Foundational Models for Tabular Data Understanding and Matching-related Tasks

Federico Belotti (federico.belotti1@unimib.it)

INSID&S Lab
Department of Informatics, Systems and Communication
Università degli Studi di Milano-Bicocca

Who am I?

Research Fellow @ UNIMIB under the supervision of Prof. Matteo Palmonari

Previously:

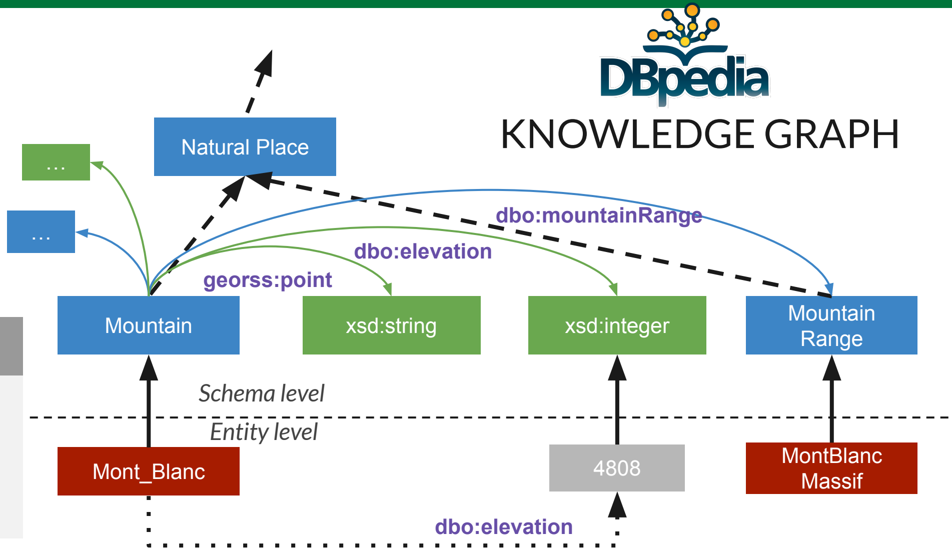
- Oct. 2020: Master thesis on unsupervised lexical semantic change detection (advisor: Prof. Matteo Palmonari)
- Nov. 2020 - Dec. 2023: Data Scientist @ Orobix S.r.l.
  - Computer Vision Team: Self-Supervised Learning, Domain Adaptation, Classification, Segmentation
  - Reinforcement Learning: PL of a project with Milestone (MotoGP), creator and maintainer of SheepRL

Outline

1. Problem definition
2. Task and models summary
3. TDU Transformer-based methods
   a. TURL
   b. Unicorn
   c. TableLlama
4. Open questions

Tabular Data Annotation

Name	Coordinates	Height	Range
Le Mont Blanc	45°49′57″N 06°51′52″E	4808	M. Blanc massif
Hohtälli	45°98′96″N 07°80′25″E	3275	Pennine Alps
Monte Cervino	45°58′35″N 07°39′31″E	4478	Pennine Alps



Given

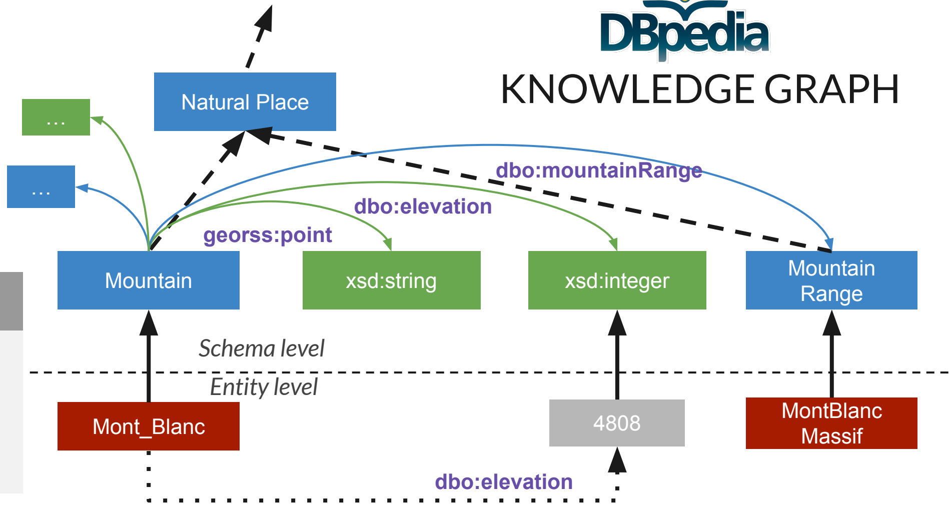
- a relational table T
- a **Knowledge Graph (entities + statements)** and an ontology (**types + predicates)**

T is annotated when:

Tabular Data Annotation

Mountain	xsd:string	xsd:integer	Mountain Range
Name	**Coordinates**	**Height**	**Range**
Le Mont Blanc	45°49′57″N 06°51′52″E	4808	M. Blanc massif
Hohtälli	45°98′96″N 07°80′25″E	3275	Pennine Alps
Monte Cervino	45°58′35″N 07°39′31″E	4478	Pennine Alps

DBpedia
KNOWLEDGE GRAPH



Given

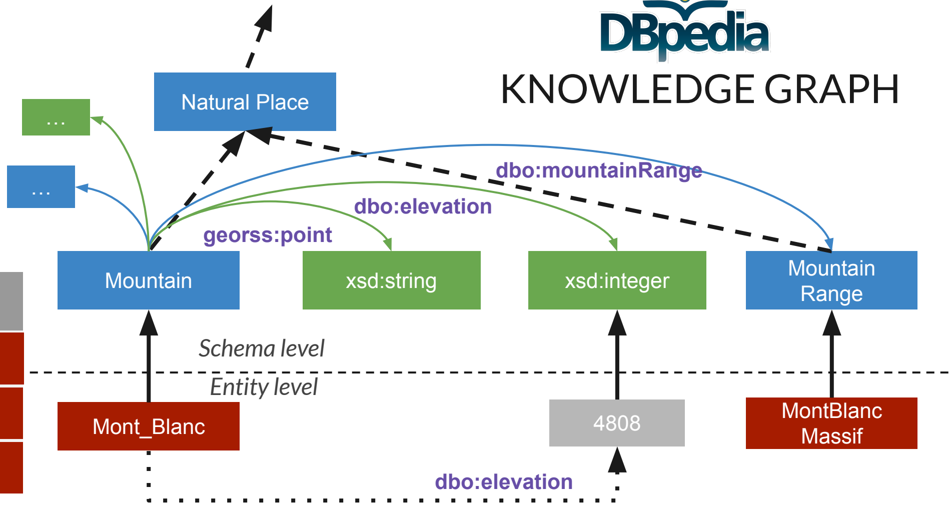
- a relational table T
- a **Knowledge Graph (entities + statements)** and an ontology (**types + predicates)**

T is annotated when:

- each column is associated with one or more KG-types (**CTA**)

