Automatic End-to-End Data Integration using Large Language Models

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

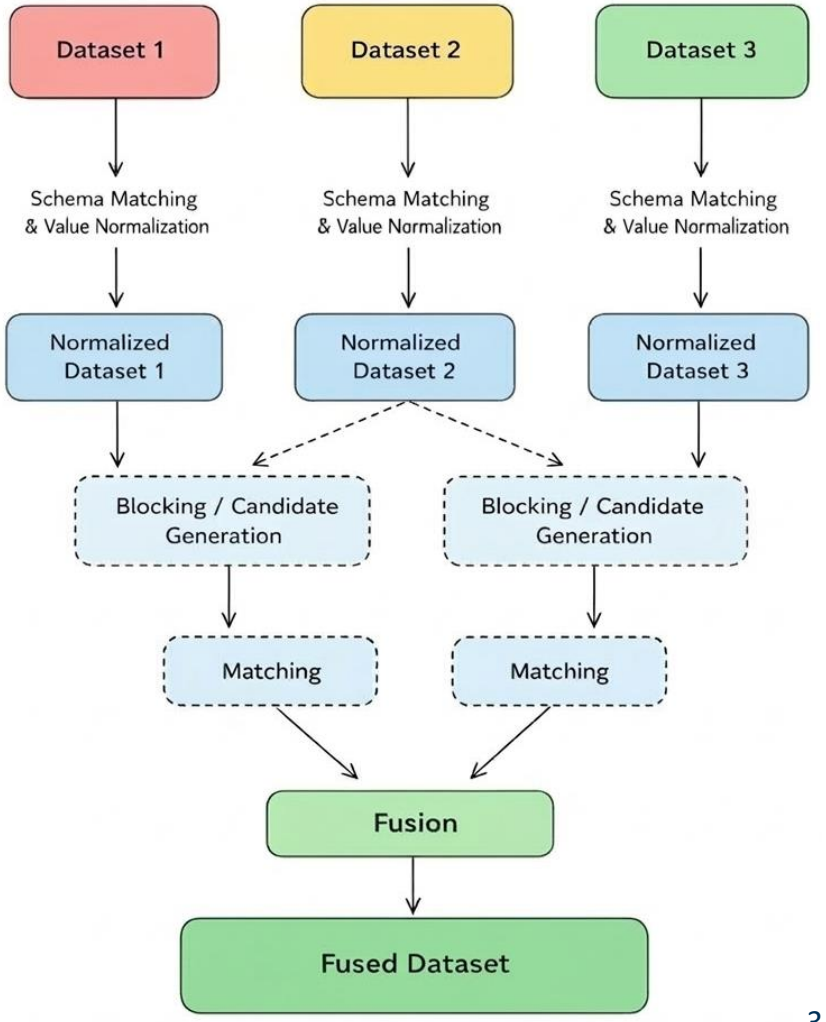
Aaron Steiner, Christian Bizer

Motivation

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- **Data integration:** combines data from multiple heterogeneous sources into a unified dataset
- Each integration step traditionally requires substantial manual effort from data engineers

→ **Research Question: Can LLMs replace human involvement in configuring end-to-end data integration pipelines?**



