	1	.382	-0.009	1	.375	-0.007	1	.387	.012	1	.458	.0071	1
xSLG	.366			.668	.0302	1	.527	-0.141	1	.574	.047	1	.545	-0.029	1	.594	.049	1	.709	.0112	1
SLG	.345			.627	.0282	1	.528	-0.099	1	.54	.0012	1	.554	.014	1	.544	-0.01	1	.686	.0142	1
K%	44.2			30.7	-13.5	1	30.5	-0.2	1	31.5	1	1	28.1	-3.4	1	25	-3.1	1	25.1	0.1	1
BB%	9.5			18.7	9.2	1	15.3	-3.4	1	14.3	-1	1	8.8	-5.5	1	11.8	3	1	15.9	4.1	1
Swing %	45.6			41.1	-4.5	1	40.3	-0.8	1	42	1.7	1	42.9	0.9	1	42.5	-0.4	1	42.7	0.2	1
Whiff %	42.5			34.9	-7.6	1	36.2	1.3	1	36.7	0.5	1	33.5	-3.2	1	29.1	-4.4	1	29.8	0.7	1
In Zone %	46.3			42	-4.3	1	44.2	2.2	1	44.7	0.5	1	43.3	-1.4	1	44.4	1.1	1	45.8	1.4	1
Out of Zone %	53.7			58	4.3	1	55.8	-2.2	1	55.3	-0.5	1	56.7	1.4	1	55.6	-1.1	1	54.2	-1.4	1
Out of Zone Swing %	30.6			22.9	-7.7	1	20.9	-2	1	22.7	1.8	1	23.2	0.5	1	23.7	0.5	1	22.9	-0.8	1
First Pitch Swing %	27.4			33.8	6.4	1	29.4	-4.4	1	34.8	5.4	1	27.2	-7.6	1	32	4.8	1	31.5	-0.5	1
First Pitch Strike %	57.9			57.5	-0.4	1	55.8	-1.7	1	61.6	5.8	1	55.3	-6.3	1	59.6	4.3	1	58.8	-0.8	1

+ View Complete Year to Year Changes Leaderboard

Reconstruction the clawing logic

Reconstruction the clawing logic

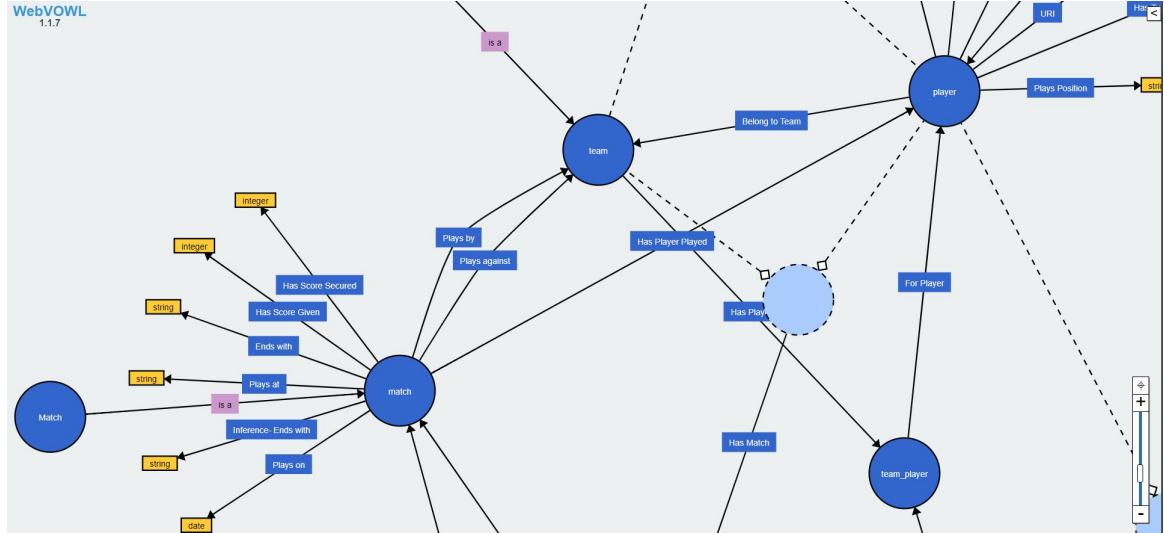
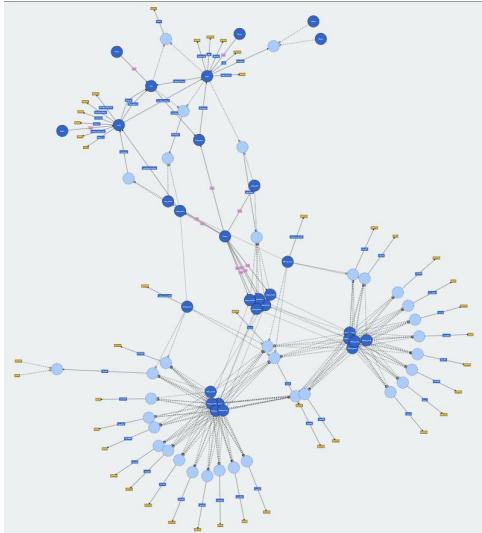
Ontology Challenges