Tabular Data Annotation

Mountain	xsd:string	xsd:integer	Mountain Range
Name	Coordinates	Height	Range
Mont_Blanc	45°49′57″N 06°51′52″E	4808	MontBlanc Massif
Hohtälli	45°98′96″N 07°80′25″E	3275	Pennine Alps
Monte Cervino	45°58′35″N 07°39′31″E	4478	Pennine Alps



Given

- a relational table T
- a **Knowledge Graph (entities + statements)** and an ontology (**types + predicates)**

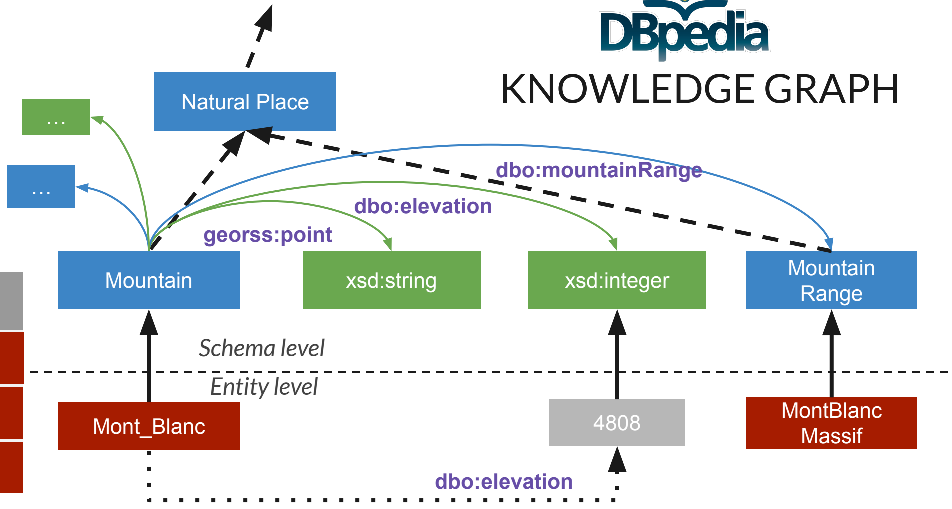
T is annotated when:

- each column is associated with one or more KG-types (**CTA**)
- each cell in "entity columns" is annotated with a KG-entity (**CEA**, a.k.a. **Entity Linking**)

Tabular Data Annotation

Mountain	xsd:string	xsd:integer	Mountain Range
Name	**Coordinates**	**Height**	**Range**
Mont_Blanc	45°49′57″N 06°51′52″E	4808	MontBlanc Massif
NIL	45°98′96″N 07°80′25″E	3275	Pennine Alps
Monte Cervino	45°58′35″N 07°39′31″E	4478	Pennine Alps

DBpedia KNOWLEDGE GRAPH



Given

- a relational table T
- a **Knowledge Graph (entities + statements)** and an ontology (**types + predicates)**

T is annotated when:

- each column is associated with one or more KG-types (**CTA**)
- each cell in "entity columns" is annotated with a KG-entity (**CEA**, a.k.a. **Entity Linking**), or NIL if not in the KB

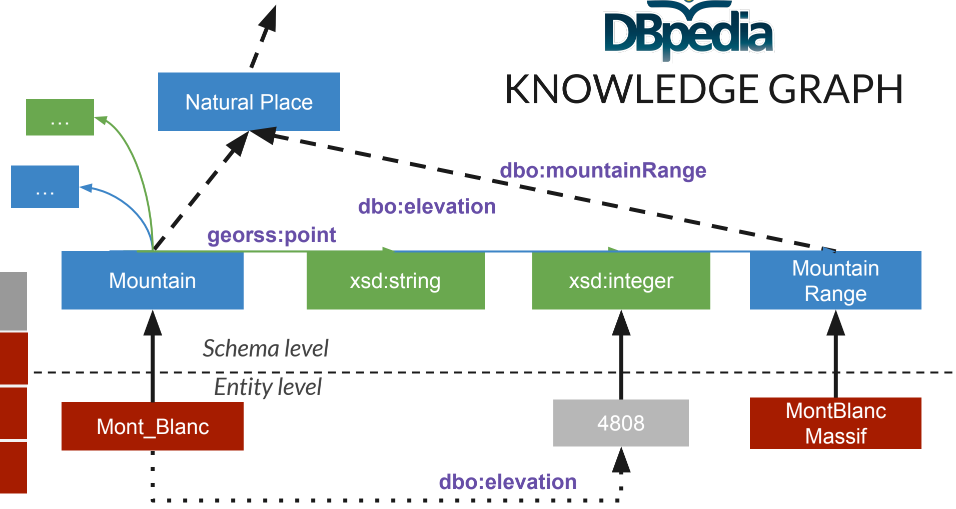
Tabular Data Annotation

Mountain	xsd:string	xsd:integer	Mountain Range
Name	**Coordinates**	**Height**	**Range**
Mont_Blanc	45°49′57″N 06°51′52″E	4808	MontBlanc Massif
NIL	45°98′96″N 07°80′25″E	3275	Pennine Alps
Monte Cervino	45°58′35″N 07°39′31″E	4478	Pennine Alps

georss:point

dbo:elevation

dbo:mountainRange



Given

- a relational table T
- a **Knowledge Graph (entities + statements)** and an ontology (**types + predicates)**

T is annotated when:

- each column is associated with one or more KG-types (**CTA**)
- each cell in “entity columns” is annotated with a KG-entity (**CEA**, a.k.a. **Entity Linking**), or NIL if not in the KB
- some pair of columns is annotated with a binary KG-predicate (**CPA**)

TDU Transformer-based methods

1. [TURL: Table Understanding through Representation Learning]
   - Pre-training/Fine-tuning paradigm on relational tables: MLM and Masked Entity Recovery (MER)
   - Structure-aware Transformer (Tiny-BERT) to capture the row-column structure
   - Pre-training dataset: 670,171 relational tables from the WikiTables corpus
2. [Unicorn: A Unified Multi-tasking Model for Supporting Matching Tasks in Data Integration]
   - Unified model for common data matching tasks: Encoder (DeBERTa) - MoE - Matcher
   - Zero-shot on unseen matching tasks
   - Pre-training dataset as union of single task dataset
3. [TableLlama: Towards Open Large Generalist Models for Tables]
   - Generalist LLM for a diversity of table-based tasks
   - Fine-tuned Llama 2 (7B) with LongLoRA to address the long context challenges
   - TableInstruct: new Instruction-Tuning dataset with a variety of realistic tables and tasks

