Transformers for Tabular Data Representation: Models and Applications

Gilbert Badaro

Paolo Papotti

EURECOM
Sophia Antipolis

VLDB 2022

Presenters

- Gilbert Badaro
  - Principal Data Scientist at Amadeus
    - Post Doc Fellow at EURECOM (May 2022)
    - Ph.D. from American University of Beirut, Lebanon, 2020
    - Data Warehouse Architect and Developer, CMS, CERN
  - Focus on NLP and ML

- Paolo Papotti
  - Associate Professor at EURECOM
    - Ph.D. from Roma Tre University, Italy, 2007
    - Research positions at QCRI and Arizona State University
  - Focus on data management and information quality

Tutorial Outline

- Motivation
  - Natural Language and Data-centric Applications
- Language Models and Transformers **[q&a]**
- Developing & Consuming Tabular Data Representation
  - Training Datasets
  - Input Processing **[q&a]**
  - Model Training & Architecture **[q&a]**
  - Tabular Language Model
  - Consuming Tabular LMs **[q&a]**
- Open Challenges **[q&a]**

Text and tabular data

- Several applications use both text and **tabular** data

Population in Million by Country

Country	Capital	Population
Australia	Canberra	25.69
France	Paris	67.39
Bolivia	La Paz	11.67

France

Capital	Paris
Population	67.39M
Size	644K Km2
President	Emmanuel Macron

Appears and Goals

Club	Season	League		
		Division	Apps	Goals
Cannes	1988-89	Ligue 3	2	0
	1989-90		0	0
	1990-91		28	1
	1991-92		31	5 (1)

Table-based Fact-Checking (TFC)

- **Fact-checking** (tabular setting): verify if an input claim, expressed in natural language (NL) is true/false against some trusted *structured data*

Population in Million by Country

Country	Capital	Population
Australia	Canberra	25.69
France	Paris	67.39
Bolivia	La Paz	11.67

Input claim: France has a population of 67.39 million.
Output: **True**

Input claim: Bolivia has more citizens than France.
Output: **False**

(Aly et al, 2022; Karagiannis et al, 2020)

- **Text Entailment:** check whether an input relational table implies or not a given NL claim

Input Text: France has a more than double population of Australia.
Output: Entail

Input Text: France has a higher population density than Bolivia.
Output: Does not entail/Not Enough Information

(Eisenschlos et al, 2020)

Demo

- https://coronacheck.eurecom.fr/en

Question Answering (QA)

- Find the cell(s) that answer a given input NL question
- Complexity ranges from simple lookup queries to complex ones involving aggregations and numerical reasoning

Population in Million by Country

Country	Capital	Population
Australia	Canberra	25.69
France	Paris	**67.39**
Bolivia	La Paz	11.67

Question: What is the population number of France?
Output: 67.39

Population in Million by Country

Country	Capital	Population
Australia	Canberra	25.69
France	Paris	**67.39**
Bolivia	La Paz	**11.67**

Question: What is the total population in France and Bolivia?
Answer: 79.06

(Herzig et al, 2020)

Demo

- google/tapas-base-finetuned-wtq · Hugging Face

Semantic Parsing (SP): Text-2-SQL

- Given a question in NL and a table, generate a declarative query expressed in SQL (or SPARQL)

Population in Million by Country (PMC)

Country	**Capital**	**Population**
Australia	Canberra	25.69
France	Paris	67.39
Bolivia	La Paz	11.67

NL text: Find the capital of Australia.
Output: Select Capital from PMC where Country = "Australia";

NL text: What is the average population?
Output: Select AVG(Population) from PMC;

(Yu et al, 2021; Gkini et al, 2021)

Table Retrieval (TR)

- Given a question in NL and a **set** of tables, identify the tables that can answer the question

Population in Millions by Country

Country	**Capital**	**Population**
Australia	Canberra	25.69
France	Paris	67.39
Bolivia	La Paz	11.67

GDP by Country in Trillions USD

Country	**Capital**	**GDP**
Germany	Berlin	3.806
France	Paris	2.603
Australia	Canberra	1.331

Statistics for France

Metric	**Value**	**Year**
Population	67M	2020
GDP	2.6	2020
Size	La Paz	11.67

Question: What is the GDP of Germany?
Table: GDP by Country in Trillions USD
(**Answer:** 3.806)

(Wang et al, 2021; Pan et al, 2021)

Why are they challenging?

Task ID	Task Label	Tasks Coverage	Input NL NL	Output
TFC	Table-based Fact-Checking or Entailment	Fact-Checking Text Refusal/Entailment	Table + Claim	True/False Refused/Entailed (Data Evidence)
QA	Question Answering	Retrieving the Cells for the Answer	Table + Question	Answer Cells
SP	Semantic Parsing	Text-to-SQL	Table + NL Query	Formal QL
TR	Table Retrieval	Retrieving Table that Contains the Answer	Tables + Question	Relevant Table(s)
TMP	Table Metadata Prediction	Column Type Prediction Table Type Classification Header Detection Cell Role Classification Column Relation Annotation Column Name Prediction	Table	Column Types Table Types Header Row Cell Role Relation between Two Cols Column Name
DI	Data Imputation	Cell Content Population	Table with Corrupted Cell Values	Table with Complete Cell Values

Table Metadata Prediction (TMP)

- Given an input table with corrupted or missing metadata, predict
  - column types and headers, and
  - intra-tables relationships
    - equivalence between columns, entity linking/resolution

Population in Millions by Country

[redacted]	Capital	Population
Australia	Canberra	25.69
France	Paris	67.39
Bolivia	La Paz	11.67

Predict that the missing column header is **Country**

Predict that the table type is a **relational** table

(Cappuzzo et al, 2020; Deng et al. 2020; Li, Yuliang et al 2020)

Data Imputation (DI)

- Given a table with corrupted/missing values, populate the missing cell data

Population in Millions by Country

Country	**Capital**	**Population**
Australia	Canberra	25.69
[redacted]	Paris	67.39
Bolivia	La Paz	11.67

Population in Millions by Country

Country	**Capital**	**Population**
Australia	Canberra	25.69
France	Paris	67.39
Bolivia	La Paz	11.67

(Deng et al. 2020; Tang et al, 2021)

Text and tabular data

- Several applications use both
  - Table-based Fact-Checking/Text Entailment (TFC)
  - Question Answering (QA)
  - Semantic Parsing / Text-to-SQL (SP)
  - Table Retrieval (TR)
  - Table Metadata Prediction (TMP)
    - detecting column types, table types, relations, header cells,
    - entity resolution and linking; column name prediction
  - Data imputation (DI)

How can we exploit NL understanding in building such applications?

Tutorial Outline

- Motivation
  - Data-centric Applications and Natural Language
- Language Models and Transformers
- Developing & Consuming Tabular Data Representation
  - Training Datasets
  - Input Processing
  - Model Training & Architecture
  - Tabular Language Model
  - Consuming Tabular LMs
- Open Challenges

Deep learning can help with NL text

- A language model (LM) is a probability distribution over sequences of words
  - Given a sequence of words, it
    - assigns a probability to the sequence
    - predicts the most probable next word in the sequence

$$p_{\text{LM}}(\text{the house is small}) > p_{\text{LM}}(\text{small the is house})$$

- Modern LMs are obtained by (unsupervised) **pre-training** on large text corpora

- Pre-trained LMs enable state-of-the-art results in downstream NLP tasks, even in cases with limited amount of annotated training data

What can we do with Language Models?

Sydney is the capital city of the state of New South Wales, and the most populous city in Australia and Oceania. Located on Australia's east coast, the metropolis surrounds Port Jackson and extends about 70 km (43.5 mi) on its periphery [...]. Sydney is made up of 658 suburbs, spread across 33 local government areas. Residents of the city are known as "Sydneysiders". As of June 2020, Sydney's estimated metropolitan population was 5,361,466, meaning the city is home to approximately 66% of the state's population. Nicknames of the city include the 'Emerald City' and the 'Harbour City'.

Fact-checking (text):
Sydney's population as of June 2020 is less than 2 millions.
False

Question Answering:
What is an example of a nickname for Sydney?
Emerald City / Harbour City

Sentiment Analysis:
Neutral

Document Classification:
Geography

Translation to French:
Sydney est la capitale de l'État de la Nouvelle-Galles du Sud et la ville la plus peuplée d'Australie et d'Océanie.

Using a small labeled dataset, we customize the same pre-trained LM for several tasks

How does it work? Big Picture

1- Develop LM through *pre-training* using large unlabeled text corpora

Transformers → Language Model

2- *Fine-tune* LM using (relatively small) labeled training data for target application

Sydney is the capital city of the state of New South Wales, and the most populous city in Australia and Oceania. Located on Australia's east coast, the metropolis surrounds Port Jackson and extends about 70 km (43.5 mi) on its periphery towards the Blue Mountains to the west, Hawkesbury to the north, the Royal National Park to the south and Macarthur to the south-west. Sydney is made up of 658 suburbs, spread across 33 local government areas. Residents of the city are known as "Sydneysiders". As of June 2020, Sydney's estimated metropolitan population was 5,361,466, meaning the city is home to approximately 66% of the state's population. Nicknames of the city include the 'Emerald City' and the 'Harbour City'.

Neutral

Transformer Based LM → Fine-Tuned LM

transfer learning

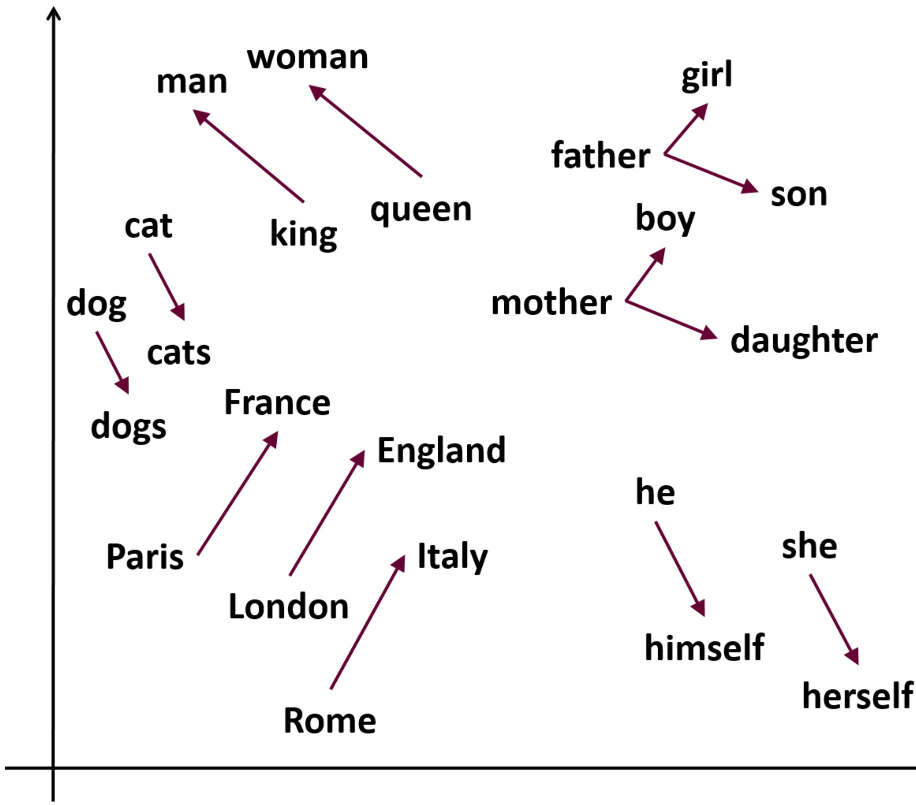
3- Given a new paragraph, predict sentiment

Paris is the capital and most populous city of France, with an estimated population of 2,165,423 residents in 2019 in an area of more than 105 km² (41 sq mi), making it the 34th most densely populated city in the world in 2020. Since the 17th century, Paris has been one of the world's major centers of finance, diplomacy, commerce, fashion, gastronomy, science, and arts, and has sometimes been referred to as the capital of the world.