Problem: lack of existing ontology capturing the required concepts

Solution:

Combine pre-defined ontologies with custom ones.

Utilize existing ontologies such as schema.org, xsd, and foaf.



Ontology Challenges (cont.)

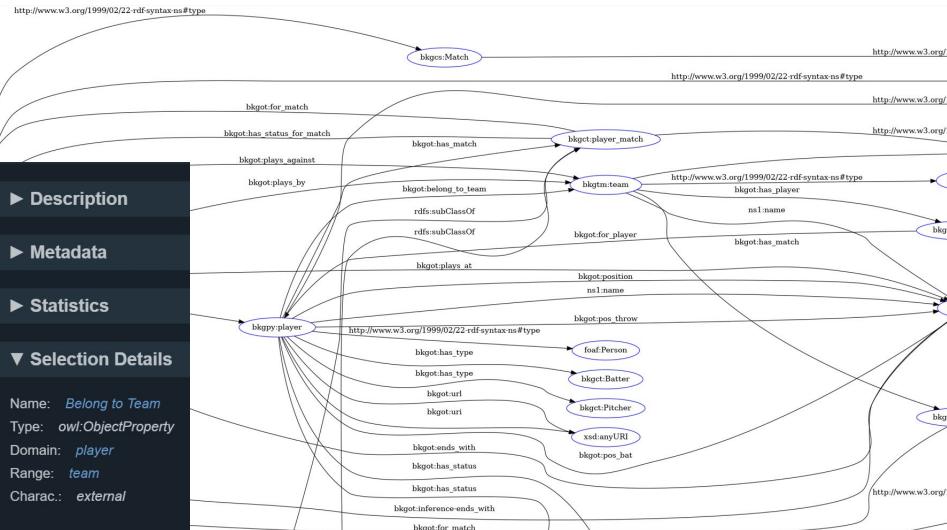
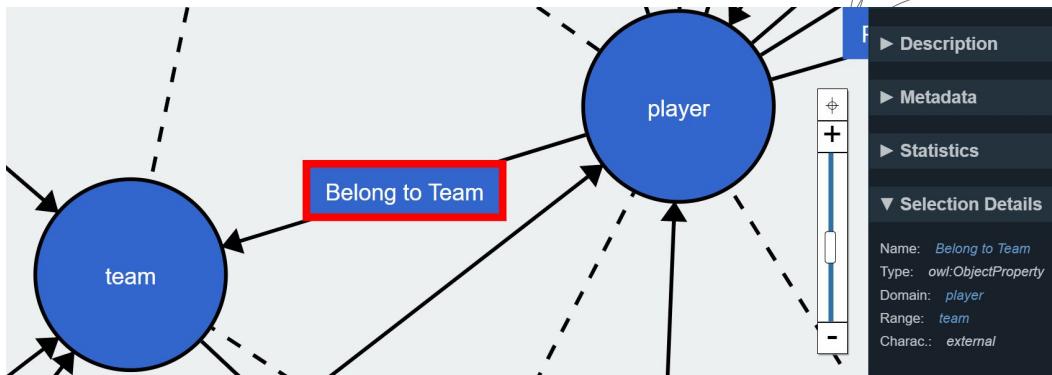
Ontology Design: Capture the ontological relationship between classes.

Define 65 classes with 89 object/ datatype properties specifying their domain/ range.

Class: player, team, match, ...

Relation: (team, plays against, team), (team, has, player), (player, plays, match), ...