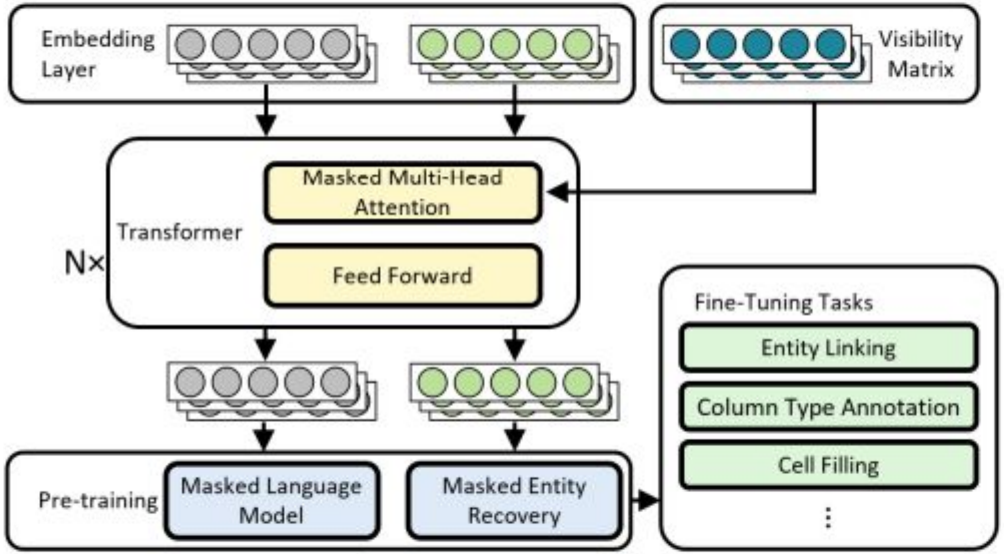
Tasks summary

	Task					
	Annotation/Matching	Augmentation	QA	Fact Verification	Dialogue Generation	Data-to-Text
TURL	CEA CTA CPA	Row Population Cell Filling Schema Aug.	N.A.	N.A.	N.A.	N.A.
Unicorn	CEA CTA Entity Matching Entity Alignment Ontology Matching Schema Matching String Matching	N.A.	N.A.	N.A.	N.A.	N.A.
TableLlama	CEA CTA CPA	Row Population Schema Aug.	Hierarchical Table QA Highlighted Cells QA Hybrid Table QA Table QA	Fact Verification	Table Grounded Dialogue Generation	Highlighted Cells Description

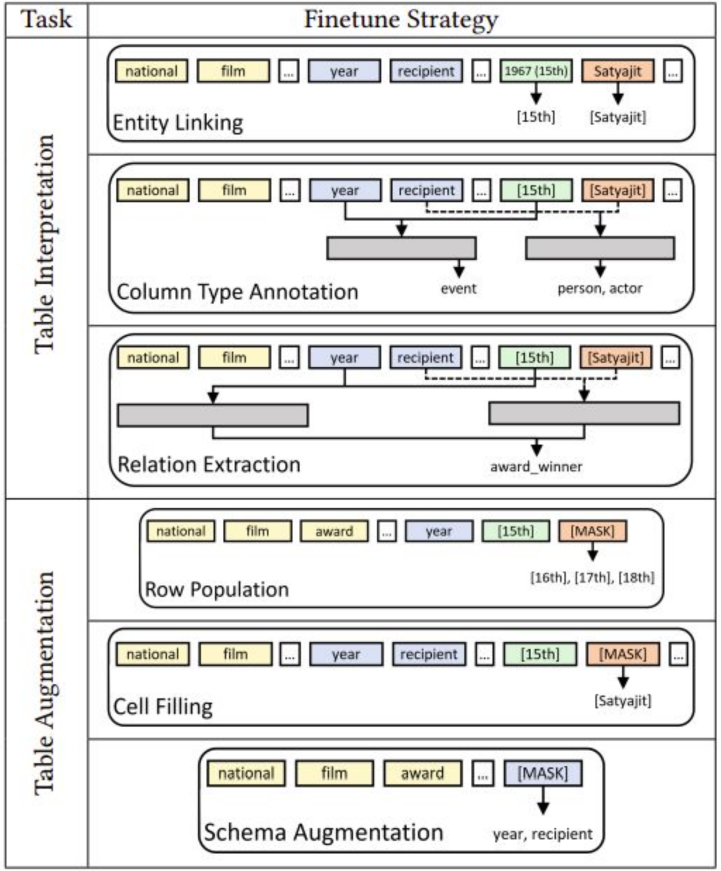
Model summary

	Input	Output	Transformer	Params
TURL	Flatten input table as: **[Table caption, Table Header-1, …, Table Header-M, Row-1, …, Row-N]**	A probability distribution over the N candidates for a mention	Tiny-BERT (Encoder-only)	14.5M
Unicorn	Encode pairs as: **[CLS] S(a) [SEP] S(b) [SEP]** where S($\star$) is a generic function for serializing any pair (a, b) from the matching tasks into a text sequence	A score in [0,1] for every pair (mention, i^{th}-candidate)	DeBERTa (Encoder-only)	147M
TableLlama	Prompt based: **<instruction, table input, question>** • <u>Instruction</u> is a detailed task description • <u>Table input</u> is the concatenation of table metadata (Wikipedia page title, section title and table caption) with the serialized table • <u>Question</u> contains all the information the model needs to complete the task and prompt it to generate an answer.	Autoregressively generated answer given the prompted question	Llama 2 (Decoder-only)	7B

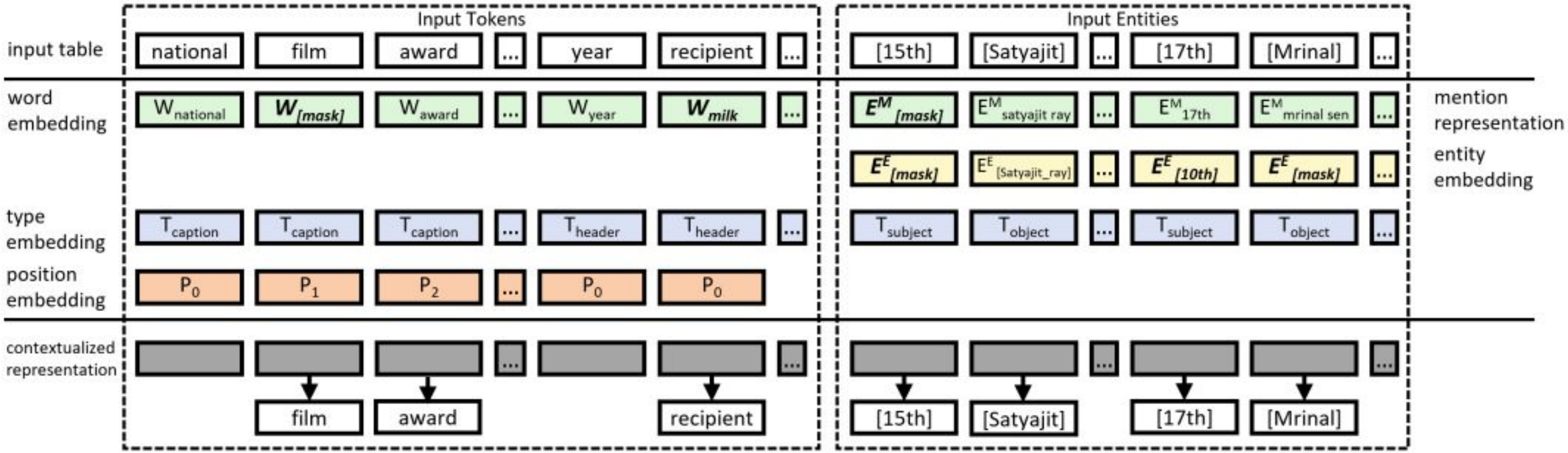
TURL: Overview



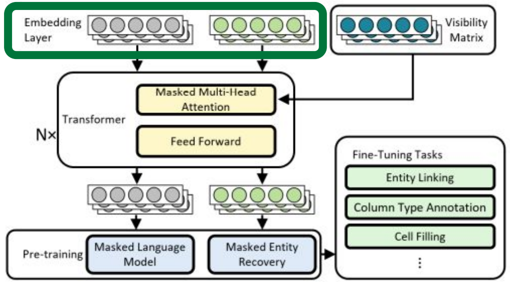
TURL: tasks


Task
Finetune Strategy
Table Interpretation
national
film
...
year
recipient
1967 (15th)
Satyajit
[15th]
[Satyajit]
Entity Linking
Column Type Annotation
event
person, actor
Relation Extraction
award_winner
Table Augmentation
award
[MASK]
[16th], [17th], [18th]
Row Population
Cell Filling
Schema Augmentation
year, recipient