Human versus Automatic Data Integration Pipeline

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

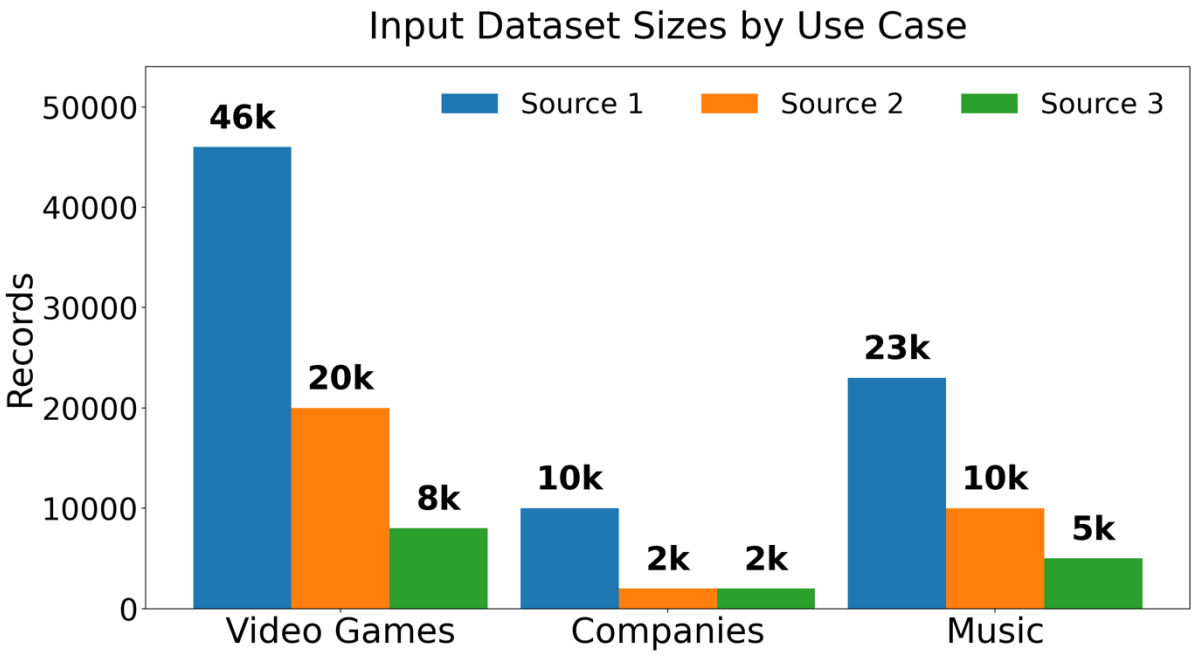
- **For each use case, we compare a human silver-standard pipeline with a fully automatic LLM-based pipeline**
- Both pipelines use the same core library: **PyDI**
- The automatic system configures the pipeline fully autonomously
- We then compare end-to-end results on three case studies

→ **Difference = who creates the configuration, trainings and validation data**

Case Studies

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- **Three case studies, each integrating 3 heterogeneous datasets:**
- **Video Games:** Challenges are matching games across platforms, multiple taxonomies (platform etc.)
- **Companies:** Challenges are name variations (abbreviations, legal suffixes), financial scale normalization
- **Music:** Challenges are duration format normalization, varying country naming conventions



Example: Schema Heterogeneity

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- Source attributes differ in naming, coverage, and semantics
- Some attributes have generic names like Attribute_x and can only be matched using the values
- Target schemas are provided by the usecases

Case Study	Source	Rows	Attrs	Density	Attributes
Music	Discogs	22,627	8	98.4%	rec_uid, title_str, performer, pub_dt, origin_loc, imprint, category, tracks_track-name
	Last.fm	9,865	5	78.7%	item_code, album_title, band, tracks_track-name, album_length_min
	MusicBrainz	4,763	7	83.3%	Attribute_1, Attribute_2, Attribute_3, Attribute_4, Attribute_5, Attribute_6, Attribute_7
	Target Schema	–	8	–	id, name, artist, release-date, release-country, label, genre, tracks, duration

Schema Matching

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- **Schema matching = map source columns to target attributes**
- Humans: inspect column names and sample values, then write mappings manually
- Automatic pipeline: uses column names, value summaries, sample rows
- Both use a target schema

→ **Same result: 67/67 correspondences, F1 = 1.00**

Benchmark	Human	LLM	Label	Instance
Games	**1.00**	**1.00**	0.23	0.00
Companies	**1.00**	**1.00**	0.35	0.32
Music	**1.00**	**1.00**	0.38	0.24
Average	**1.00**	**1.00**	0.32	0.19