Ends up with around 146,500 lines
more than 100 nodes



Entity Resolution Challenges

Problem: distinct representation of data across various sources

Player	IP	H	R	ER	SO	HR	ERA
Adam Wainwright	2.0	2	1	2	0	0	4.50
Génésis Cabrera	2.0	2	1	2	0	0	4.50

Player	IP	H	R	ER	SO	HR	ERA
Will Smith	2.0	2	1	2	0	0	4.50
Chris Taylor	2.0	2	1	2	0	0	4.50

Entity Resolution Challenges (cont.)

Problem: distinct representation of data across various sources

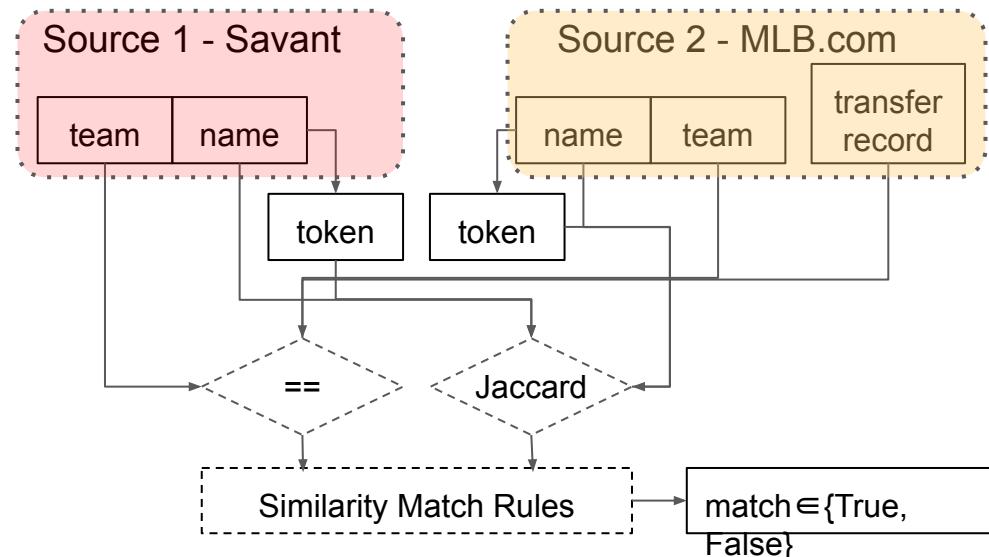
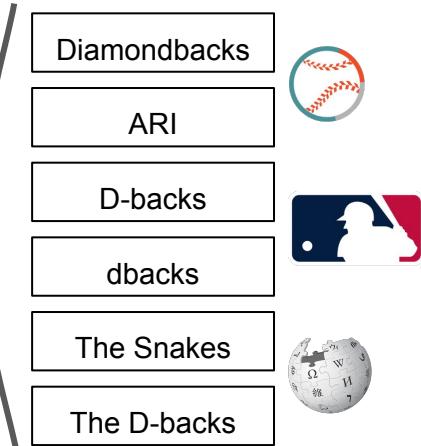
Solution:

Team name - Implement a rule-based system for direct replacement.

Player name - Use attributes from each source for entity resolution.