TURL: Example

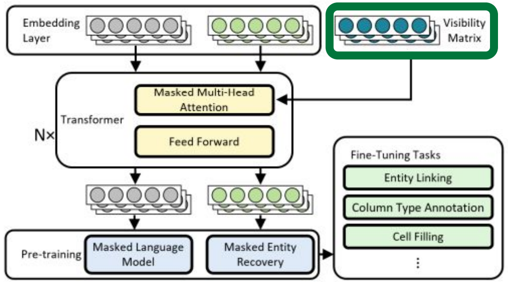


TURL: Embedding Layer



1. Given $T=(C,H,E,e_t)$, flattens the input to a sequence of tokens by concatenating [C;H] and scanning the table content row by row
2. The vector representation of every token w in [C;H] is $\mathbf{x^t = w + t + p}$
    - **w**: word embedding (initialized from pre-trained Tiny-BERT word embedding)
    - **t**: type embedding (i.e., caption/header tokens, randomly initialized)
    - **p**: relative position within C or H (initialized from pre-trained Tiny-BERT word embedding)
3. The vector representation of every e in E is $\mathbf{x^e} = \text{LINEAR}(\,[\mathbf{e^e} ; \mathbf{e^m}]) + \mathbf{t^e}$
    - $\mathbf{e^e}$: embedding of the linked entity (initialized using averaged word embeddings in entity names)
    - $\mathbf{e^m}$: MEAN($\mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_j}, \ldots$), i.e., the mean of the entity mention word embeddings
    - $\mathbf{t^e}$: type embedding vector to differentiate three types of entity cells (i.e., subject/object/topic entities)
4. $\mathbf{x^t}$ and $\mathbf{x^e}$ are fed into the Structured-aware Transformer

TURL: Structure-aware Transformer

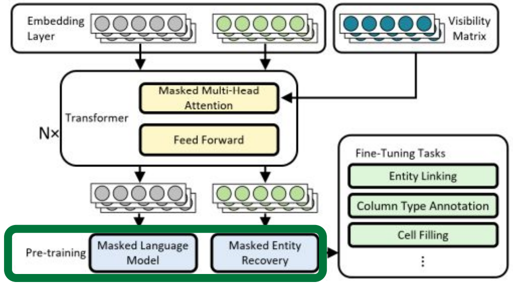


Standard BERT Transformer with modified attention matrix:

- Caption and topic entity are visible to all components of the table
- Entities and text content in the same row or the same column are visible to each other

Figure 5: Graphical illustration of masked self-attention by our visibility matrix. Each token/entity in a table can only attend to its directly connected neighbors (shown as edges here).

TURL: Pre-training objectives



- **MLM**: Given an input token sequence from [C;H], mask some percentage of the tokens at random and then predict masked tokens
- **MER**: Mask a certain percentage of input entity cells and then recover the linked entity based on surrounding entity cells and table metadata.
  - For some percentage of masked entities mask $\mathbf{e}^e$ only, such that the model receives additional entity mention information (from $\mathbf{e}^m$)

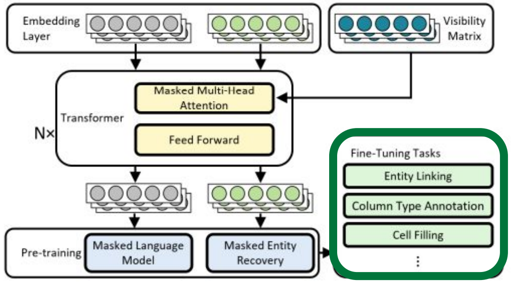
$$P(w) = \frac{\exp\left(\mathrm{LINEAR}(\mathbf{h}^t) \cdot \mathbf{w}\right)}{\sum_{w_k \in \mathcal{W}} \exp\left(\mathrm{LINEAR}(\mathbf{h}^t) \cdot \mathbf{w}_k\right)}$$

$$P(e) = \frac{\exp\left(\mathrm{LINEAR}(\mathbf{h}^e) \cdot \mathbf{e}^e\right)}{\sum_{e_k \in \mathcal{E}} \exp\left(\mathrm{LINEAR}(\mathbf{h}^e) \cdot \mathbf{e}^e_k\right)}$$

$$loss = \sum \log\left(P(w)\right) + \sum \log\left(P(e)\right)$$

Small candidate set

TURL: CEA fine-tuning



Given a fine-tuning EL dataset:

1. Each cell is a potential entity and input both entity mention and table metadata to TURL and obtain a contextualized representation $\mathbf{h}^{\mathbf{e}}$
2. Represent a Knowledge-Base Entity as $\mathbf{e}^{\mathbf{kb}}$ = [$\mathrm{MEAN}_{w \in N}(\mathbf{w})$, $\mathrm{MEAN}_{w \in D}(\mathbf{w})$, $\mathrm{MEAN}_{t \in T}(\mathbf{t})$]
   - **w** initialized from the pre-trained word embedding
   - **t** is the entity-type embedding <u>to be learned</u>
3. Pre-trained entity embeddings are not used
   - goal is to link mentions to entities in a target KB, not necessarily those appear in the pre-training table corpus
4. The model is fine-tuned with a cross-entropy loss

TURL: CEA results

- Fine-tuned on pre-train dataset w/o WikiGS tables
- Wikidata Lookup: top-1 candidate returned
- WikiLookup (Oracle): correct if the groundtruth entity is in the candidate set
- Entity description is very important
- Reweightning: max(0.8 * TURL, WikiLookup)

Table 4: Model evaluation on entity linking task. All three datasets are evaluated with the same TURL + fine-tuning model.