Value Normalization

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

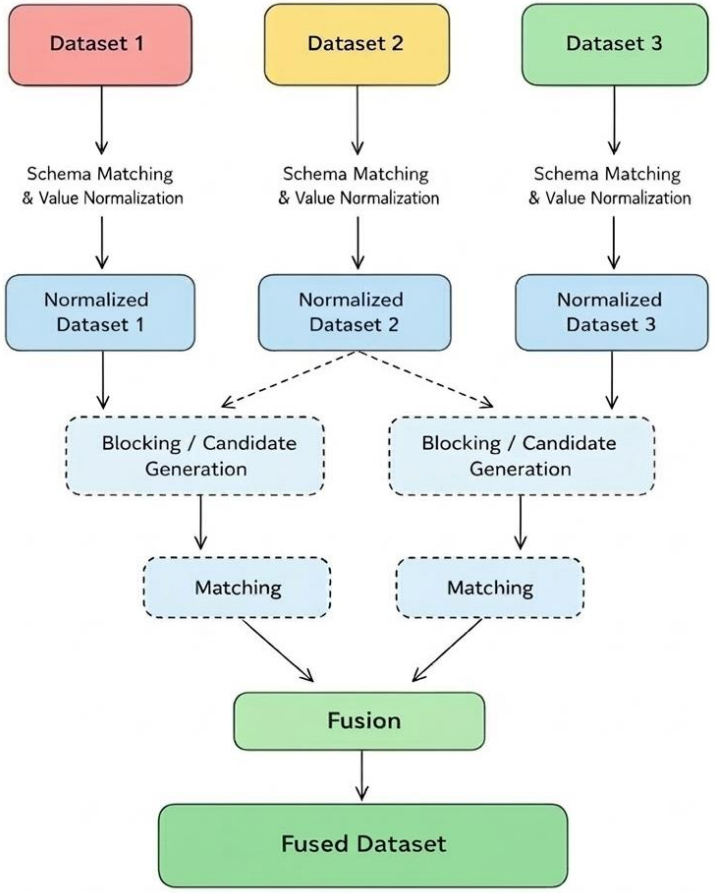
- Humans: custom normalization functions and simple value mappings
- **Automatic pipeline:**
- Automatic column profiling selects code normalizers
- Maps values to target taxonomies

→ **Automatic pipeline normalizes more categorical and taxonomy values than humans**

Games Examples

Type	Raw value	Human	LLM
Taxonomy	PS4	PlayStation 4	PlayStation 4
Taxonomy	NES	NES	Nintendo Entertain-ment System
Date	11/15/2013	2013-11-15	2013-11-15

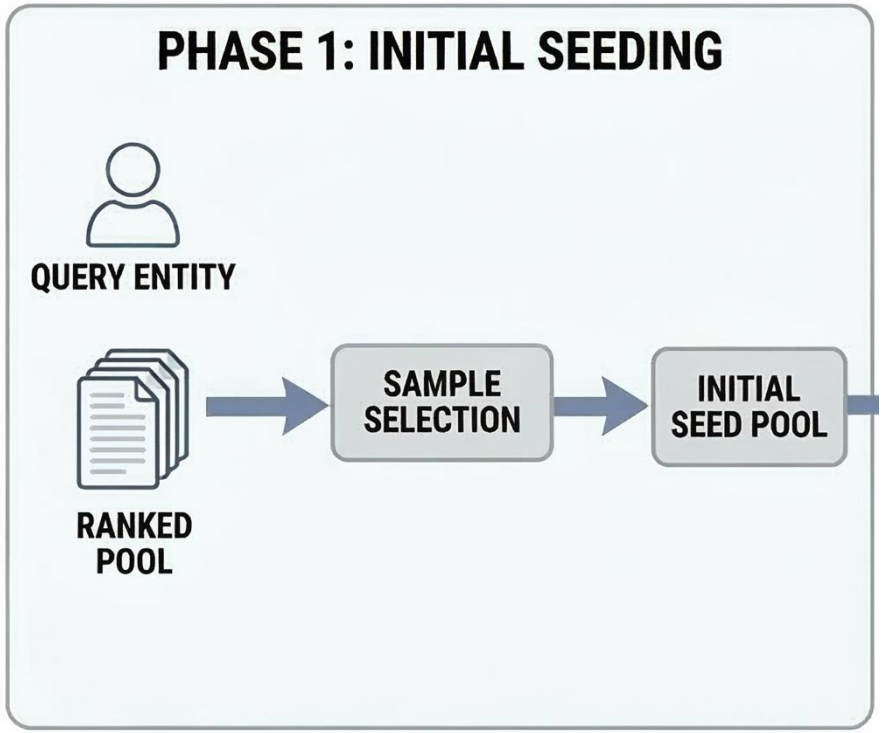
Recap - Where Are We?