Arizona Diamondbacks

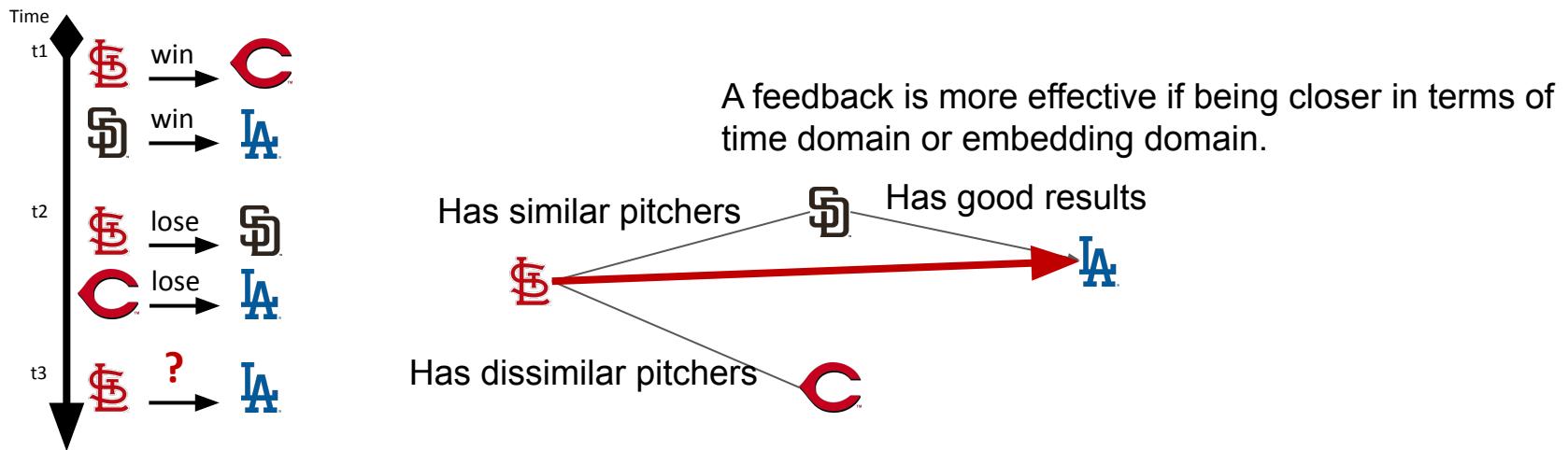


Match-Result Prediction

Problem: predict results for matches in the future (extrapolation) - Win/Lose
 Solution:

Incorporating recent performance with feedbacks from similar teams

1. Construct temporal embeddings for each team.
2. Retain feedbacks based on similarities to other teams.
3. Access previous match results for the both targeted team and opponent.



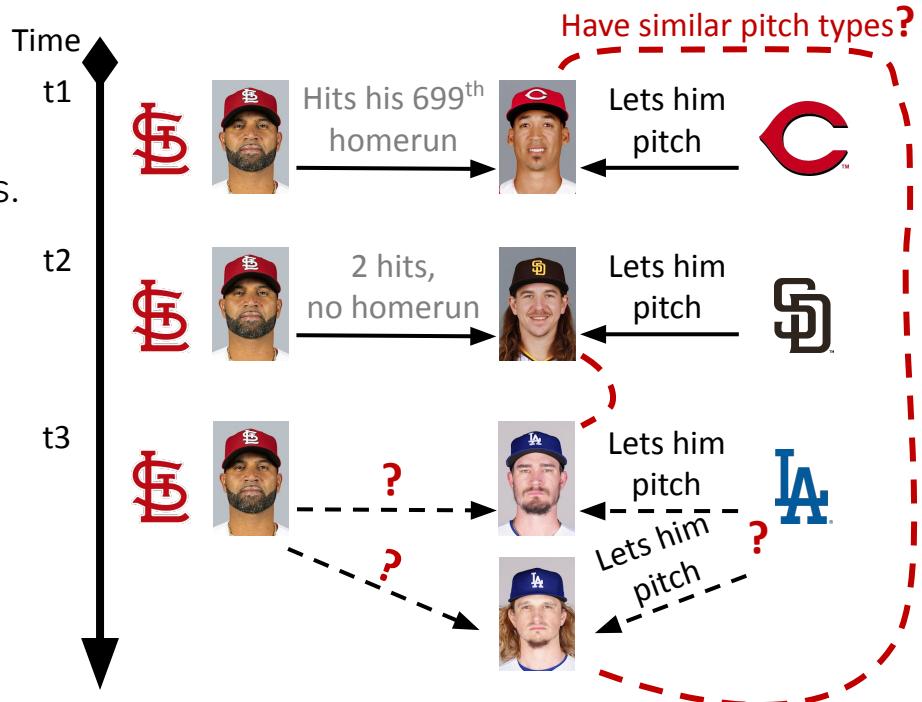
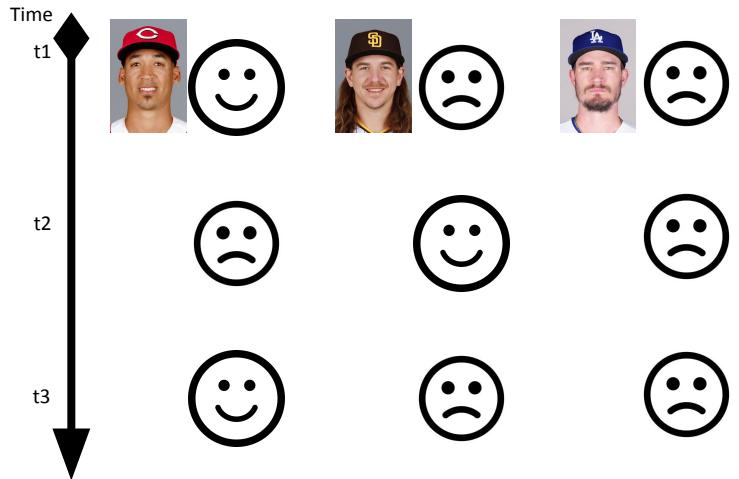
Player-Performance Prediction