Method	WikiGS			Our Test Set			T2D		
	F1	P	R	F1	P	R	F1	P	R
T2K [35]	34	70	22	-	-	-	82	**90**	76
Hybrid II [16]	64	69	**60**	-	-	-	**83**	85	**81**
Wikidata Lookup	57	67	49	62	62	60	80	86	75
TURL + fine-tuning	**67**	**79**	58	**68**	**71**	**66**	78	83	73
w/o entity desc.	60	70	52	60	63	58	-	-	-
w/o entity type	66	78	57	67	70	65	-	-	-
+ reweighting	-	-	-	-	-	-	82	88	77
WikiLookup (Oracle)	74	88	64	79	82	76	90	96	84

TURL: Pros and Cons

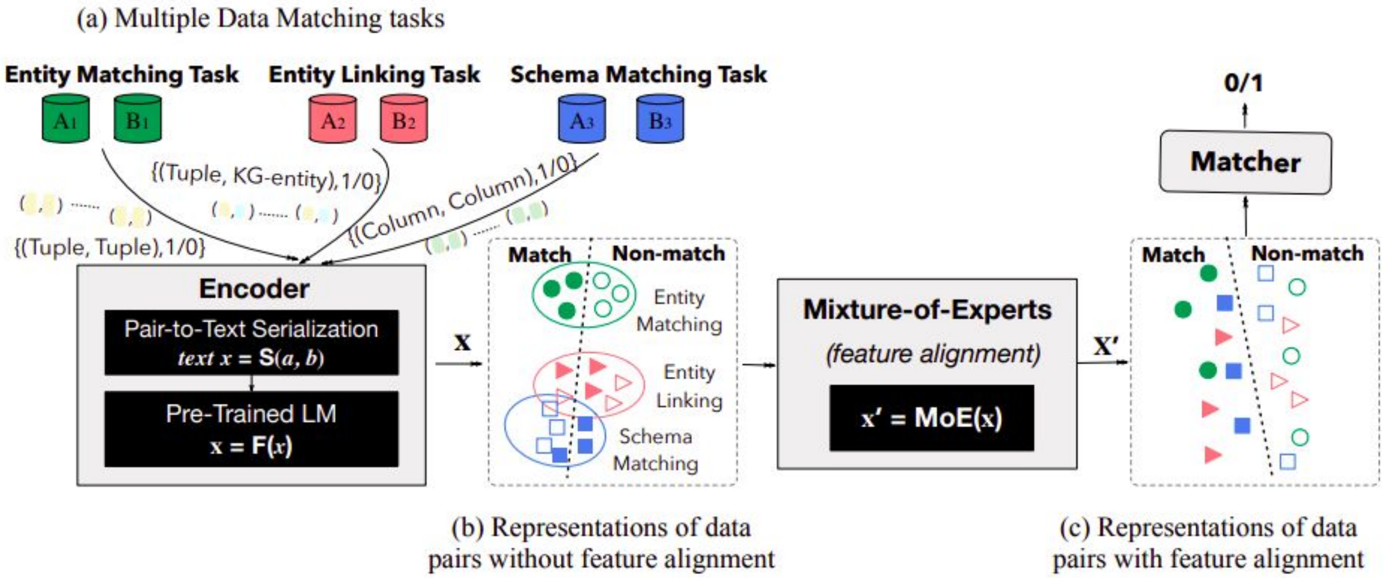
Pros:

- Single pre-trained model
- Considers global table structure
- Masked Entity Recovery (MER) objective
- Easily fine-tuned for:
  - CEA, CPA and CTA
  - Row Population, Cell Filling, Schema Aug.
- Open-Source

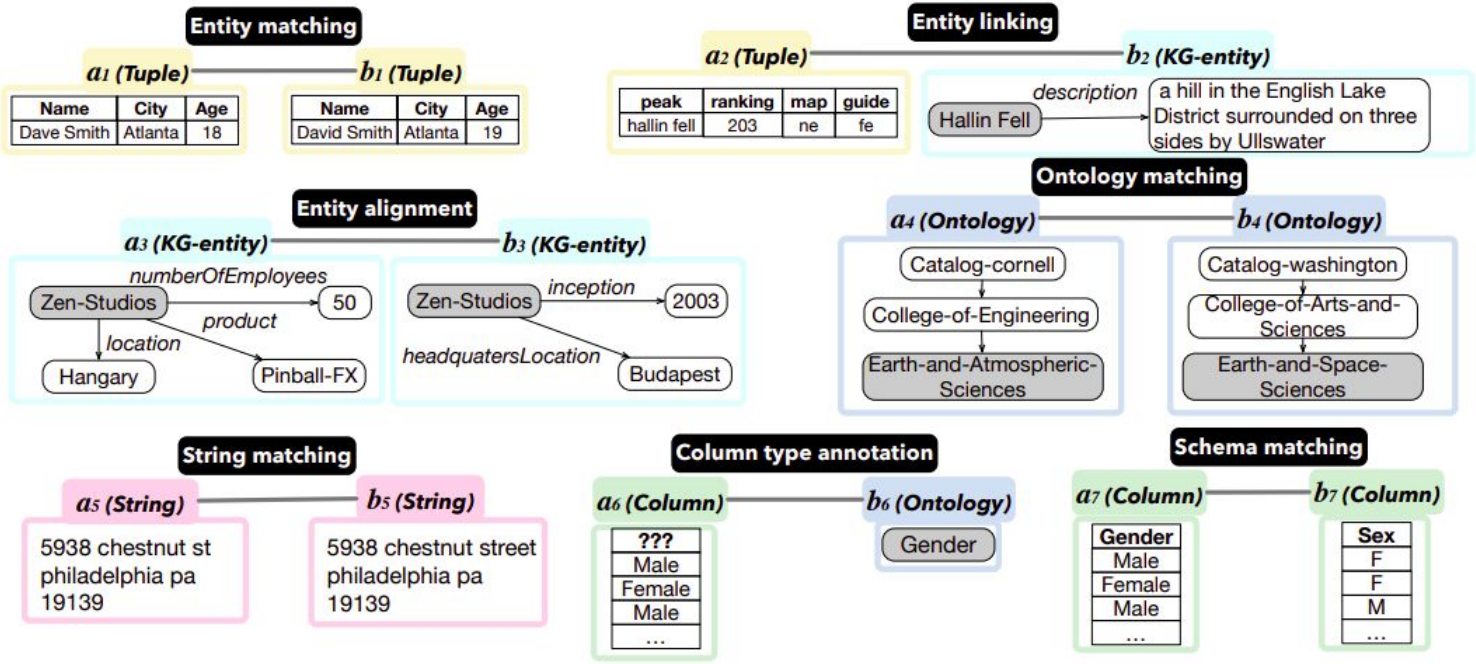
Cons:

- BERT limited context length (512 tokens)
  - Monarch-Mixer can help
- What about too long tables?
- No zero-shot
- No pre-defined NIL prediction
- Unsure results