Fine-Tuned LM → Neutral

Embeddings

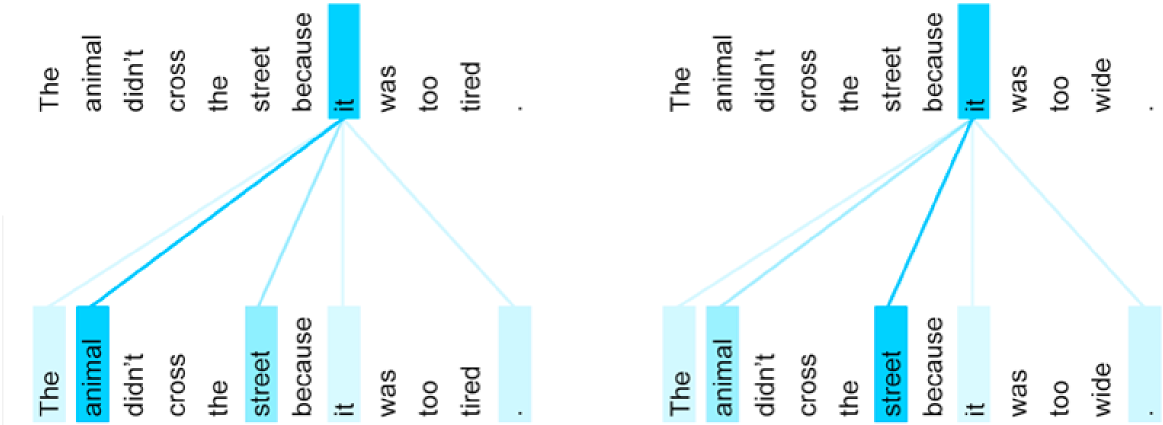
- Focus on **neural** language models
- Instead of using probabilities, each word is mapped to the distributed representation encoded in the networks' hidden layers
  - one word → one vector
- Use continuous representations based on n-dimensional real-valued **word** (token) **embeddings**
  - words closer in the vector space are expected to be similar in meaning

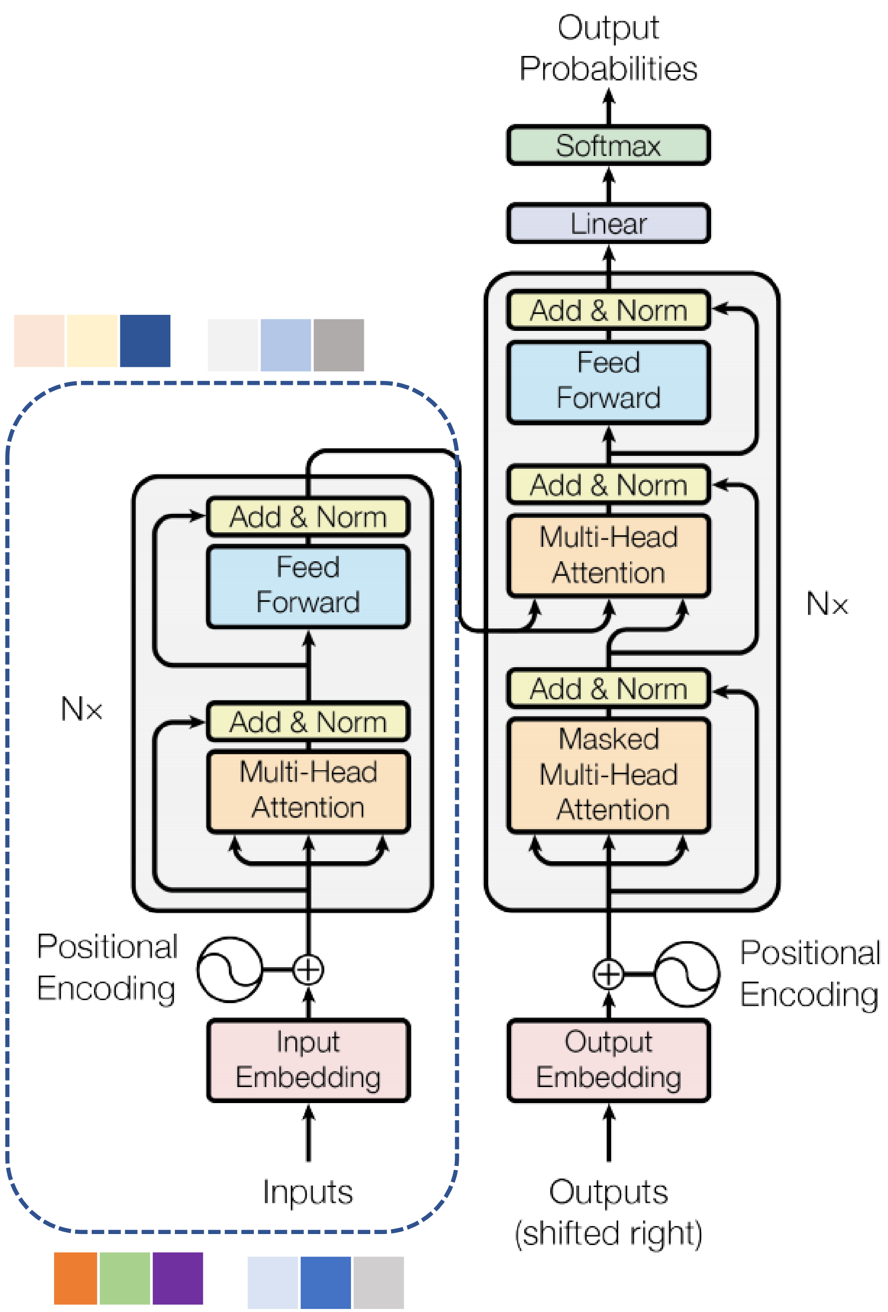


(Mikolov et al, 2013)

Transformers 1/3

- Many ways to obtain a LM
- **Transformers** introduced *parallelism* (→GPU/TPU) and enabled *larger models*
  - **Encoder**-decoder architecture
  - (Self) Attention mechanism to understand relationships between all words in a sentence, regardless of their respective position

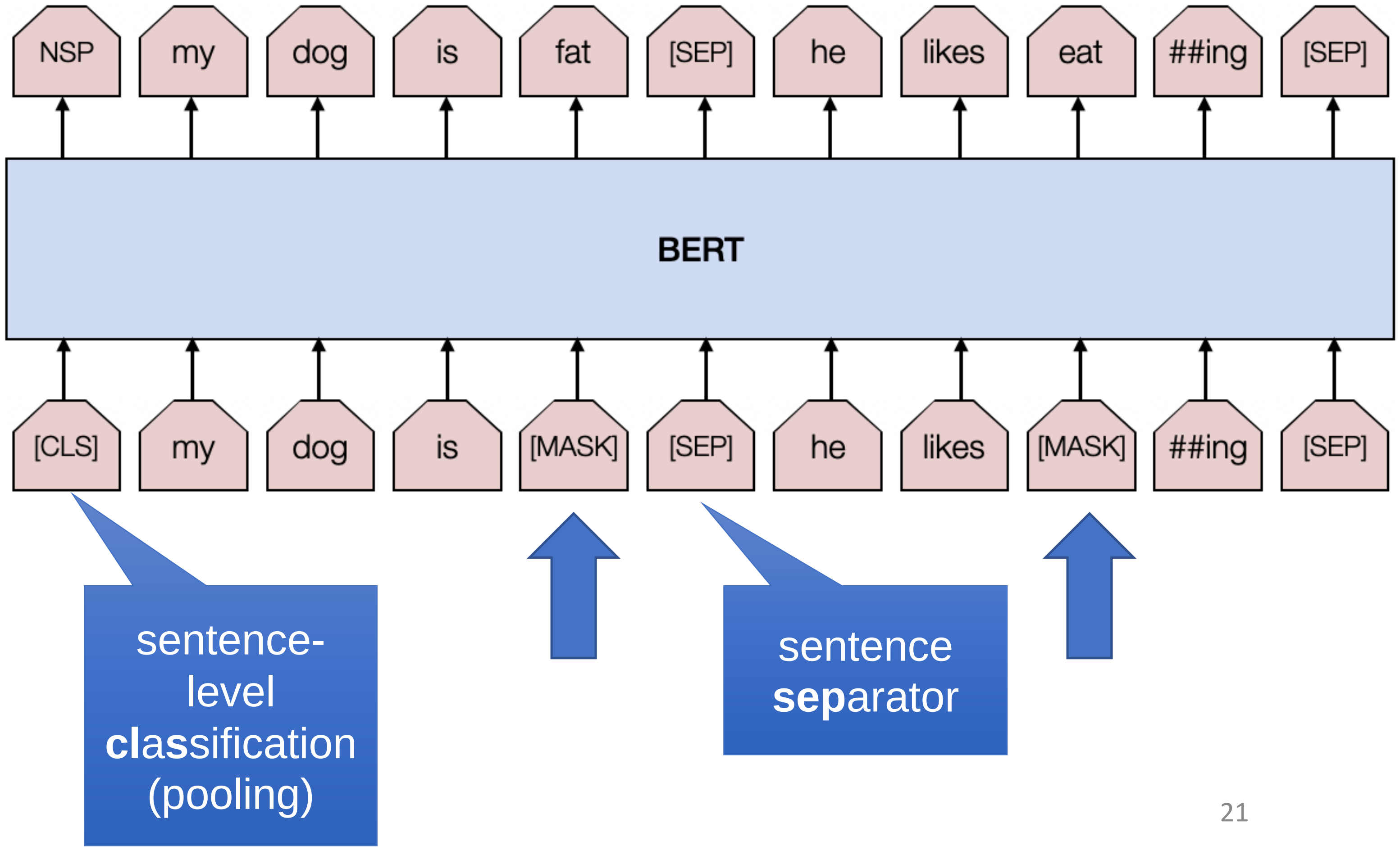




(Vaswani et al, 2017)

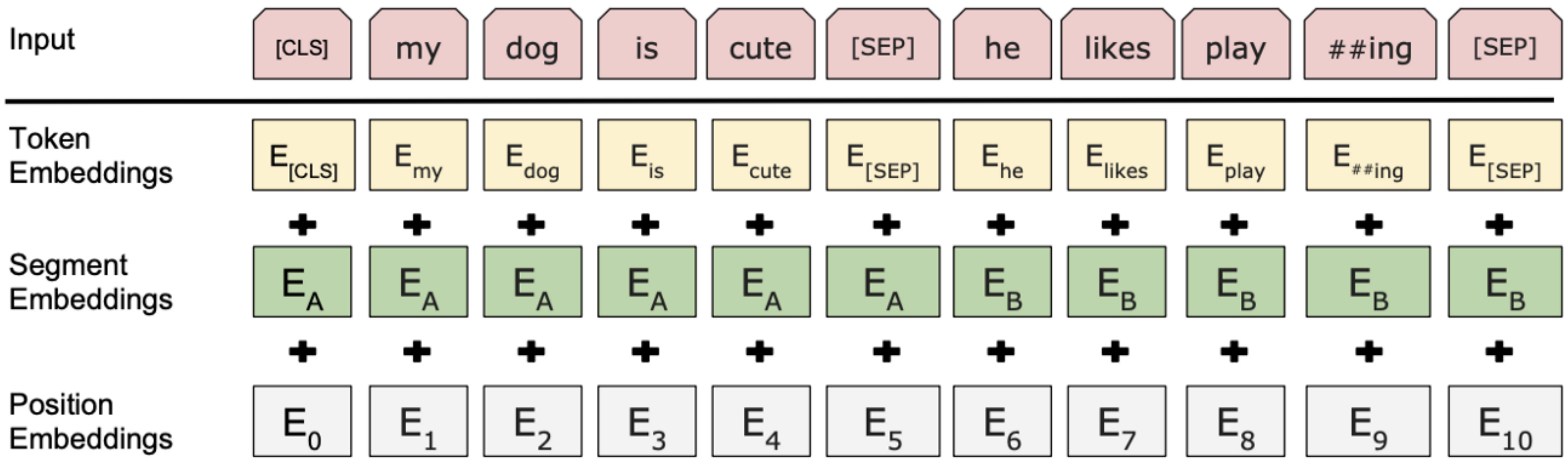
Transformers 2/3

- BERT (encoder only) got SOTA in most NLP task with
  - New pre-training (**masking**, next sentence)
  - Left and right **context** from the word
- The LM learns relationships among tokens at multiple levels
  - Grammar/Syntax
  - Semantic



Transformers 3/3

- Token embeddings are complemented with more information
- Position is key as a transformer is not a RNN
  - sequential nature of RNNs precludes parallelization within training examples



More on Transformers from Immanuel Trummer in VLDB **Tutorial 9** (Thu 8th 10:30 - 12:00)
From BERT to GPT-3 Codex: Harnessing the Potential of Very Large Language Models for Data Management

How does it work for Tabular Data?