Problem: predict performance of players in the future (extrapolation) - RBI/ER

Solution:

Incorporating recent performance
with feedbacks from similar players

Players with similar characteristics.
(e.g., in pitching style, hitting patterns)





User Interface

Search for query

Select a sample query

Player belongs to which team?

Copy sample query from the text box

```
SELECT ?person ?object_name
WHERE {
    ?person <http://dsci558.org/ontology/belong_to_team> ?object.
    ?object <http://schema.org/name> ?object_name
}
```

Enter a query

MLB KG

This project is about to build a knowledge graph for MLB.

This web was created by *Zihao H., Fandel Lin.*

Select a Page

- [Homepage](#)
- [Homepage](#)
- [Search with Sparql](#)
- [Search with selection](#)
- [Predict](#)
- [Player Information](#)
- [Schedule](#)

Selection menu

Sample query(can copy and paste to box)

And

Query search(with your own query)

Enter a query

```
SELECT ?person ?object_name
WHERE {
    ?person <http://dsci558.org/ontology/belong_to_team> ?object.
    ?object <http://schema.org/name> ?object_name
}
```

this may take a while...

```
(rdflib.term.URIRef('http://dsci558.org/player/669004'), rdflib.term.Literal('Kansas City
Royals', datatype=rdflib.term.URIRef('http://www.w3.org/2001/XMLSchema#string')))

(rdflib.term.URIRef('http://dsci558.org/player/677951'), rdflib.term.Literal('Kansas City
Royals', datatype=rdflib.term.URIRef('http://www.w3.org/2001/XMLSchema#string')))

(rdflib.term.URIRef('http://dsci558.org/player/643217'), rdflib.term.Literal('Kansas City
Royals', datatype=rdflib.term.URIRef('http://www.w3.org/2001/XMLSchema#string')))
```

User Interface (cont.)

Search by class with query

And output with its graph
and results

And some visualization



Please select the class you want to search for

Select a class

player

match

team

Please select the class you want to search for

Select a class

player

Search for a player with predicates

Select a predicate

belong_to_team

belong_to_team

has_match

pos_bat

pos_throw

position

Diagram illustrating the search for a player with predicates. A query is made for 'belong_to_team'. The system identifies the 'belong_to_team' predicate and shows the corresponding triples in the graph:

```
graph TD; ?object[?object] -- "ont:belong_to_team" --> schema_name[schema.name]; schema_name -- "schema.name" --> person[person];
```

Cole Irvin plays at Oakland Athletics

Seth Brown plays at Oakland Athletics

Tony Kemp plays at Oakland Athletics

Elvis Andrus plays at Oakland Athletics

Please select the class you want to search for

Select a class

team

Search for a player with predicates

Select a predicate

has_player

```
(rdflib.term.URIRef('http://dsci558.org/team/Kansas_City_Royals'),  
rdflib.term.URIRef('http://dsci558.org/cluster/Kansas_City_Royals_player'))
```

```
(rdflib.term.URIRef('http://dsci558.org/team/Miami_Marlins'),  
rdflib.term.URIRef('http://dsci558.org/cluster/Miami_Marlins_player'))
```



User Interface (cont.)