Unicorn: Overview



Unicorn: tasks



Unicorn: Pair-To-Text

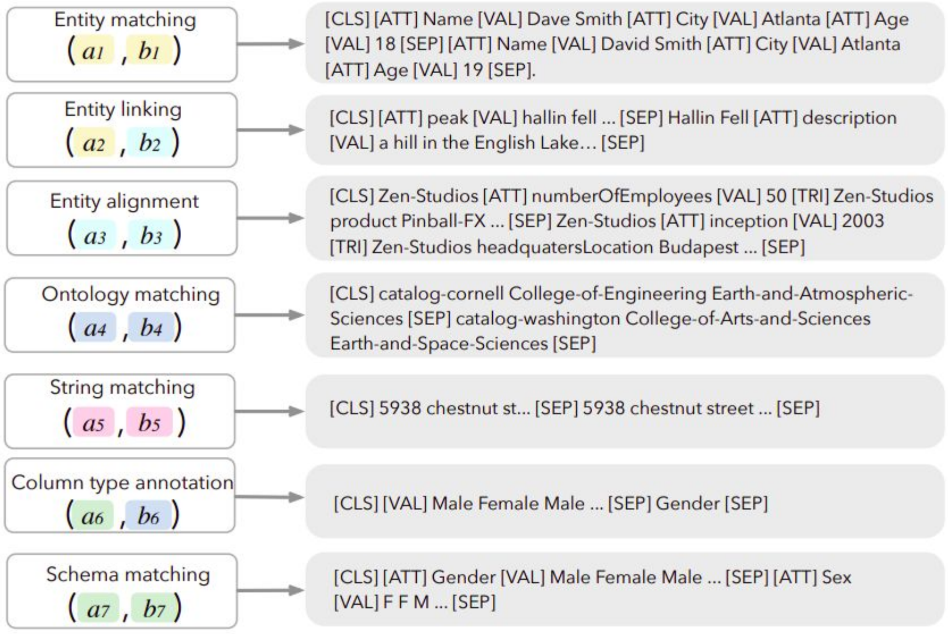
Given a tuple (a,b):

x = S(a, b) = [CLS] S(a) [SEP] S(b) [SEP]

- [CLS] start of the sequence token
- [SEP] separation token
- S({a,b}) = string representation of a/b

x = S(a, b) = [CLS] does S(a) [SEP] match with S(b) [SEP]

Instruction-tuned



Unicorn: Structure-aware encoding

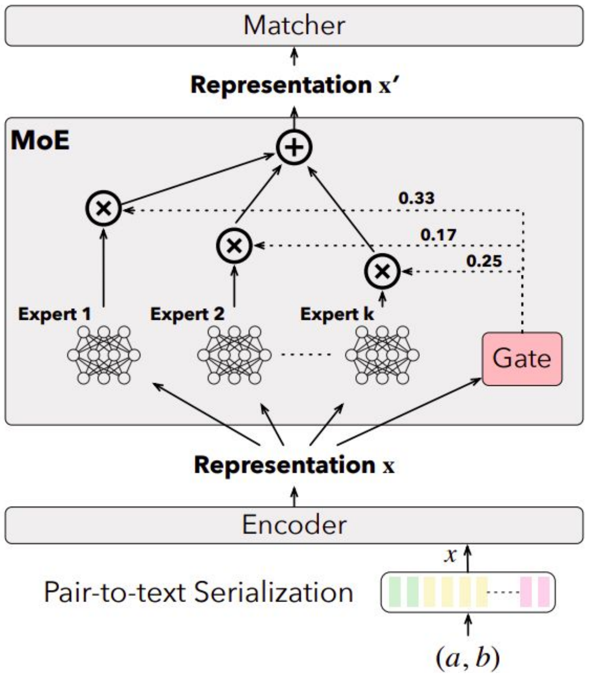
Structure-aware encoding is done on sentence level through DeBERTa:

- Every token *i* represented with both $\mathbf{H}_i$ (content) and $\mathbf{P}_{i|j}$ (relative position of token *i* with token *j*)
- Disentangled attention matrices on contents and positions to model:
  - Content-to-content
  - Content-to-position
  - Position-to-content
  - Position-to-position
- Absolute position to context embedding right after all the transformer layers but before the Softmax during MLM pre-training (Enhanced Mask Decoder, EMD)

Unicorn: MoE

MoE specializes the feature space, i.e., it maps different distributions of multiple tasks to a same shared distribution

- All the experts to be used in a balanced way
- Any specific training pair has to be assigned to a few specific experts



Unicorn: Results

Type	Task	Metric	Unicorn w/o MoE	Unicorn	Unicorn ++	Previous SOTA (Paper)
EM	Walmart-Amazon	F1	85.12	86.89	**86.93**	86.76 (Ditto [30])
	DBLP-Scholar	F1	95.38	95.64	**96.22**	95.6 (Ditto [30])
	Fodors-Zagats	F1	97.78	**100**	97.67	**100** (Ditto [30])
	iTunes-Amazon	F1	94.74	96.43	**98.18**	97.06 (Ditto [30])
	Beer	F1	90.32	90.32	87.5	**94.37** (Ditto [30])
CTA	Efthymiou	Acc.	98.08	98.42	**98.44**	90.4 (TURL [10])
	T2D	Acc.	98.81	99.14	**99.21**	96.6 (HNN+P2Vec [5])
	Limaye	Acc.	96.11	96.75	**97.32**	96.8 (HNN+P2Vec [5])
EL	T2D	F1	79.96	91.96	**92.25**	85 (Hybrid I [20])
	Limaye	F1	83.12	86.78	**87.9**	82 (Hybrid II [20])
StM	Address	F1	97.81	98.68	99.47	**99.91** (Falcon [39])
	Names	F1	86.12	91.19	**96.8**	95.72 (Falcon [39])
	Researchers	F1	96.59	97.66	**97.93**	97.81 (Falcon [39])
	Product	F1	84.61	82.9	**86.06**	67.18 (Falcon [39])
	Citation	F1	96.34	96.27	**96.64**	90.98 (Falcon [39])
ScM	FabricatedDatasets	Recall	81.19	**89.6**	89.35	81 (Valentine [27])
	DeepMDatasets	Recall	66.67	96.3	96.3	**100** (Valentine [27])
OM	Cornell-Washington	Acc.	90.64	**92.34**	90.21	80 (GLUE [15])
EA	SRPRS: DBP-YG	Hits@1	99.46	99.67	99.49	**100** (BERT-INT [46])
	SRPRS: DBP-WD	Hits@1	97.11	97.22	97.28	**99.6** (BERT-INT [46])
AVG			90.8	94.21	**94.56**	91.84
Model Size			139M	147M	147M	995.5M

Unicorn: Zero-Shot Results

Type	Task	Metric	Unicorn w/o MoE	Unicorn	Unicorn-ins	SOTA (# of labels)
EM	DBLP-Scholar	F1	90.91	95.39	**97.08**	95.6 (22,965)
CTA	Limaye	Acc.	96.2	96.59	96.5	**96.8** (80)
EL	Limaye	F1	74.16	78.92	**82.8**	82 (-)
StM	Product	F1	60.71	74.92	**78.76**	67.18 (1,020)
ScM	DeepMDatasets	Recall	74.07	92.59	96.3	**100** (-)
EA	SRPRS: DBP-WD	Hits@1	95.55	97.25	96.17	**99.6** (4,500)
AVG			81.93	89.28	**91.27**	90.2

- Task-Type seen during pre-training
- Task-Data unseen

Type	Task	Metric	Unicorn-ins	SOTA (# of labels)
EM	DBLP-Scholar	F1	94.5	**95.6** (22,965)
CTA	Limaye	Acc.	96.23	**96.8** (80)
EL	Limaye	F1	79.59	**82** (-)
StM	Product	F1	**74.26**	67.18 (1,020)
ScM	DeepMDatasets	Recall	88.89	**100** (-)
EA	SRPRS: DBP-WD	Hits@1	97	**99.6** (4,500)
AVG			88.41	**90.2**