- LMs are state-of-the-art for NL but tabular data has different forms (relational tables, spreadsheets, entity tables, …) and different relationships
  - E.g., Position, co-occurrence vs same-row, same-column

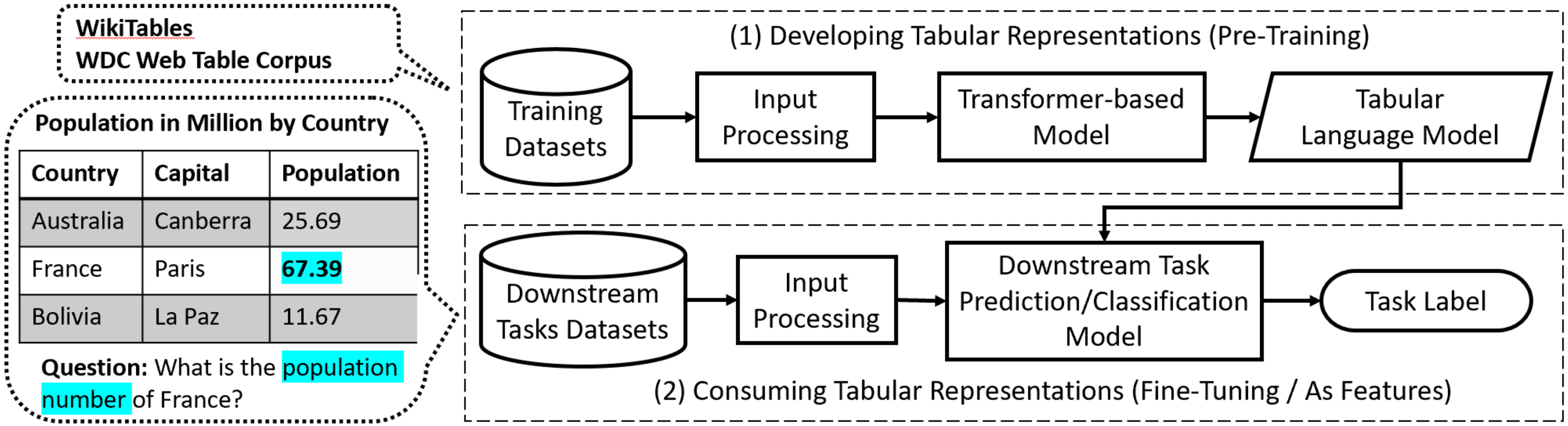
- Problem: develop LMs that model tabular data
  - How to change the transformer architecture to account for the 2D characteristics of tables and its relationships?

Questions?

Tutorial Outline

- Motivation
  - Data-centric Applications and Natural Language
- Transformers and Language Models
- Developing & Consuming Tabular Data Representation
  - Training Datasets
  - Input Processing
  - Model Training & Architecture
  - Tabular Language Model
  - Consuming Tabular LMs
- Open Challenges

Characterization Study



Dimensions

1. Training Datasets
2. Input Processing
   - Data retrieval and filtering
   - Table serialization
   - Context and table concatenation
3. Model Architecture and Training
4. Output Model Representation: Tabular Language Model
5. Fine-tuning Representation for Downstream Tasks

Training Datasets

Training Datasets

- Large number of tables along with their context are used for pre-training
  - Better representation, less bias
- Context represents additional textual data that comes with tables
  - Text describing the table: caption, title or document surrounding the table
  - Table metadata: table orientation, header, keys
  - Question and claims addressed by the table
- Two types of datasets:
  - **Unlabeled**, such as Wikipedia Tables, are used exclusively for pre-training
  - **Labeled**, such as SPIDER (Yu et al., 2018), can also be used for fine-tuning

GDP by Country in Trillions USD

Country	Capital	GDP
Germany	Berlin	3.806
France	Paris	2.603
Australia	Canberra	1.331

Question: What is the GDP of Germany?
Table: GDP by Country in Trillions USD
(**Answer:** 3.806)

Summary of Training Datasets: exclusively for pre-training (not labeled)

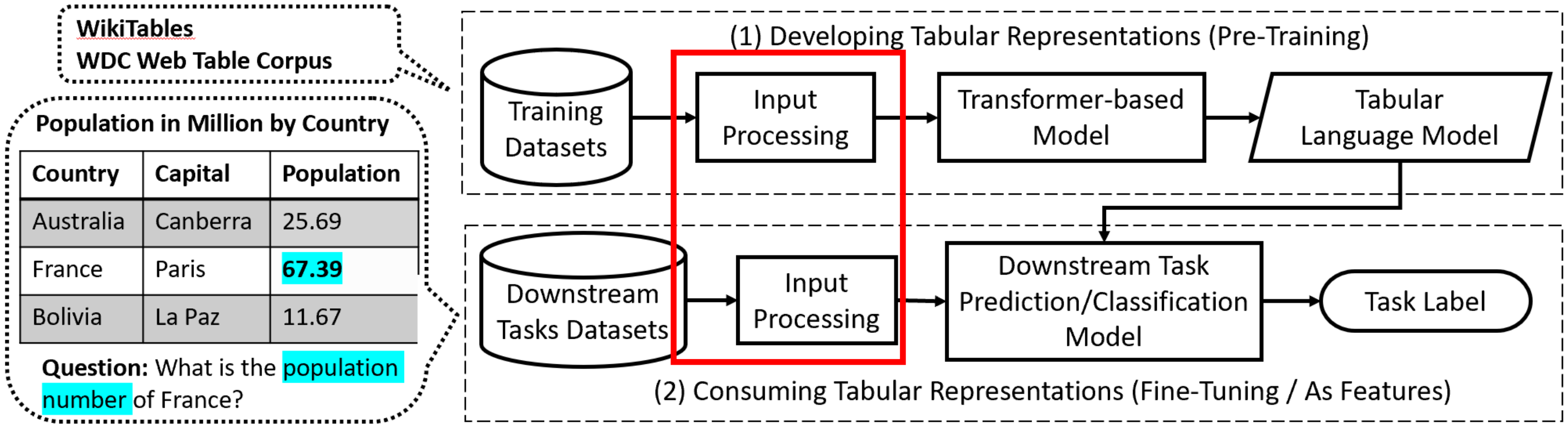
Dataset	Reference	Task Categories						Number of Tables	Large Tables	Context	Application Example
		TFC	QA	SP	TR	TMP	DI				
Wikipedia Tables	Wikipedia		✔	✔			✔	-	✔	**Surrounding Text:** table caption, page title, page description, segment title, text of the segment. **Table Metadata:** statistics about number of headings, rows, columns, data rows.	TAPAS
WDC Web Table Corpus	(Lehmberg et al., 2016)		✔	✔				233M	✔	**Table Metadata:** Table orientation, header row, key column, timestamp before and after table. **Surrounding Text:** table caption, text before and after table, title of HTML page.	TABERT
VizNet	(Hu et al., 2019)						✔	1M	✘	**Table Metadata:** Column Types.	TABBIE
Spreadsheets	(Dong et al., 2019)					✔		3,410	✘	**Table Metadata:** Cell Roles (Index, Index Name, Value Name, Aggregation and Others).	TABULARNET

Mostly Fine-Tuning Datasets (1/2)

Dataset	Reference	Task Categories						Number of Tables	Large Tables	Context	Application Example
		TFC	QA	SP	TR	TMP	DI				
NQ-Tables	(Herzig et al., 2021)		✔					169,898	✔	**Questions:** 12K.	DTR
TABFACT	(Chen et al., 2020a)	✔						16K	✘	**Textual Claims:** 118K claims.	DECO
WikiSQL	(Zhong et al., 2017)		✔	✔	✔			24,241	✘	**Questions:** 80,654.	MMR
TabMCQ	(Jauhar et al., 2016)		✔		✔			68	✘	**Questions:** 9,092.	RCI
SPIDER	(Yu et al., 2018)			✔				200 databases	✘	**Questions:** 10,181 Queries: 5,693.	GRAPPA
WikiTable Question (WikiTQ)	(Pasupat and Liang, 2015)		✔	✔				2,108	✘	**Questions:** 22,033.	TAPEX

Mostly Fine-Tuning Datasets (2/2)

Dataset	Reference	Task Categories						Number of Tables	Large Tables	Context	Application Example
		TFC	QA	SP	TR	TMP	DI				
Natural Questions (NQ)	(Kwiatkowski et al., 2019)			✔				169,898	✔	**Questions:** 320K.	MMR
OTT-QA	(Chen et al., 2021)		✔		✔			400K	✔	**Surrounding Text:** page title, section title, section text limited to 12 first sentences. **Questions:** 45,841.	MMR
Web Query Table	(Sun et al., 2019)				✔			273,816	✘	**Surrounding Text:** captions. **Queries:** 21,113.	GTR
HybridQA	(Chen et al., 2020b)		✔					13K	✘	**Questions:** 72K. **Surrounding Text:** first 12 sentences surrounding the table.	MATE
FEVEROUS	(Aly et al., 2021)	✔						28.8K	✘	**Claims:** 87K. **Surrounding Text:** article title. **Table Metadata:** row and column headers.	MATE



Input Processing

Data Retrieval and Filtering

Table Serialization: Reshaping 2D tabular structure to 1D

Context and Table Concatenation

Data Retrieval and Filtering

- Why do we need it?
  - Meet the limit (typically of 512 tokens) of Transformers
    - Transformers architecture theoretically has no limits on the input size
    - However, practically it is not the case: limit derived from positional embeddings, fixed attention size and computational complexity
  - Improve training time
  - Eliminate potential noise in output representations

(Devlin et al., 2019; Yin et al., 2020; Liu et al. 2021a)

Data Retrieval and Filtering

- How?
  - Can be downstream task by itself, Table Retrieval
  - Using a ranking function like BM25 (Robertson et al., 1995)
  - Using content snapshot (TABERT (Yin et al., 2020))
  - Term Frequency Inverse Document Frequency (TFIDF) (RCI (Glass et al., 2021))
  - Setting a threshold to limit the number of columns/rows allowed (DRT (Thorne et al., 2021)
  - Splitting Tables into smaller chunks (TUTA (Wang et al., 2021b), TabularNet (Du et al., 2021))

In which city did Piotr's last 1st place finish occur?

	Year	Venue	Position	Event
R_1	2003	Tampere	3rd	EU Junior Championship
R_2	2005	Erfurt	1st	EU U23 Championship
R_3	2005	Izmir	1st	Universiade
R_4	2006	Moscow	2nd	World Indoor Championship
R_5	2007	Bangkok	1st	Universiade

Selected Rows as Content Snapshot : $\{R_2, R_3, R_5\}$

Country	Capital	Population
Australia	Canberra	25.69
France	Paris	67.39
Bolivia	La Paz	11.67

Keeping 2 columns

Country	Population
Australia	25.69
France	67.39
Bolivia	11.67

Country	Capital
Australia	Canberra
France	Paris
Bolivia	La Paz

Keeping 2 rows

Country	Capital	Population
Australia	Canberra	25.69

Country	Capital	Population
France	Paris	67.39

Country	Capital	Population
Bolivia	La Paz	11.67

Population in Million by Country

Country	Capital	Population
Australia	Canberra	25.69
France	Paris	**67.39**
Bolivia	La Paz	11.67

Question: What is the population number of France?

Table Serialization

- Four possible ways:
  1- Horizontal scanning of the table row by row
  - Flattened table with value separators
    - Country | Capital | Population | Australia | Canberra | 25.69 ... Bolivia | La Paz | 11.67
  - Flattened table with **special token separator** to indicate beginning of a new row, new cell, new header (TAPEX (Liu et al. 2021a), TUTA (Wang et al. 2021b), ForTaP (Cheng et al., 2021))
    - Country | Capital| Population **[SEP]** Australia | Canberra | 25.69 … **[SEP]** Bolivia | La Paz | 11.67
  - Flattened table where each cell is represented as a concatenation of the **column name**, **column type** and cell value (TABERT)
    - **Country: varchar:** Australia | Capital: varchar: Canberra | Population: float: 25.69 ... Country: varchar: Italy | Capital: varchar: Rome | Population: float: 59.55
  - Flattened **column headers** only (GRAPPA (Yu et al., 2021))
    - Country|Capital|Population

Table Serialization (continued)

2- Vertical scanning of the table column by column

- Simple concatenation of column values or by using special separator tokens (DODUO (Suhara et al. , 2021)

3- Combining the output from both horizontal and vertical serialization

- element-wise product (RCI (Glass et al., 2021)), CLTR (Pan et al., 2021),
- average pooling and concatenation (TabularNet),
- average of row and column embeddings (TABBIE (Iida et al., 2021)).

4- Transforming data to text

- using meaningful sentences generated out of the tabular data (DRT (Thorne et al., 2021)
- using table-to-text systems such as Totto (Parikh et al., 2020).

Name	Profession	Location
Nicholas	Doctor	Washington D.C.
Sarah	Doctor	NY

Name	Birth City	Birth Year
Sheryl	…	1978
Sarah	Chicago	1982
Teuvo	Ruskala	1912

Husband Name	Wife Name	Marriage Year
Nicholas	Sheryl	…
John	Sarah	2010

- Nicholas lives in Washington D.C. with his wife.
- Sheryl is Nicholas's wife.
- Teuvo was born in 1912 in Ruskala.
- Sheryl's mother gave birth to her in 1978.
- Nicholas is a doctor.
- Sarah was born in Chicago in 1982.
- Sarah married John in 2010.
- Sarah works in a hospital in NY as a doctor.