Doing Prediction with query
to see its score

Select a prediction query

Prediction - Player Performance - How player (a hitter) performs in a match

Prediction - Player Performance - How player (a pitcher) performs in a match

```
?match a <http://dsci558.org/class/Match>.

?match <http://dsci558.org/ontology/has_player_played> ?pl
?match <http://dsci558.org/ontology/plays_at> ?location.
?match <http://dsci558.org/ontology/plays_on> ?date.
?match <http://dsci558.org/ontology/plays_by> ?team1.
?match <http://dsci558.org/ontology/plays_against> ?team2.

?player1 <http://dsci558.org/ontology/has_match> ?player_m
?player_match <http://dsci558.org/ontology/has_status> ?pl
?player_status_in_match <http://dsci558.org/ontology/has_s
?player_status_in_match <http://dsci558.org/ontology/infer
?player_status_in_match <http://dsci558.org/ontology/has_E

?team1 <http://dsci558.org/ontology/has_player> ?player1_c
?player1_cluster <http://dsci558.org/ontology/for_player>

?player1 <http://schema.org/name> ?player1_name.
?team1 <http://schema.org/name> ?team1_name.
?team2 <http://schema.org/name> ?team2_name.
}

LIMIT 10
```

Shane McClanahan <(from)- Tampa Bay Rays -(has ER in match) 5.0 <-(inference)-> 1.0 -(plays against)>
Houston Astros -(plays on)-> 2022-09-20 -(plays at)-> HOME

Shane McClanahan <(from)- Tampa Bay Rays -(has ER in match) 4.0 <-(inference)-> 1.0 -(plays against)>
Toronto Blue Jays -(plays on)-> 2022-09-25 -(plays at)-> HOME

Shane McClanahan <(from)- Tampa Bay Rays -(has ER in match) 1.0 <-(inference)-> 2.0 -(plays against)>
New York Yankees -(plays on)-> 2022-05-29 -(plays at)-> HOME

Sandy Alcantara <(from)- Miami Marlins -(has ER in match) 4.0 <-(inference)-> 2.0 -(plays against)-> New
York Mets -(plays on)-> 2022-06-24 -(plays at)-> HOME

Search stats of players by name

Search for a player

Enter a player name

jos

Abreu_Jose

Stats

	Unnamed: 0	PA	AB	R	H	HR	RBI	SB	Avg	OBP	SLG	OPS
0	2022	679	601	85	183	15	75	0	0.3040	0.3780	0.4460	0.8240
1	9 Seasons	5506	4954	697	1445	243	863	11	0.2920	0.3540	0.5060	0.8600

Altuve_Jose

Stats

	Unnamed: 0	PA	AB	R	H	HR	RBI	SB	Avg	OBP	SLG	OPS
0	2022	604	527	103	158	28	57	18	0.3000	0.3870	0.5330	0.9200
1	12 Seasons	6950	6305	986	1935	192	696	279	0.3070	0.3620	0.4680	0.8300

Bell_Josh

Stats

	Unnamed: 0	PA	AB	R	H	HR	RBI	SB	Avg	OBP	SLG	OPS
--	------------	----	----	---	---	----	-----	----	-----	-----	-----	-----



Validation

Validate correctness of the KG in each curation step

Data Extraction & Data Cleaning

Generate error dump file with known patterns, inducing problems by type for further amendment

Data Curation & TTL Triple Generation

Use SPARQL queries to examine triples and distinguish problems by patterns

Entity resolution

Manual labeling for validation

Ontology correctness

Online RDF validation tools (e.g., RDF validator)
Visualization tools (e.g., webowl)

KG quality

Manual check for the quality dimensions (intrinsic/ contextual/ representational/ accessibility)
1. Data for the latest MLB season is extracted with sufficient breadth and depth in the domain
2. Additional ontologies are defined to be interpreted by human & machines easily without ambiguity
3. Free of error or bias with a consistent representation in URL and interlinks to trustworthy resources

Extrapolated prediction

F1 score with sliding window (for match-result prediction)
MAE with sliding window (for player-performance prediction)