Entity Matching: LLM Training Data

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

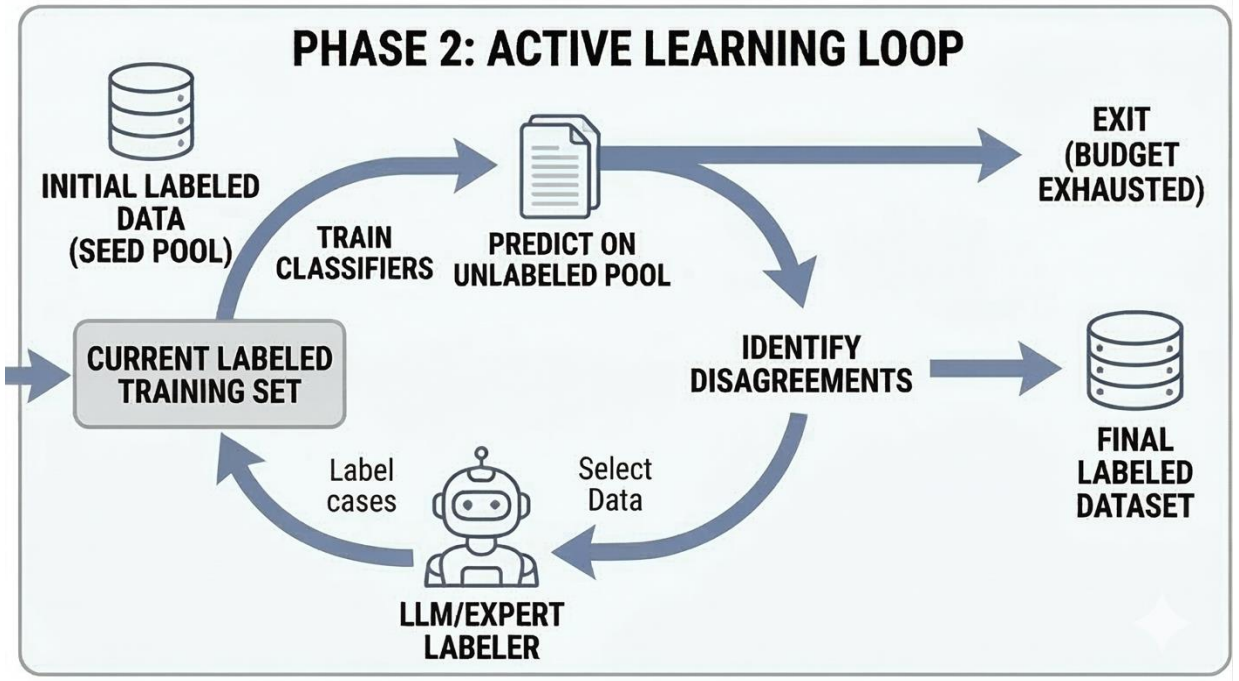
- Humans hand-annotate pairs for training classifiers
- **Active learning workflow with LLM as labeler**
- Step 1 - Seeding label pairs
  - Yields ~100 (30/70) labeled seed pairs per dataset pair



Entity Matching: LLM Training Data

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- **Active learning workflow with LLM as labeler**
- **Step 2 - Active learning loop:**
  - Train 5 classifiers, select 100 pairs with highest prediction disagreement
  - GPT-5.2 labels disputed pairs → retrain → repeat until budget exhausted
  - Training set augmented with 20% random samples to counter selection bias