DRT (Thorne et al., 2021)

Table Serialization: Which method to choose?

- Most of the systems do not compare the different approaches:
  - One approach is typically selected and followed
- TABERT reports experiments with different table linearization strategies:
  - adding type information and cell values
  - phrasing the input as a sentence such as in TABFACT (Chen et al., 2020a)

  => Improvement in results
- (Veltri et al., 2022) in a table to text generation task experimented row vs column serialization:
  - Row performed better

Context and Table Concatenation

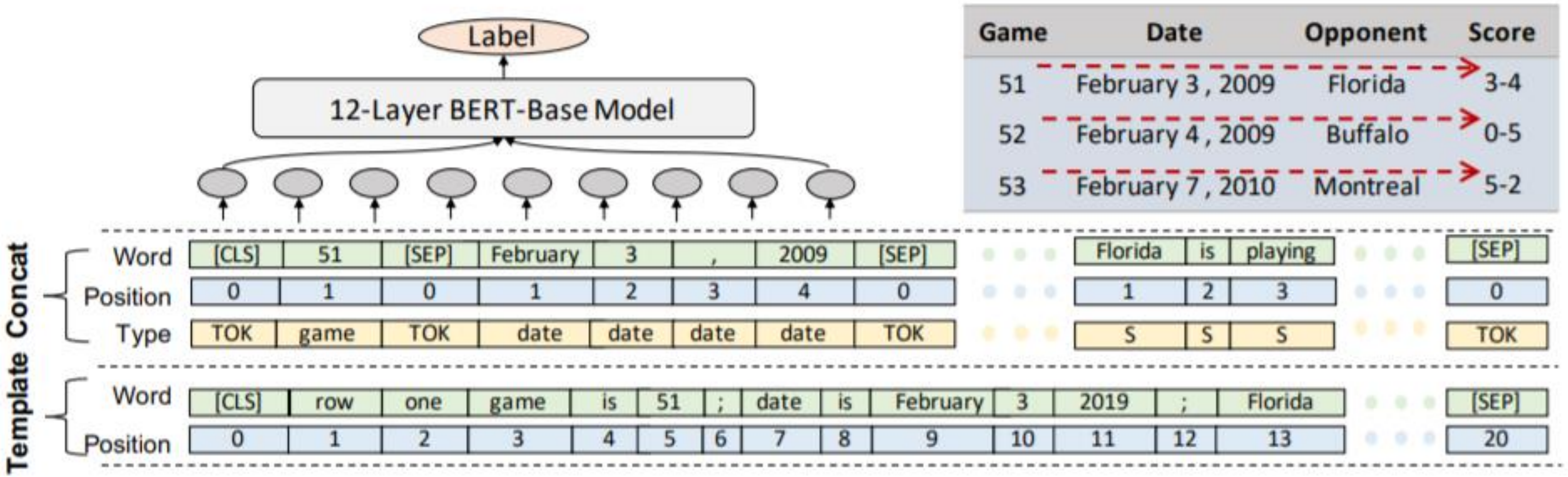
- **Context** is either **prepended** or **appended** to the serialized table.
- Common case is to be prepended to the serialized table
- TabFACT tested both strategies:
  - **no** significant difference in performance
- Type of context added usually depends on target downstream application
  - QA: a question is prepended to the serialized table.
- Some works like RCI (Glass et al., 2021) encode the context and the serialized table separately

- Some works, like TABBIE (Iida et al., 2021), Doduo, TabularNet, do not include context
  - Due to nature of downstream tasks specifically **TMP** and **DI**

Context and Table parsed by row:
[CLS] *Population in Million by Country* [CLS] Country | Capital | Population [SEP] France | Paris | 67.39 ... [SEP] Italy | Rome | 59.55

Context and Table parsed by column:
[CLS] *Population of Countries* [CLS] Country | France | ... | Italy | ... [SEP] Capital | France | ... | Rome | ... [SEP] Population | 67.39 | ...

TabFact experimented different serializations



Model	Val	Test	Test (simple)	Test (complex)
BERT classifier w/o Table	50.9	50.5	51.0	50.1
Table-BERT-Horizontal-F+T-Concatenate	50.7	50.4	50.8	50.0
Table-BERT-Vertical-F+T-Template	56.7	56.2	59.8	55.0
Table-BERT-Vertical-T+F-Template	56.7	57.0	60.6	54.3
Table-BERT-Horizontal-F+T-Template	66.0	65.1	79.0	58.1
Table-BERT-Horizontal-T+F-Template	**66.1**	**65.1**	**79.1**	**58.2**

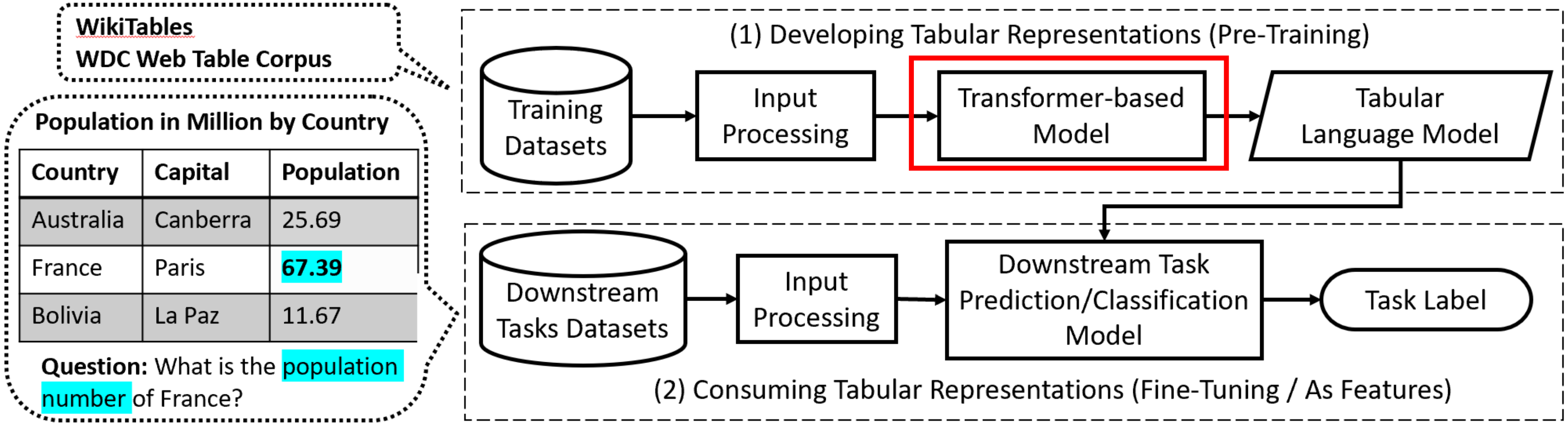
TabFact

Questions?

Model Training & Architecture

Customizations to account for tabular data structure

Extensions at the input/output level and/or on the internals of the architecture

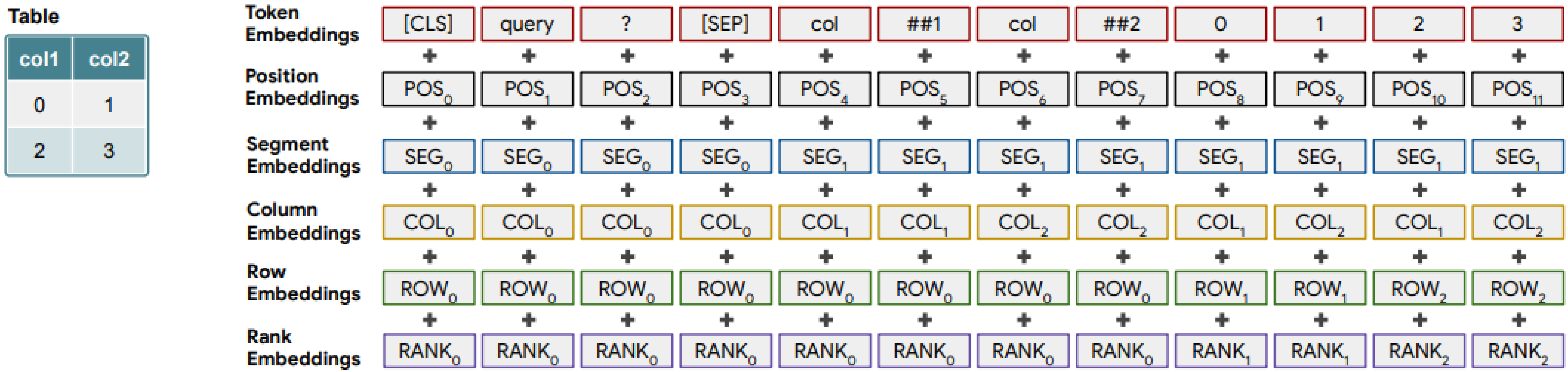


Adaptations of Transformers’ Architecture

- Model with tabular data structure aware **=>** Customization to Vanilla transformer-based LMs
- Extensions are at different levels:
  - Input
  - Internal
  - Output
  - Training procedure

Input Level

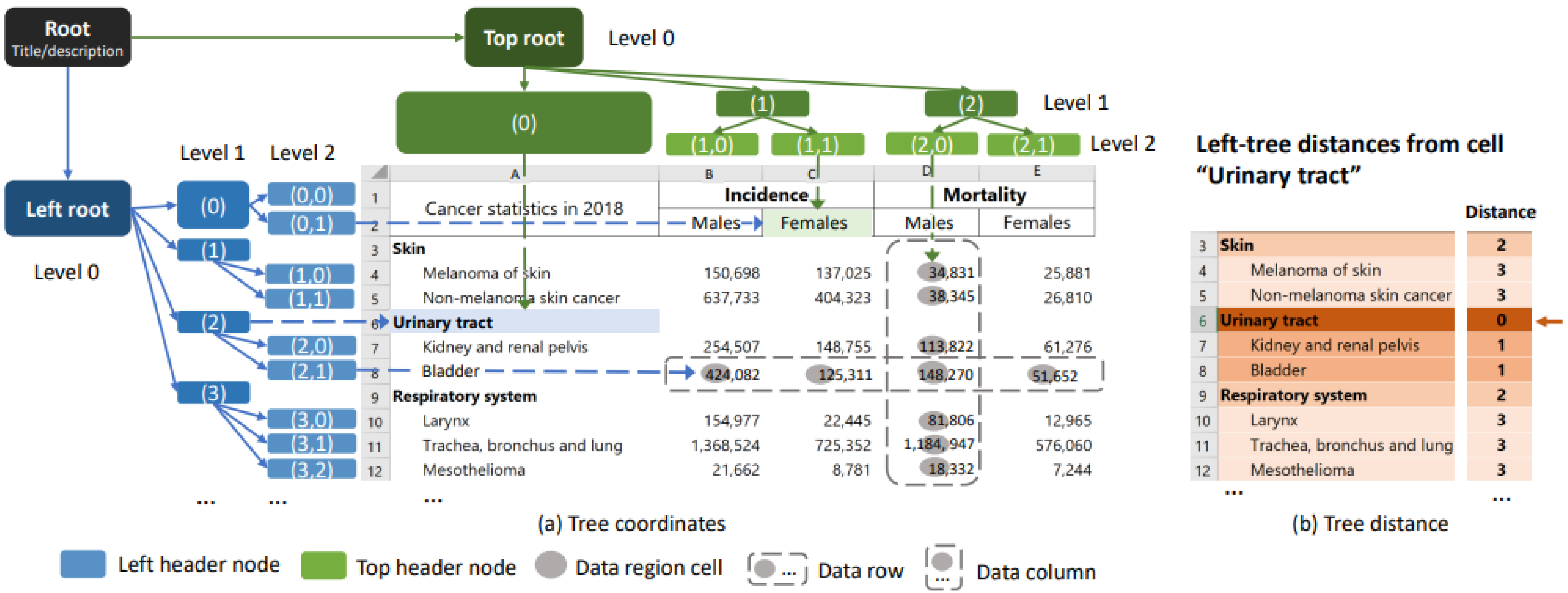
- Additional positional embeddings to explicitly model the table structure
  - Typically for relational tables
  - Example position of the cell (row and column IDs), segment id: whether it is a context or a table entry, relative positional information of a token in cell/column header and rank id for sorting floats and dates



(TAPAS (Herzig et al., 2020)

Input Level (continued)

- Tree-based positional embeddings (TUTA (Wang et al, 2021))
  - Typically for entity tables or spreadsheets
  - Encode the position of a cell using top and left embeddings of a bi-dimensional coordinate tree.