- Task-Type unseen during pre-training
- Task-Data unseen

Unicorn: Pros and Cons

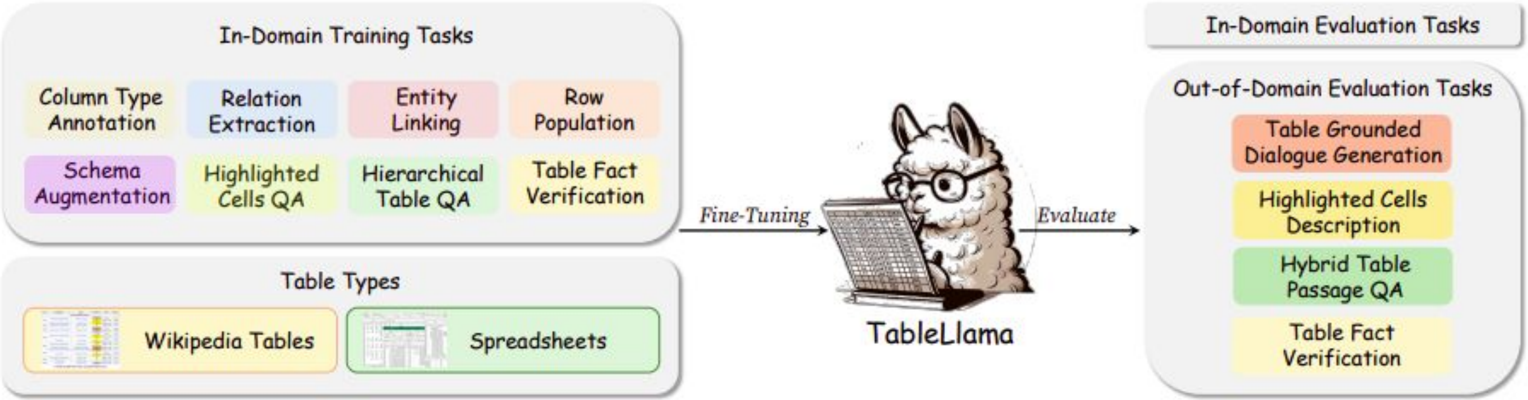
Pros:

- Single Multi-Task pre-trained model
- End-to-End training
- No global information needed
- Zero-shot enabled
- Table/KB agnostic
- Suitable for Meta-Learning (Model Agnostic Meta Learning)
- Extension to Ensamble of Matchers
  - Uncertainty estimation
- Open-Source

Cons:

- Entity Linking definition differs:
  - Unicorn matches entire row with KB Entity
- No pre-defined NIL prediction
  - Binary classifier can be helpful
- No CPA
  - Can this be casted as a matching task?
- Poor scalability:
  - Comparison with every candidates for every single mention

TableLlama: Overview



TableLlama: CEA example

Entity Linking

Below is an instruction that describes a task, paired with an input that provides further context. Write a response that appropriately completes the request.

Instruction: This is an entity linking task. The goal for this task is to link the selected entity mention in the table cells to the entity in the knowledge base. You will be given a list of referent entities, with each one composed of an entity name, its description and its type. Please choose the correct one from the referent entity candidates. Note that the Wikipedia page, Wikipedia section and table caption (if any) provide important information for choosing the correct referent entity.

Input: [TLE] The Wikipedia page is about A-League all-time records. The Wikipedia section is about Average season attendances. [TAB] col: | season | league average | total gate receipts | highest club | average | lowest club | average | row 1: | 2005-06 | 10,955 | 920,219 | Sydney FC | 16,669 | New Zealand Knights | 3,909 | [SEP] row 2: | 2006-07 | 12,927 | ...

Question: The selected entity mention in the table cell is: Melbourne Victory. The column name for 'Melbourne Victory' is highest club. The referent entity candidates are: <Melbourne Victory FC W-League [DESCRIPTION] None [TYPE] SoccerClub>, <2016\u201317 Melbourne Victory FC season [DESCRIPTION] None [TYPE] SoccerClubSeason>, <2011\u201312 Melbourne Victory season [DESCRIPTION] Association football club 2011/12 season for Melbourne Victory [TYPE] SoccerClubSeason>, ... What is the correct referent entity for the entity mention 'Melbourne Victory' ?

Response: <Melbourne Victory [DESCRIPTION] association football team from Australia [TYPE] SoccerClub>.

TableLlama: CTA example

Column Type Annotation

Below is an instruction that describes a task, paired with an input that provides further context. Write a response that appropriately completes the request.

Instruction:
This is a column type annotation task. The goal for this task is to choose the correct types for one selected column of the table from the given candidates. The Wikipedia page, section and table caption (if any) provide important information for choosing the correct column types.

Input:
[TLE] The Wikipedia page is about 1958 Nippon Professional Baseball season. The Wikipedia section is about Central League. The table caption is Pitching leaders. [TAB] col: | stat | player | team | total | [SEP] row 1: | Wins | Masaichi Kaneda | Kokutetsu Swallows | 31| [SEP] row 2: | Losses | Noboru Akiyama | ...

Question:
The column 'player' contains the following entities: <Masaichi Kaneda>, <Noboru Akiyama>, etc. The column type candidates are: tv.tv_producer, astronomy.star_system_body, location.citytown, sports.pro_athlete, biology.organism, medicine.muscle, baseball.baseball_team, baseball.baseball_player, aviation.aircraft_owner, people.person, ... What are the correct column types for this column (column name: player; entities: <Masaichi Kaneda>, <Noboru Akiyama>, etc)?

Response:
sports.pro_athlete, baseball.baseball_player, people.person.

TableLlama: TableInstruct

Large-scale instruction tuning dataset with diverse, realistic tasks based on real-world tables. **TableInstruct** boasts a collection of:

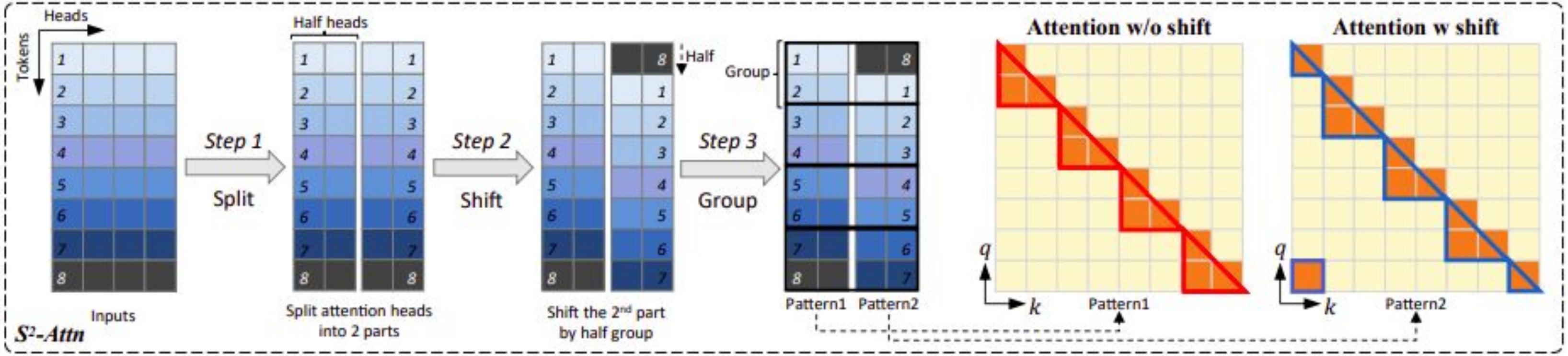
- **14 datasets of 11 tasks** in total, which is curated from **1.24M** tables containing **2.6M** instances
- All data items are **unified into an instruction tuning manner** for LLM training
- All data items in **TableInstruct** are collected from real tables and real tasks
- Both **in-domain training tasks** to empower the model with fundamental table understanding abilities, and **in-domain & out-of-domain evaluation tasks** to test the model's generalization and high-level reasoning ability