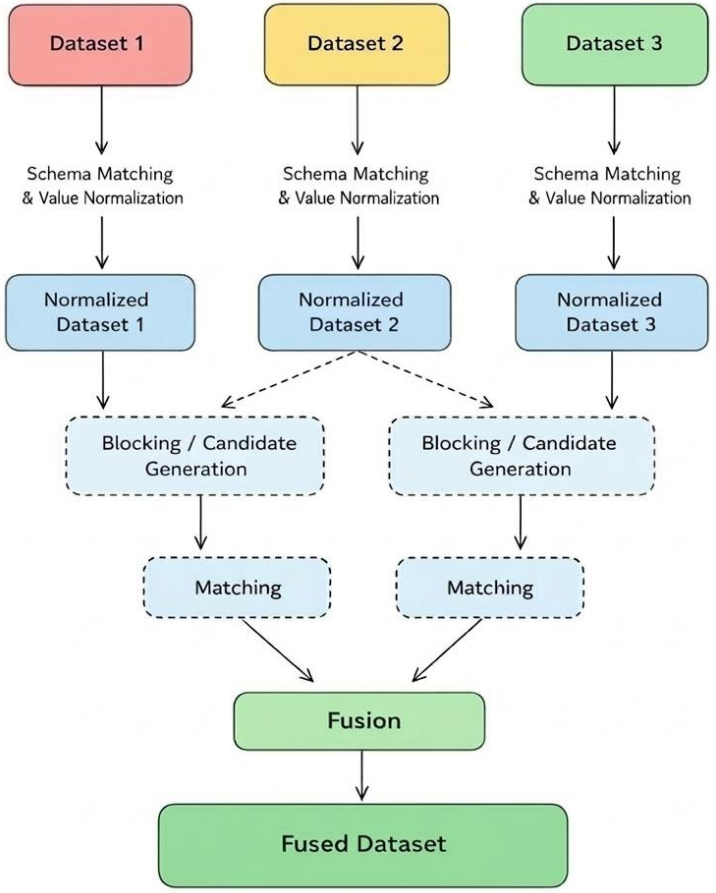
Entity Matching: Results

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

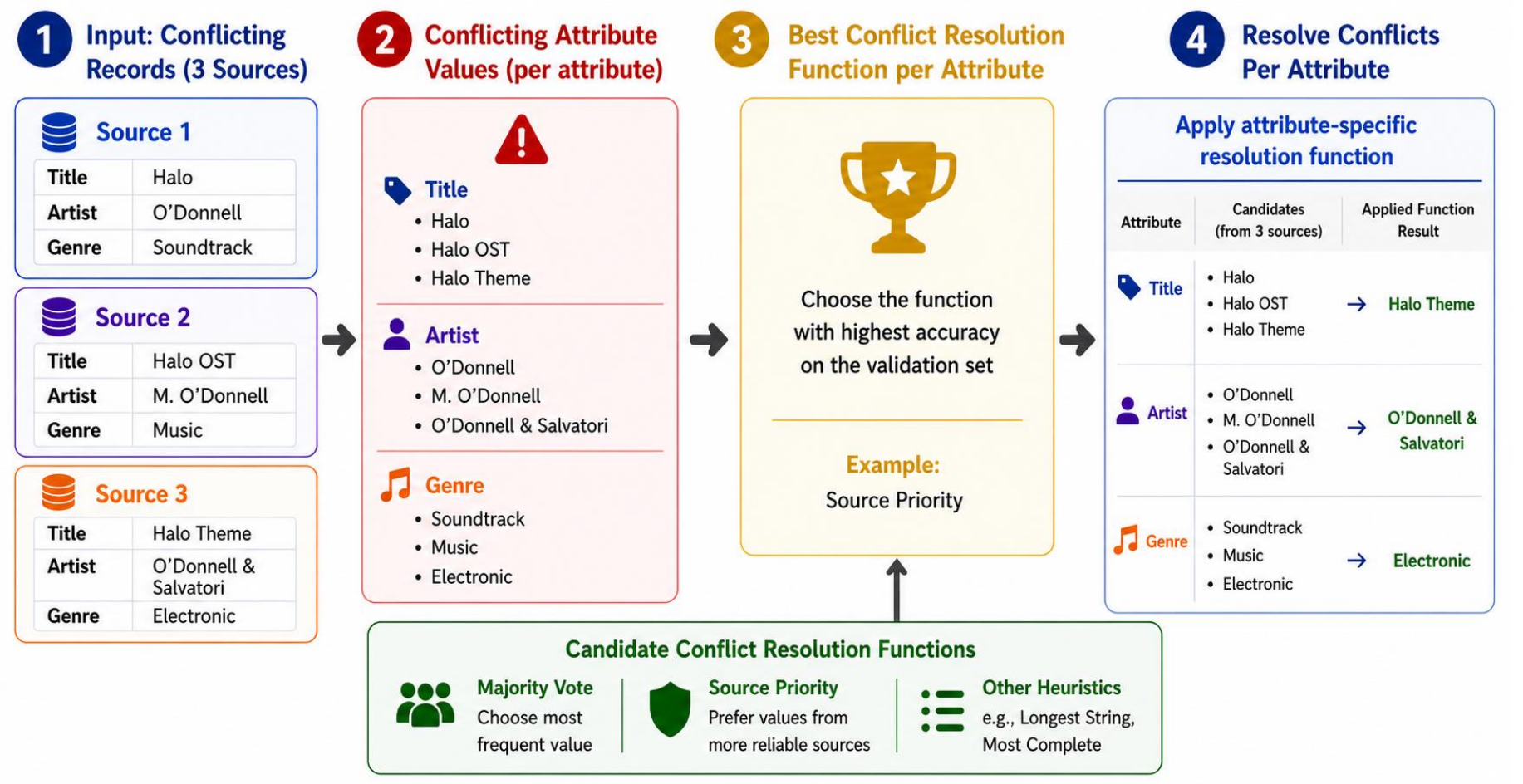
Dataset Pair	Human Config	Human Labels	LLM Labels
Games: DBp - Meta	**0.93**	0.83	0.85
Games: DBp - Sales	0.93	0.84	**0.98**
Comp: DBp - Forbes	0.86	**0.95**	0.94
Comp: Forbes - FC	0.87	**0.90**	**0.90**
Music: Disc - MB	0.80	**0.99**	**0.99**
Music: Last - MB	0.98	**0.99**	0.97
Average	0.89	0.92	**0.94**