Evaluation

KG Usefulness	Ontology Correctness	KG Quality																										
<p>Support user-defined or pre-defined sets of SPARQL queries</p> <p>⊕ Search for query</p> <p>Select a sample query Player belongs to which team?</p> <p>Copy sample query from the text box</p> <pre>SELECT ?person ?object_name WHERE { ?person <http://dsc1558.org/ontology/belong_to_team> ?object. ?object <http://schema.org/name> }</pre> <p>Enter a query SELECT ?person ?object_name WHERE { ?person <http://dsc1558.org/ontology/belong_to_team> ?object. ?object <http://schema.org/name> ?object_name }</p> <p>this may take a while...</p>	<p>Vasiaulize designed ontology</p>	<p>Statistics for entity labeling in KG</p> <hr/> <p>Entity resolution for validation set</p> <hr/> <table border="1"> <thead> <tr> <th data-bbox="1230 354 1614 379">Approach (ablation)</th> <th data-bbox="1653 354 1806 379">F1 Score</th> </tr> </thead> <tbody> <tr> <td data-bbox="1230 386 1345 411">ours</td> <td data-bbox="1691 386 1806 411">1.0000</td> </tr> <tr> <td data-bbox="1230 429 1614 454">- team consistency & transfer</td> <td data-bbox="1691 429 1806 454">0.9597</td> </tr> <tr> <td data-bbox="1230 473 1461 497">- string similarity</td> <td data-bbox="1691 473 1806 497">0.7299</td> </tr> </tbody> </table>	Approach (ablation)	F1 Score	ours	1.0000	- team consistency & transfer	0.9597	- string similarity	0.7299																		
Approach (ablation)	F1 Score																											
ours	1.0000																											
- team consistency & transfer	0.9597																											
- string similarity	0.7299																											
<h2>Extrapolated Prediction</h2>																												
<p>F1 score and MAE with sliding window for match-result and player-performance prediction (a sliding window enables prediction per month based on data from the previous month)</p>	<hr/> <table border="1"> <thead> <tr> <th data-bbox="328 800 846 825">Performance for match-result prediction</th> <th data-bbox="1038 800 1633 825">Performance for player-performance prediction</th> </tr> </thead> <tbody> <tr> <td data-bbox="328 843 846 868">Approach (ablation)</td> <td data-bbox="1038 843 1633 868">Approach (ablation)</td> </tr> <tr> <td data-bbox="328 875 846 900">ours</td> <td data-bbox="1038 875 1633 900">ours</td> </tr> <tr> <td data-bbox="328 908 846 933">- team embedding</td> <td data-bbox="1038 908 1633 933">- recent performance decay</td> </tr> <tr> <td data-bbox="328 940 846 965">- recent performance</td> <td data-bbox="1038 940 1633 965">- player embedding</td> </tr> <tr> <td data-bbox="328 973 846 997">- opponent's performance</td> <td data-bbox="1038 973 1633 997">- recent performance</td> </tr> <tr> <td data-bbox="328 1005 846 1030">baseline (random value)</td> <td data-bbox="1038 1005 1633 1030">baseline (average value)</td> </tr> </tbody> </table>	Performance for match-result prediction	Performance for player-performance prediction	Approach (ablation)	Approach (ablation)	ours	ours	- team embedding	- recent performance decay	- recent performance	- player embedding	- opponent's performance	- recent performance	baseline (random value)	baseline (average value)	<hr/> <table border="1"> <thead> <tr> <th data-bbox="328 843 846 868">F1 Score</th> <th data-bbox="1038 843 1633 868">MAE</th> </tr> </thead> <tbody> <tr> <td data-bbox="328 875 846 900">0.6639</td> <td data-bbox="1038 875 1633 900">0.6213</td> </tr> <tr> <td data-bbox="328 908 846 933">0.6597</td> <td data-bbox="1038 908 1633 933">0.6700</td> </tr> <tr> <td data-bbox="328 940 846 965">0.5770</td> <td data-bbox="1038 940 1633 965">0.7326</td> </tr> <tr> <td data-bbox="328 973 846 997">0.5413</td> <td data-bbox="1038 973 1633 997">0.7441</td> </tr> <tr> <td data-bbox="328 1005 846 1030">0.4951</td> <td data-bbox="1038 1005 1633 1030">0.8049</td> </tr> </tbody> </table>	F1 Score	MAE	0.6639	0.6213	0.6597	0.6700	0.5770	0.7326	0.5413	0.7441	0.4951	0.8049
Performance for match-result prediction	Performance for player-performance prediction																											
Approach (ablation)	Approach (ablation)																											
ours	ours																											
- team embedding	- recent performance decay																											
- recent performance	- player embedding																											
- opponent's performance	- recent performance																											
baseline (random value)	baseline (average value)																											
F1 Score	MAE																											
0.6639	0.6213																											
0.6597	0.6700																											
0.5770	0.7326																											
0.5413	0.7441																											
0.4951	0.8049																											



Conclusion

Apply KG for capturing complex relationships in baseball matches

Support up-to-date and free-of-error data with consistent representation

Construct ontology that organizes relationships with trustworthy interlinks

Provide an easy-to-use user interface with multiple querying approach

Exploit data with sparse relationships in KG for extrapolated prediction