TableLlama: TableInstruct

Task Category	Task Name	Dataset	In-domain	#Train (Table/Sample)	#Test (Table/Sample)	Input Token Length min	max	median
Table Interpretation	Col Type Annot.	TURL (Deng et al., 2020)	Yes	397K/628K	1K/2K	106	8192	2613
	Relation Extract.		Yes	53K/63K	1K/2K	2602	8192	3219
	Entity Linking		Yes	193K/1264K	1K/2K	299	8192	4667
Table Augmentation	Schema Aug.	TURL (Deng et al., 2020)	Yes	288K/288K	4K/4K	160	1188	215
	Row Pop.		Yes	286K/286K	0.3K/0.3K	264	8192	1508
Question Answering	Hierarchical Table QA	HiTab (Cheng et al., 2022b)	Yes	3K/7K	1K/1K	206	5616	978
	Highlighted Cells QA	FeTaQA (Nan et al., 2022)	Yes	7K/7K	2K/2K	261	5923	740
	Hybrid Table QA	HybridQA (Chen et al., 2020b)	No	–	3K/3K	248	2497	675
	Table QA	WikiSQL (Zhong et al., 2017)	No	–	5K/16K	198	2091	575
	Table QA	WikiTQ (Pasupat and Liang, 2015)	No	–	0.4K/4K	263	2688	709
Fact Verification	Fact Verification	TabFact (Chen et al., 2020a)	Yes	16K/92K	2K/12K	253	4975	630
		FEVEROUS (Aly et al., 2021)	No	–	4K/7K	247	8192	648
Dialogue Generation	Table Grounded Dialogue Generation	KVRET (Eric et al., 2017)	No	–	0.3K/0.8K	187	1103	527
Data-to-Text	Highlighted Cells Description	ToTTo (Parikh et al., 2020)	No	–	7K/8K	152	8192	246

TableLlama: Model Architecture

- Llama 2 7B
- LongLora with Shift-Short Attention (S^2-Attention)
    - Context split into several groups and conducts attention in each group individually
    - Tokens shifted by 1/2 group size in half attention heads to ensure the information flow between neighboring groups
    - Shift-Short attention with group size 2048 to approximate total 8196 context length training



TableLlama: In-Domain Results

For CTA and EL they uniformly subsampled the test-set from the original test data

In-domain Evaluation				
Datasets	Metric	Base	TableLlama	SOTA
Column Type Annotation	F1	3.01	94.39	**94.54***† (Deng et al., 2020)
Relation Extraction	F1	0.96	91.95	**94.91***† (Deng et al., 2020)
Entity Linking	Accuracy	31.80	**93.65**	84.90*† (Deng et al., 2020)
Schema Augmentation	MAP	36.75	**80.50**	77.55*† (Deng et al., 2020)
Row Population	MAP	4.53	58.44	**73.31***† (Deng et al., 2020)
HiTab	Exec Acc	14.96	**64.71**	47.00*† (Cheng et al., 2022a)
FeTaQA	BLEU	8.54	**39.05**	33.44 (Xie et al., 2022)
TabFact	Accuracy	41.65	82.55	**84.87*** (Zhao and Yang, 2022)

TableLlama: Generalization

Training Data	In-domain								Out-of-domain					
	ColType	RelExtra	EntLink	ScheAug	RowPop	HiTab	FeTaQA	TabFact	FEVER.	HybridQA	KVRET	ToTTo	WikiSQL	WikiTQ
	F1	F1	Acc	MAP	MAP	Acc	BLEU	Acc	Acc	Acc	Micro F1	BLEU	Acc	Acc
Base	3.01	0.96	31.80	36.75	4.53	14.96	8.54	41.65	23.66	20.72	38.90	10.39	14.84	25.48
ColType	94.32	0	0	0	0	0.13	0.52	0	0	0	0	1.11	0.35	0.21
RelExtra	45.69	**93.96**	0.45	8.72	0.99	7.26	1.44	0	2.38	8.17	5.90	5.60	7.02	9.58
EntLink	0.86	0.03	88.45	2.31	0.94	5.37	4.79	0	39.04	3.06	0	1.76	3.42	7.07
ScheAug	-	-	-	80.00	-	-	-	-	-	-	-	-	-	-
RowPop	-	-	-	-	53.86	-	-	-	-	-	-	-	-	-
HiTab	0.20	0.14	7.15	40.81	5.45	63.19	2.07	49.46	46.81	24.70	38.70	2.45	32.86	27.97
FeTaQA	0	0.40	0	30.23	0.15	19.57	38.69	1.20	1.21	**33.79**	**50.69**	**23.57**	13.79	27.12
TabFact	0	0	0	0	0	0	0	74.87	56.15	0	0	0	0	0
TableInstruct	**94.39**	91.95	**93.65**	**80.50**	**58.44**	**64.71**	**39.05**	**82.55**	**72.30**	27.61	48.73	20.77	**41.68**	**31.63**

TableLlama: Pros and Cons

Pros:

- Enable the latest developments on LLMs:
  - RLHF with HITL
  - Self-Rewarding LLM
- A lot of fine-tuning strategies:
  - LoRA/VeRA
  - Mixture of LoRAs
  - In-Context Learning?
- KB agnostic

Cons:

- What about the sub-sampled test set?
- Poor scalability
- Prompt can become quite long
- Prohibitive pre-training (if needed):
  - 9d on 48 A100-80GB
- What about hallucinations?

Open Questions

- **Single, unified and standard benchmark for those methods is missing!**
  - How this models perform on SemTab Challange datasets?
- **Hard to make them comparable**
  - Every method has its own test case given the same test dataset
- **How they perform in a zero-shot setting?**
- **How they perform w.r.t. NIL entities?**
  - What they tell us about NIL entities?
- **Can Meta-Learning be applied to learn new matching tasks with few examples?**
  - Unicorn is the most suitable
- **Can those method be fine-tuned with Human-In-The-Loop feedbacks? If so, how?**
  - TableLlama is the most suitable (RLHF)
- **What about candidate generation?**

Thank you for your

$$\text{Attention}(Q,K,V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Next steps

- Creation of a shared GitHub repository
  - PyTorch-Lightning based
  - Collection of multiple Semantic Table Interpretation (STI) algorithms
  - Standardized dataset ingestion for safe Train/Val/Test splits
- Test TURL, Unicorn and TableLlama on SemTab dataset
- Test MAML on Unicorn?