→ **LLM Labels achieve 0.94 avg F1 (best)**

Recap - Where Are We?



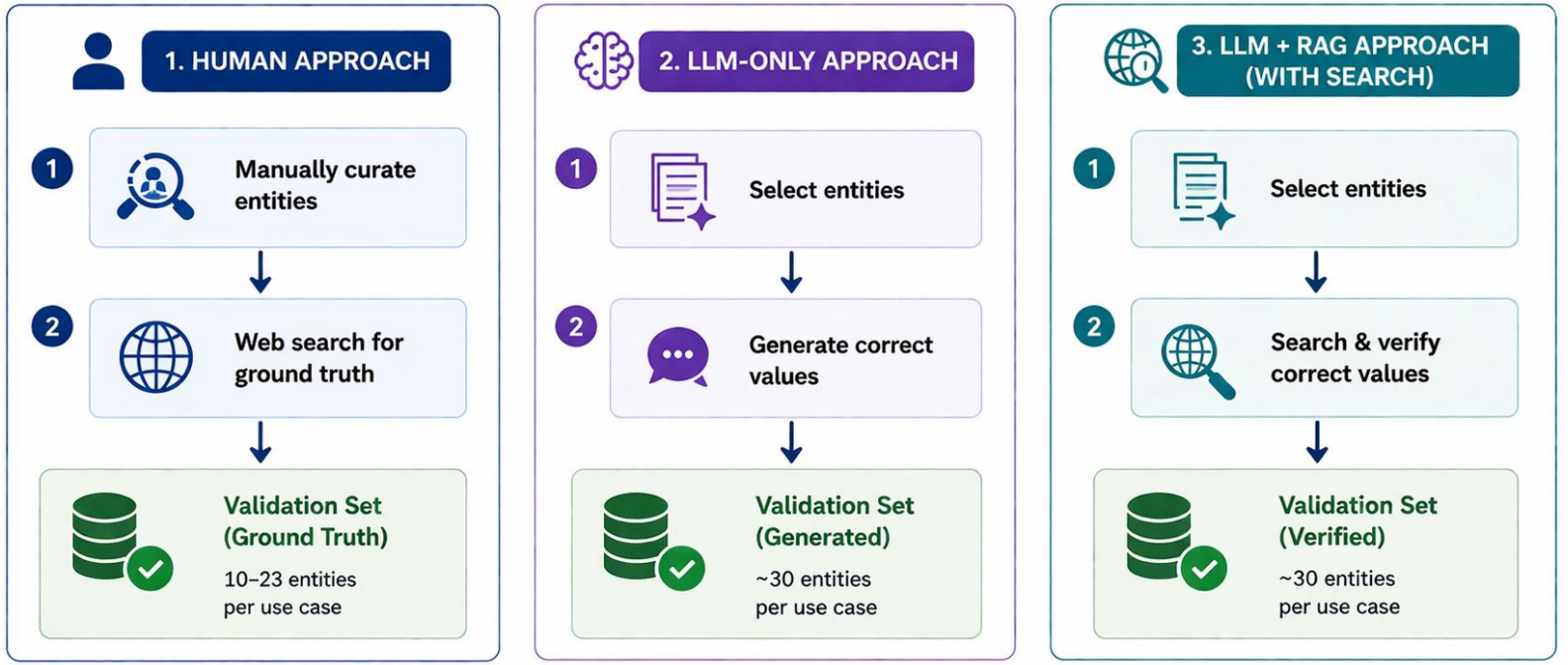
Data Fusion


1 Input: Conflicting Records (3 Sources)
Source 1
Title Halo
Artist O'Donnell
Genre Soundtrack
Source 2
Title Halo OST
Artist M. O'Donnell
Genre Music
Source 3
Title Halo Theme
Artist O'Donnell & Salvatori
Genre Electronic
2 Conflicting Attribute Values (per attribute)
Title
• Halo
• Halo OST
• Halo Theme
Artist
• O'Donnell
• M. O'Donnell
• O'Donnell & Salvatori
Genre
• Soundtrack
• Music
• Electronic
3 Best Conflict Resolution Function per Attribute
Choose the function with highest accuracy on the validation set
Example:
Source Priority
4 Resolve Conflicts Per Attribute
Apply attribute-specific resolution function
Attribute
Candidates (from 3 sources)
Applied Function Result
Title
• Halo
• Halo OST
• Halo Theme
→ Halo Theme
Artist
• O'Donnell
• M. O'Donnell
• O'Donnell & Salvatori
→ O'Donnell & Salvatori
Genre
• Soundtrack
• Music
• Electronic
→ Electronic
Candidate Conflict Resolution Functions
Majority Vote
Choose most frequent value
Source Priority
Prefer values from more reliable sources
Other Heuristics
e.g., Longest String, Most Complete

Fusion Validation Data Creation

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

Goal: generate validation sets to select the best conflict resolution methods functions per attribute



Data Fusion

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- **Data fusion = choose the correct value when sources disagree**
- Humans & automatic pipeline use the validation set to iteratively find the best fusion heuristic