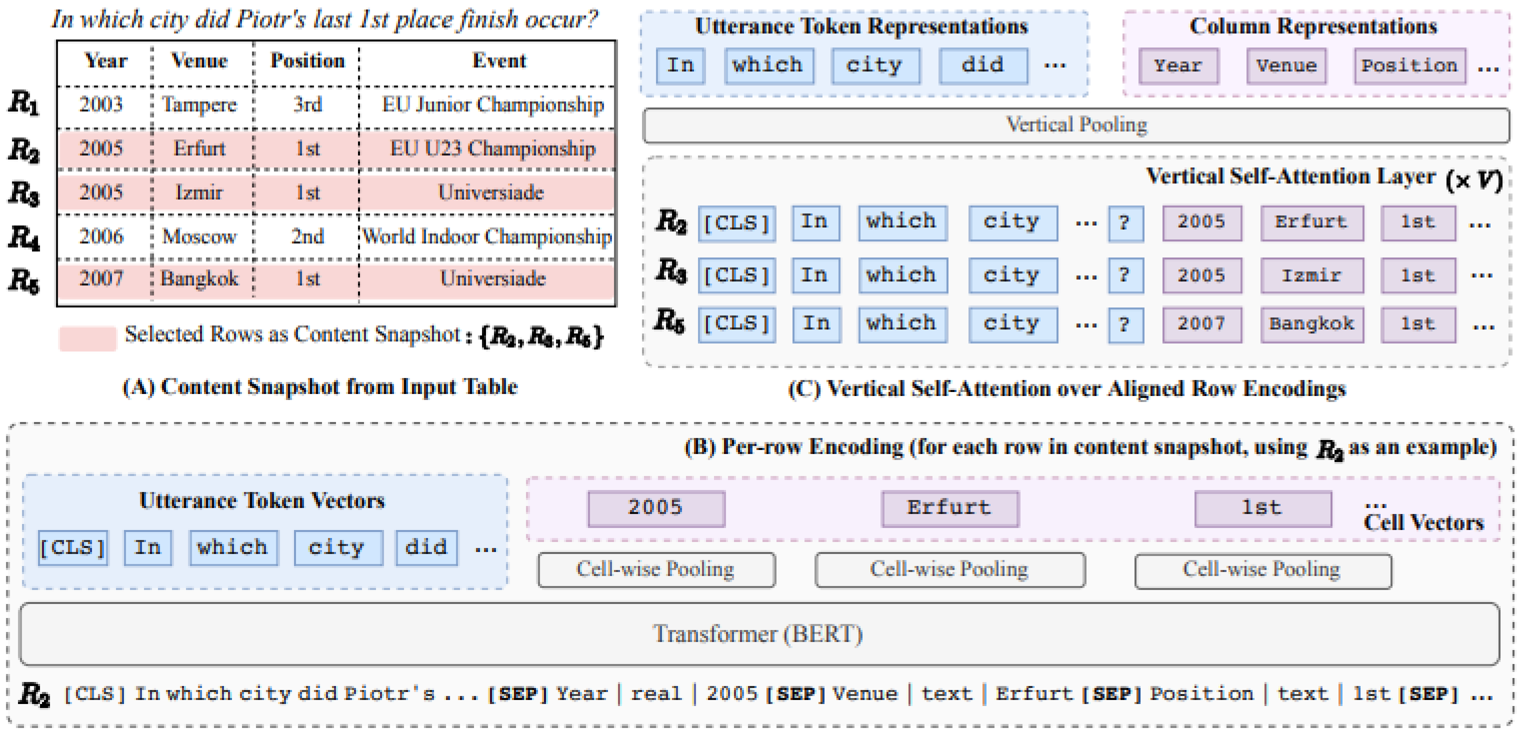


(a) Tree coordinates

(b) Tree distance

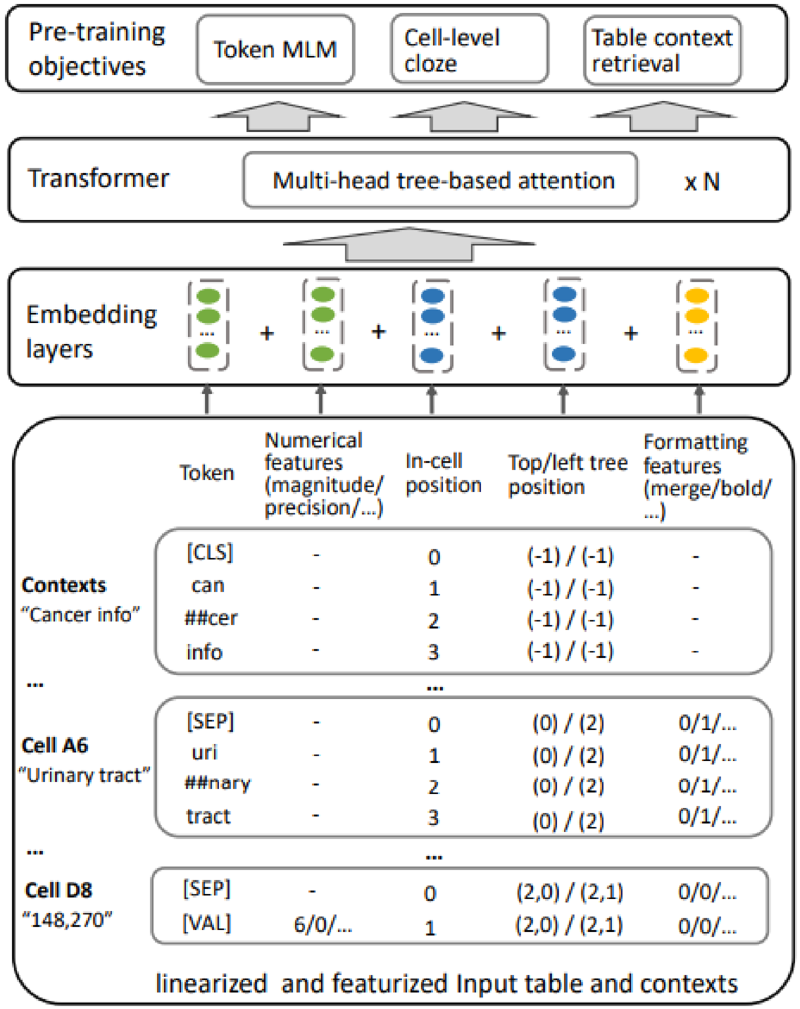
Internal Level

- The attention module is the mostly concerned with updates at the internal level
- **Vertical self-attention** layers capture cross-row dependencies on cell values (TABERT)



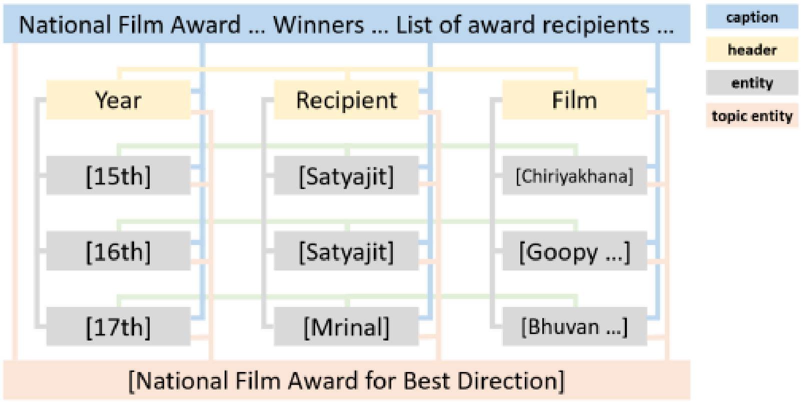
Internal Level

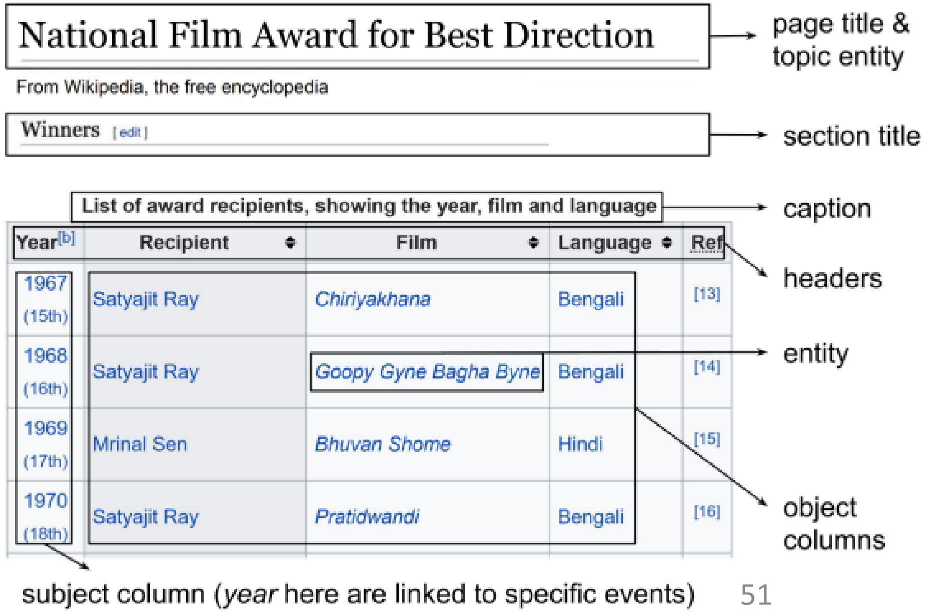
- **Tree-based attention (TA):** row-wise or column-wise attention instead of additional positional embeddings (TUTA (Wang et al, 2021))
  - Tree-structure injected using a symmetric binary matrix to indicate visibility between tokens
  - Based on ablation study results:
    - TA improved accuracy for both cell-type classification and table-type classification compared to basic attention mechanism where all cells are visible to each other.



Internal Level

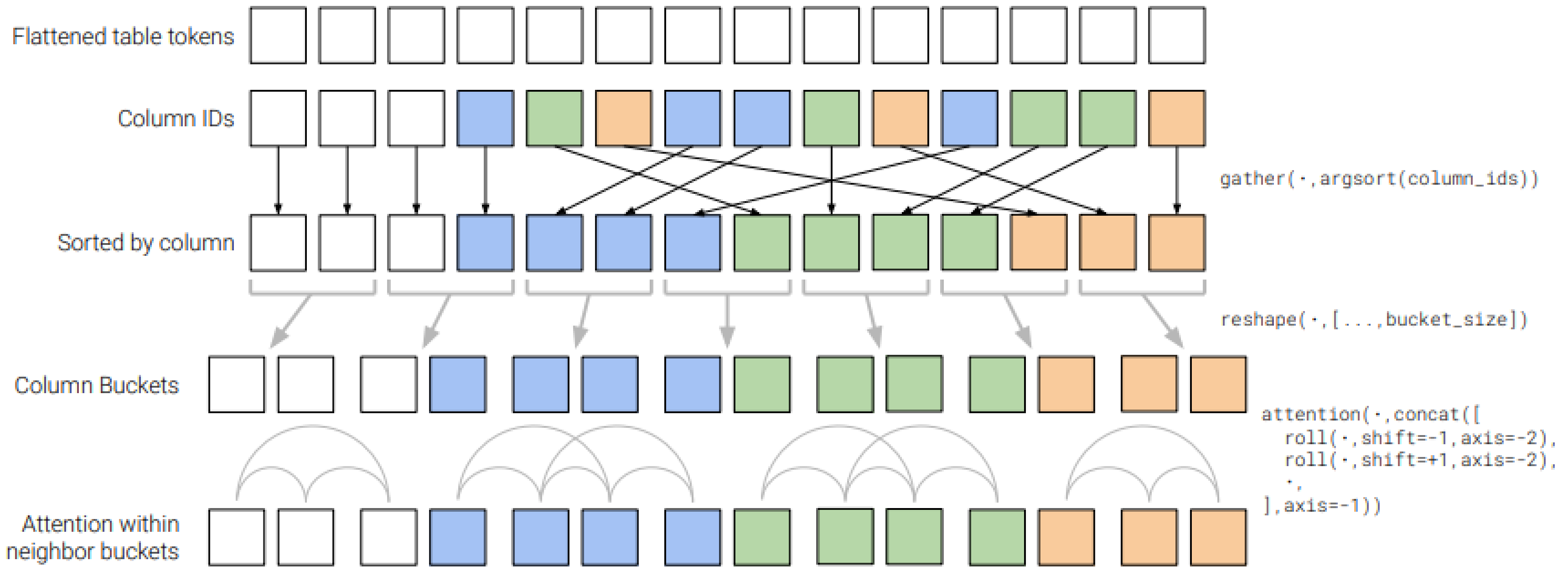
- **Masked self-attention** module attends to structurally related elements (TURL (Deng et al., 2021)):
  - Elements in one row or in one column (using a visibility matrix)
  - Different than vanilla transformer where each element attends to all other elements
  - Helps model to capture:
    - Factual knowledge embedded in the table
    - Associations between table metadata and table content