→ **Best result still comes from human validation; the gap is mainly Companies because values are time-sensitive**

Benchmark	Human Config	Human Val.	LLM Val.	LLM + RAG
Games	0.83	**0.87**	**0.87**	**0.87**
Companies	0.80	**0.86**	0.71	0.74
Music	**0.77**	0.72	0.71	0.71
Average	**0.80**	**0.82**	0.76	0.77

Runtime and Cost Analysis

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

LLM API costs per use case; Config: ~1.9h (LLM) vs. ≥19h (human)

Pipeline Step	Games	Companies	Music	Total
Schema Matching	$0.02	$0.02	$0.02	$0.06
Normalization	$0.13	$0.36	$0.17	$0.66
Training Set Gen.	$1.33	$1.96	$1.66	$4.95
Fusion Val. (LLM)	$0.22	$0.15	$0.23	$0.60
Fusion Val. (RAG)	$7.46	$6.07	$7.45	$20.98
Total per Case	**$9.16**	**$8.56**	**$9.53**	**$27.25**

→ **10× reduction in configuration effort at ~$9 per use case**

Conclusion

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- **LLM pipeline produced outcomes close to or surpassing humans**
  - Strength of LLM pipeline: schema matching, value normalization, and entity-matching labels
  - Main weakness: data fusion on time-sensitive attributes
- **10x runtime reduction: ~1.9h versus ≥19 human hours per use case**
- **9 USD per use case**

Future Work

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

- **Agent-based refinement:** agentic architecture feeding downstream results back to earlier steps
- **Human-agent collaboration:** pipeline outputs as starting point for collaborative refinement
- **Dynamic code generation:** LLM-based coding agents for transformation functions

Thank you for listening

UNIVERSITY OF MANNHEIM
School of Business Informatics and Mathematics

Paper

Github

LinkedIn