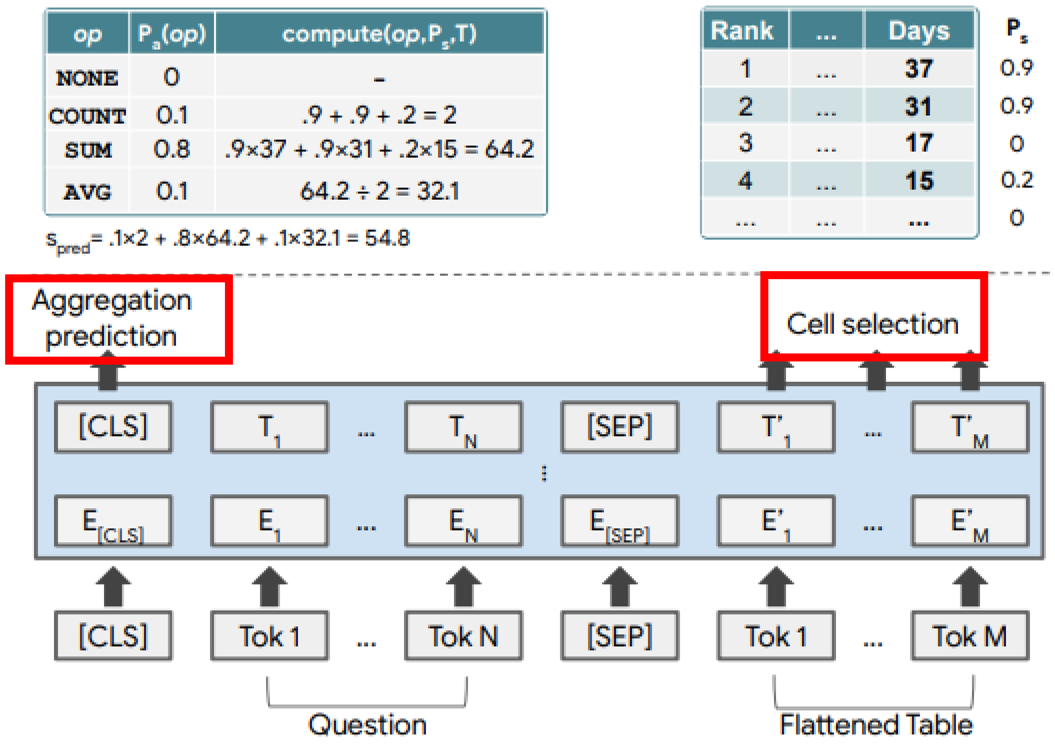
Internal Level

- **Sparse attention** method is used to address the limit of input size of transformers((512 tokens)
  - MATE sparsifies the attention matrix to attend to either rows and columns



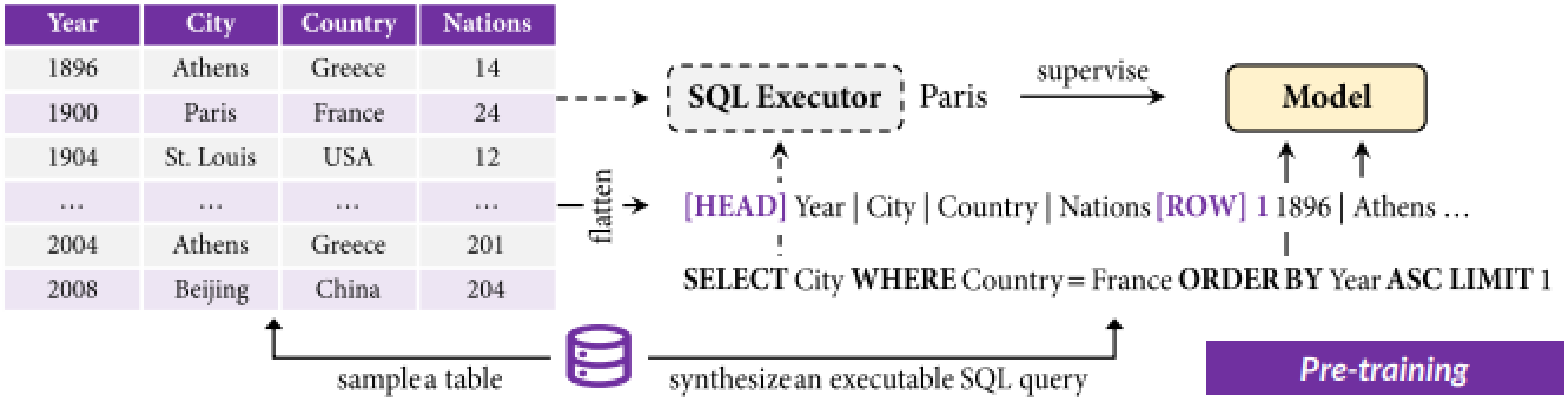
Output Level

- Additional layers added on top of the feed-forward networks (FFNs) of the LM based on the targeted downstream task
- Question Answering (QA):
  - Additional classification layer for aggregations and cell selection (TAPAS)



Training Procedure Level

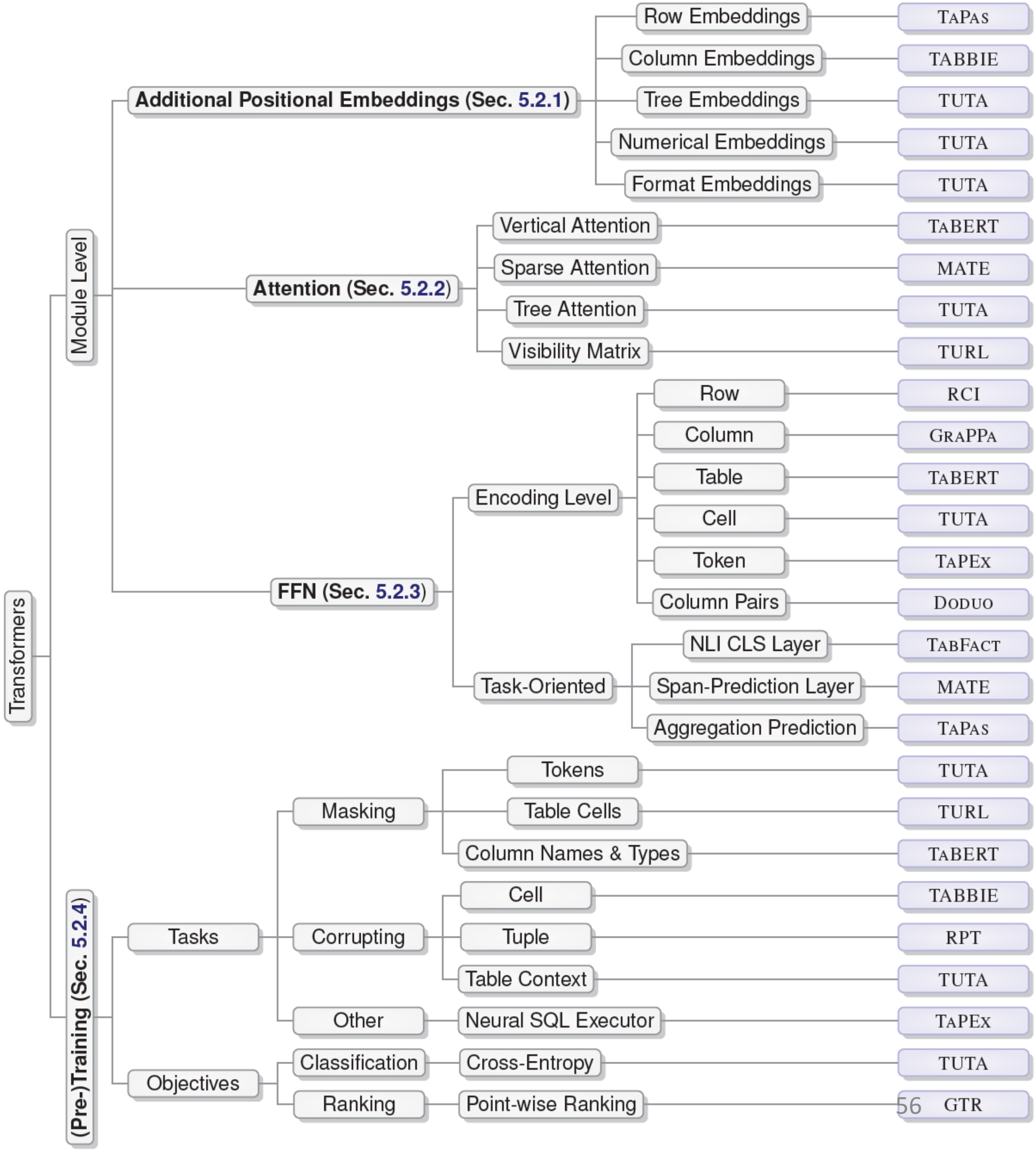
- Pretraining Task:
  - Prior to fine-tuning
  - Typically consist of reconstruction tasks, i.e., reconstruct correct input out of corrupted one
    - Usually using cross-entropy loss as objective function
  - Modifications on the typical MLM are applied to take into consideration the tabular structure:
    - Masking tokens from cells
    - Masking the whole cell regardless of the number of tokens it has
      - Enables the model to integrate the factual knowledge embedded in the table content and its context
    - Masking columns names and data types
  - Occasionally an SQL engine is used (TAPEX) to train the model to act as a neural SQL executor
    - Enable to mimic SQL semantics with relational tables


Year
City
Country
Nations
1896
Athens
Greece
14
1900
Paris
France
24
1904
St. Louis
USA
12
…
…
…
…
2004
Athens
Greece
201
2008
Beijing
China
204
SQL Executor
Paris
supervise
Model
flatten
[HEAD] Year | City | Country | Nations [ROW] 1 1896 | Athens …
SELECT City WHERE Country = France ORDER BY Year ASC LIMIT 1
sample a table
synthesize an executable SQL query
Pre-training

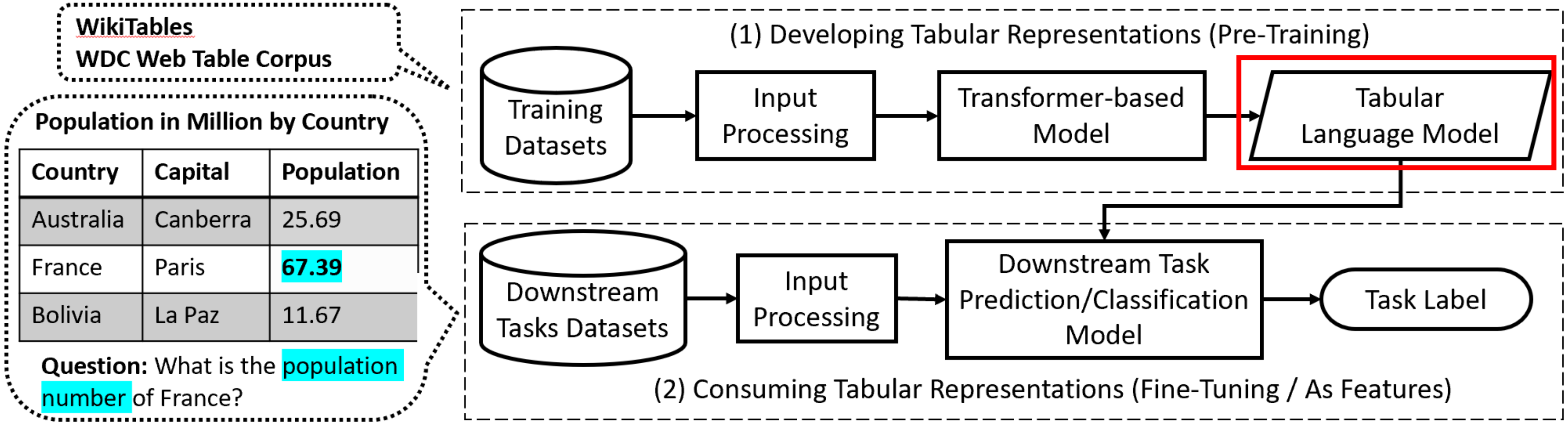
(TAPEX)

Summary of customization

- A more table structure aware LM requires modifications:
  - Input level through additional embeddings
  - Internal level through adjustment of attention module
  - Training procedure level through pre-training task and objective that are table related such as masking and reconstructing cells
  - Output level through additional classification layers that are task dependent



Questions?



Tabular Language Model

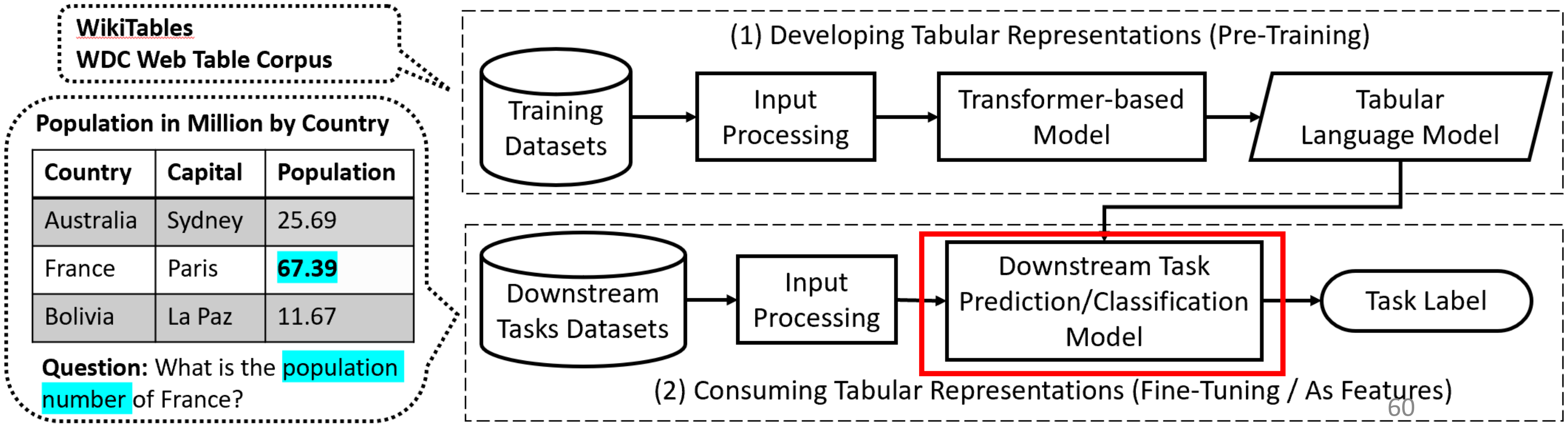
Output data representation and granularity

Tabular Language Model

- As a result of (1), pre-trained tabular language model is developed
- Two major ways to be utilized:
  - Build on top of the encoder with more modules and fine-tuning
  - Use the encoder as part of a bigger architecture in a more task-oriented fashion rather than encoder-oriented (as embeddings feature)
- This model can be fine-tuned to learn the specifics of a downstream task, or it can be used as is with standard supervised machine learning algorithms
- Output representations can be extracted at different granularities:
  - Token
  - Cell
  - Row
  - Column
  - Column pairs (Doduo)
  - Table
  - Table pairs (Deco)
- While token and cell are the most common, the granularity is highly dependent on the target task
  - E.g.: Table representation for TR task

Consuming Tabular Language Models

(2) In

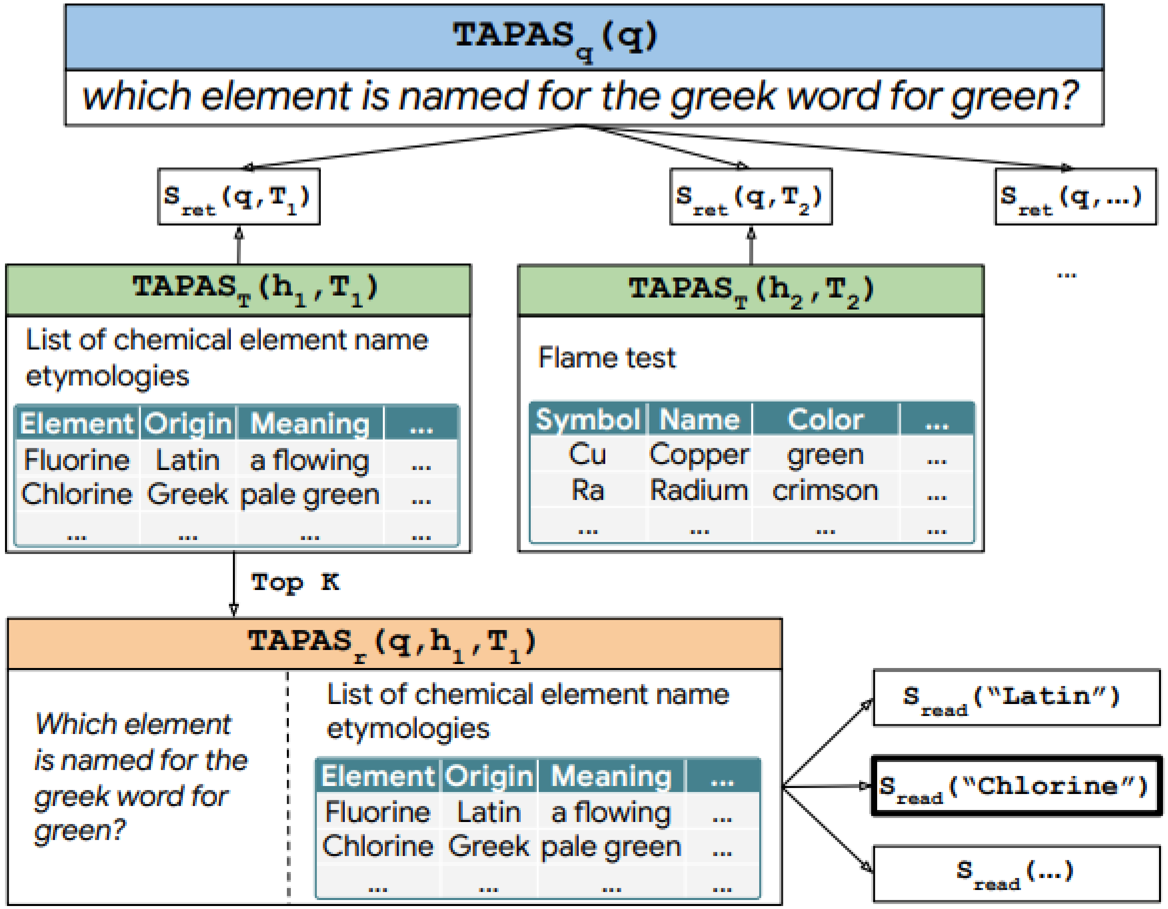


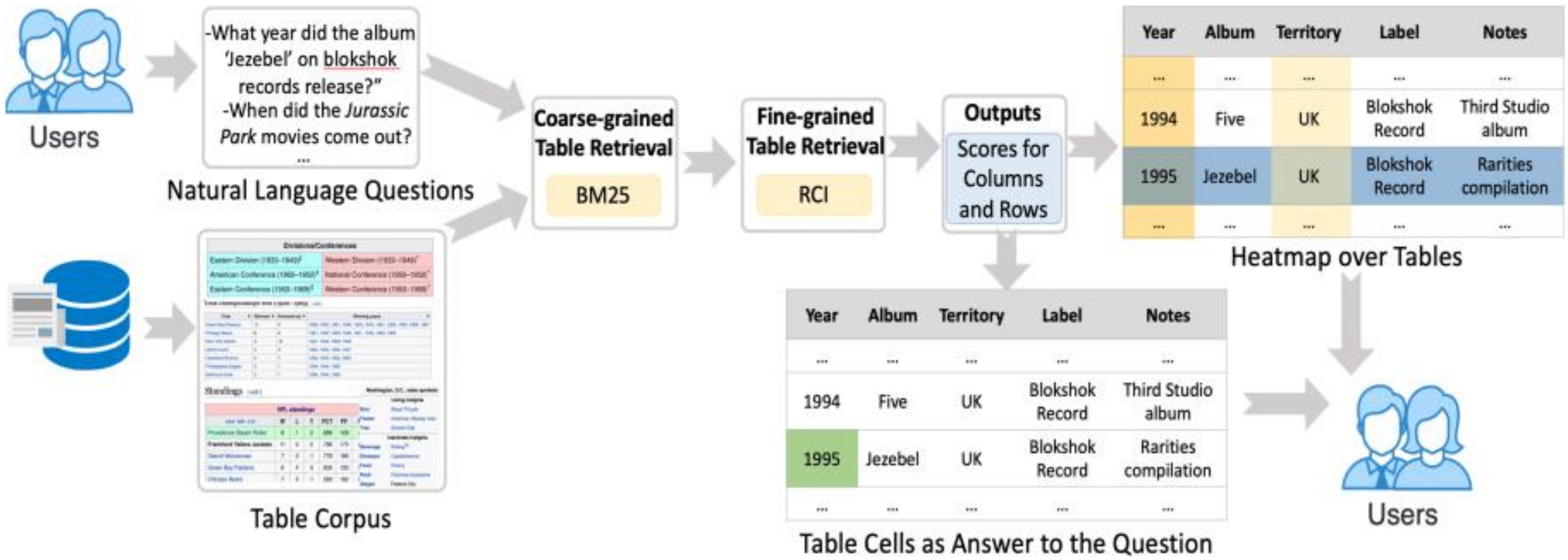
Prediction/Classification Systems

- Pre-trained transformer-based LM act as encoders of the input and typically:
  - Used as building block in a bigger system
  - Additional layers are added on top and the entire model is fine-tuned for a specific downstream task

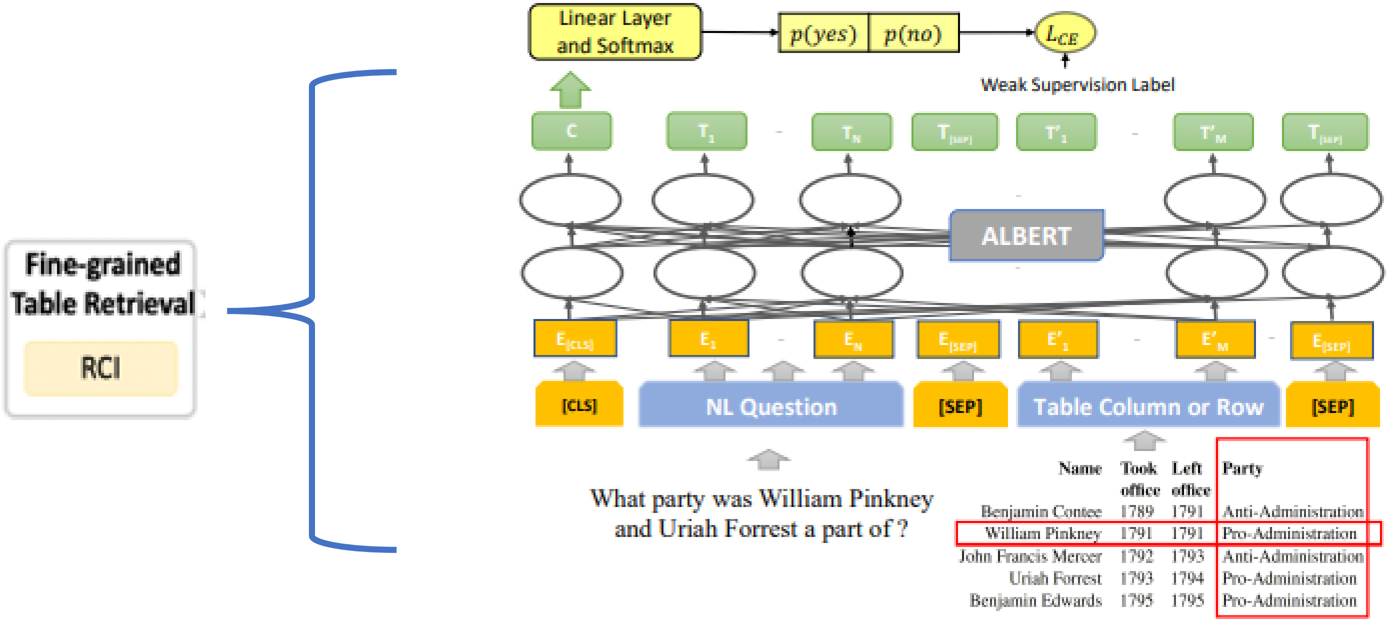
Prediction/Classification Systems

- Sometimes LM are:
  - Employed as components of bigger system whose aim is to develop end-to-end trained system towards a certain task
  - Examples:
    - DTR (Herzig et al., 2021) compute a similarity score between embeddings of question and embedding of table
    - CLTR classifies whether an associated row/column with a given question contains the answer





CLTR (Pan et al., 2021)



Tutorial Outline

- Motivation
  - Natural Language and Data-centric Applications
- Language Models and Transformers
- Developing & Consuming Tabular Data Representation
  - Training Datasets
  - Input Processing
  - Model Training & Architecture
  - Tabular Language Model
  - Consuming Tabular LMs
- Open Challenges

Complex Queries and Rich Tables

- Few systems support aggregation operations such as max, min, avg
- No support for joins
- No support for dependencies
- No support for heterogeneity
    - E.g., columns with different measurement units such as adding kgs and lbs

Model Efficiency

- Transformers suffer from the upper bound limit of 512 tokens
  - Problem for large tables
- Multiple techniques to improve computation and memory usage
  - Locality sensitive hashing to replace attention
  - Approximate self-attention by a low-rank matrix
- New methods to make transformers more efficient for long context
  - only done on free text and not tabular data

(Treviso et al, 2022; Zaheer et al, 2020)

Benchmarking Data Representations

- No benchmark datasets to establish baselines for tabular language models
- Current evaluation is extrinsic
  - Only considers the performance of the language model on the downstream tasks
- There is a need for intrinsic evaluation to evaluate the quality of those tabular representations
  - Checklist: generation of general linguistic capabilities and test types
  - We can design tests that evaluate properties of **rows/columns/dependencies**

(Ribeiro et al, 2020; Cappuzzo et al 2020)

Green Tabular LMs

- Large-scale transformers with billion of parameters requires heavy computation: several days of GPUs/TPUs for training
  - Contributes to global warming
- Need for new techniques that limit carbon footprint of tabular LMs without decrease in performance of downstream tasks
- One direction: reduce training data by removing redundant or less informative cells, tuples, tables
  - How to identify such data is a key challenge

(Yang et al 2009)

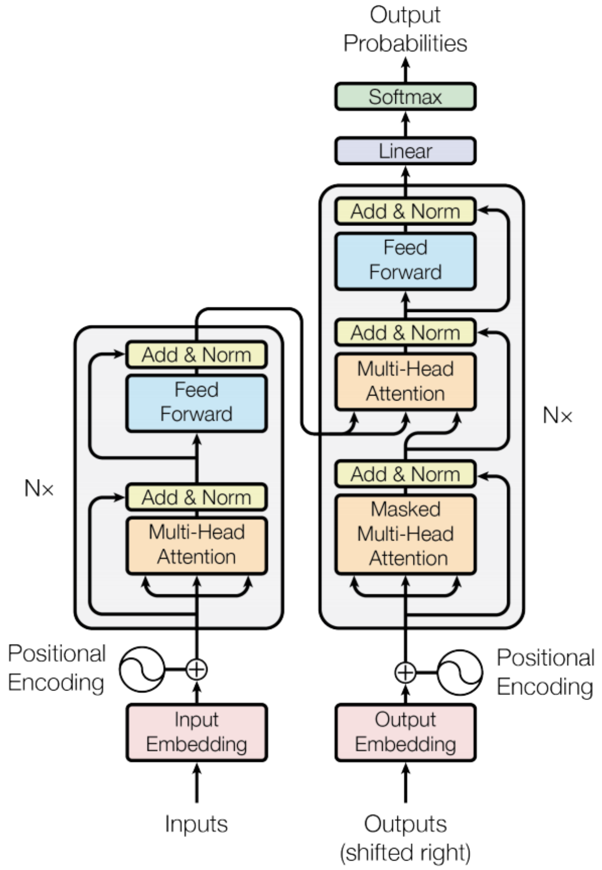
More general challenges

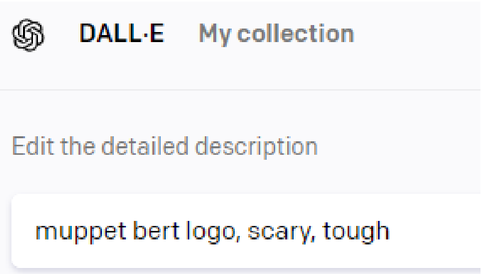
- Data bias
  - NLP LMs incorporate stereotypes + race, gender bias in the model parameters
    - Bias inherited from the dataset used for training the models
  - Reduce bias by preprocessing training dataset or postprocessing LMs
- Interpretability
  - How to justify the final output for a given task?
    - E.g., provide the cells that led to a given output (True/False)
  - Look at attention weights wrt input tokens to capture their influence on output
- Error Analysis
  - Most systems report only evaluation scores (p, r, accuracy)
  - no explanations for the cases where the model fails
    - for a QA task with a set of wrong answers, a pattern could explain misclassification
      - E.g., two column names having an overlap of more than 5 characters

Conclusions

Representation learning for tabular data

Task ID	Task Label	Tasks Coverage	Input NL NL	Output
TFC	Table-based Fact-Checking or Entailment	Fact-Checking Text Refusal/Entailment	Table + Claim	True/False Refused/Entailed (Data Evidence)
QA	Question Answering	Retrieving the Cells for the Answer	Table + Question	Answer Cells
SP	Semantic Parsing	Text-to-SQL	Table + NL Query	Formal QL
TR	Table Retrieval	Retrieving Table that Contains the Answer	Tables + Question	Relevant Table(s)
TMP	Table Metadata Prediction	Column Type Prediction Table Type Classification Header Detection Cell Role Classification Column Relation Annotation Column Name Prediction	Table	Column Types Table Types Header Row Cell Role Relation between Two Cols Column Name
DI	Data Imputation	Cell Content Population	Table with Corrupted Cell Values	Table with Complete Cell Values





Questions?

References

- (Aly et al, 2021) Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. FEVEROUS: Fact Extraction and VERification Over Unstructured and Structured information. In NeurIPS.
- Gilbert Badaro, Ramy Baly, Hazem Hajj, Wassim El-Hajj, Khaled Bashir Shaban, Nizar Habash, Ahmad Al-Sallab, and Ali Hamdi. 2019. A survey of opinion mining in Arabic: A comprehensive system perspective covering challenges and advances in tools, resources, models, applications, and visualizations. ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP) 18, 3 (2019), 1–52.
- Gilbert Badaro, Hazem Hajj, and Nizar Habash. 2020. A link prediction approach for accurately mapping a large-scale Arabic lexical resource to English WordNet. ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP) 19, 6 (2020), 1–38.
- Gilbert Badaro, Mohammed Saeed, and Paolo Papotti. 2021. Transformers for Tabular Data Representation: A survey of models and applications. EURECOM Technical Report, October 2021: https://www.eurecom.fr/publication/6721.
- Chandra Sekhar Bhagavatula, Thanapon Noraset, and Doug Downey. 2015. TabEL: Entity linking in web tables. In International Semantic Web Conference. Springer, 425–441.
- (Cappuzzo et al, 2020) Riccardo Cappuzzo, Paolo Papotti, and Saravanan Thirumuruganathan. 2020. Creating embeddings of heterogeneous relational datasets for data integration tasks. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data. 1335–1349.
- (Chen et al., 2020a) Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020. TabFact: A Largescale Dataset for Table-based Fact Verification. In International Conference on Learning Representations. https://openreview.net/forum?id=rkeJRhNYDH
- (Deng et al, 2020) Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. 2020. TURL: Table understanding through representation learning. Proceedings of the VLDB Endowment 14, 3 (2020), 307–319.
- (Devlin et al., 2019) Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In NACL: HLT. ACL, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In International Conference on Learning Representations.

References

- (Du et al., 2021) Lun Du, Fei Gao, Xu Chen, Ran Jia, Junshan Wang, Jiang Zhang, Shi Han, and Dongmei Zhang. 2021. TabularNet: A Neural Network Architecture for Understanding Semantic Structures of Tabular Data. In ACM SIGKDD. 322–331.
- (Eisonschlos et al., 2021) Julian Martin Eisenschlos, Maharshi Gor, Thomas Müller, and William W Cohen. 2021. MATE: Multi-view Attention for Table Transformer Efficiency. arXiv preprint arXiv:2109.04312 (2021).
- (Eisenschlos et al, 2020) Julian Eisenschlos, Syrine Krichene, and Thomas Müller. 2020. Understanding tables with intermediate pre-training. In EMNLP 2020, pages 281–296. ACL.
- (Glass et al., 2021) Michael Glass, Mustafa Canim, Alfio Gliozzo, Saneem Chemmengath, Vishwajeet Kumar, Rishav Chakravarti, Avirup Sil, Feifei Pan, Samarth Bharadwaj, and Nicolas Rodolfo Fauceglia. 2021. Capturing Row and Column Semantics in Transformer Based Question Answering over Tables. In NACL: HLT. 1212–1224.
- (Gkini et al, 2021) Orest Gkini, Theofilos Belmpas, Georgia Koutrika, Yannis E. Ioannidis: An In-Depth Benchmarking of Text-to-SQL Systems. SIGMOD Conference 2021: 632-644
- Yuan Gong, Yu-An Chung, and James Glass. 2021. AST: Audio Spectrogram Transformer. arXiv preprint arXiv:2104.01778 (2021).
- (Herzig et al., 2021) Jonathan Herzig, Thomas Mueller, Syrine Krichene, and Julian Eisenschlos. 2021. Open Domain Question Answering over Tables via Dense Retrieval. In NACL: HLT. 512–519.
- (Herzig et al, 2020) Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Mueller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly Supervised Table Parsing via Pretraining. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 4320–4333.
- (Iida et al., 2021) Hiroshi Iida, Dung Thai, Varun Manjunatha, and Mohit Iyyer. 2021. TABBIE: Pretrained Representations of Tabular Data. In NACL: HLT. 3446–3456.
- (Karagiannis et al, 2020) Georgios Karagiannis, Mohammed Saeed, Paolo Papotti, and Immanuel Trummer. 2020. Scrutinizer: A Mixed-Initiative Approach to Large-Scale, Data-Driven Claim Verification. Proc. VLDB Endow. 13, 11 (2020), 2508–2521.
- George Katsogiannis-Meimarakis and Georgia Koutrika. 2021. A Deep Dive into Deep Learning Approaches for Text-to-SQL Systems. In Proceedings of the 2021 International Conference on Management of Data. 2846–2851.
- [20] Bogdan Kostić, Julian Risch, and Timo Möller. 2021. Multi-modal Retrieval of Tables and Texts Using Tri-encoder Models. In Proceedings of the 3rd Workshop on Machine Reading for Question Answering. ACL, Punta Cana, Dominican Republic, 82–91.

References

- (Lehmberg et al., 2016) Oliver Lehmberg, Dominique Ritze, Robert Meusel, and Christian Bizer. 2016. A large public corpus of web tables containing time and context metadata. In WWW Companion. 75–76.
- Yuliang Li, Jinfeng Li, Yoshihiko Suhara, AnHai Doan, and Wang-Chiew Tan. 2020. Deep Entity Matching with Pre-Trained Language Models. Proc. VLDB Endow. 14, 1 (2020), 50–60. https://doi.org/10.14778/3421424.3421431
- (Liu et al., 2021) Qian Liu, Bei Chen, Jiaqi Guo, Zeqi Lin, and Jian-guang Lou. 2021. TAPEX: Table pre-training via learning a neural SQL executor. arXiv preprint arXiv:2107.07653 (2021).
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. CoRR abs/1907.11692 (2019). arXiv:1907.11692 http://arxiv.org/abs/1907.11692
- (Mikolov et al, 2013) Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
- (Pan et al, 2021) Feifei Pan, Mustafa Canim, Michael Glass, Alfio Gliozzo, and Peter Fox. 2021. CLTR: An End-to-End, Transformer-Based System for Cell-Level Table Retrieval and Table Question Answering. In ACL System Demo. 202–209.
- Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. In ACL. ACL, Online, 4902–4912. https://doi.org/10.18653/v1/2020.acl-main.442
- Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. 2020. Green ai. Commun. ACM 63, 12 (2020), 54–63.
- Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2020. Energy and policy considerations for modern deep learning research. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34. 13693–13696.
- Yoshihiko Suhara, Jinfeng Li, Yuliang Li, Dan Zhang, Çağatay Demiralp, Chen Chen, and Wang-Chiew Tan. 2021. Annotating Columns with Pre-trained Language Models. arXiv preprint arXiv:2104.01785 (2021).
- (Tang et al, 2021) Nan Tang, Ju Fan, Fangyi Li, Jianhong Tu, Xiaoyong Du, Guoliang Li, Samuel Madden, and Mourad Ouzzani. 2021. RPT: Relational Pre-trained Transformer Is Almost All You Need towards Democratizing Data Preparation. Proc. VLDB Endow. 14, 8 (2021), 1254–1261.

References

- (Thorne et al., 2021) James Thorne, Majid Yazdani, Marzieh Saeidi, Fabrizio Silvestri, Sebastian Riedel, and Alon Halevy. 2021. Database reasoning over text. In ACL. 3091–3104.
- (Vaswani et al, 2017) Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30
- Enzo Veltri, Donatello Santoro, Gilbert Badaro, Mohammed Saeed, and Paolo Papotti. 2022. Pythia: Unsupervised Generation of Ambiguous Textual Claims from Relational Data. In SIGMOD - Demo track. ACM.
- (Veltri et al., 2022) Enzo Veltri, Gilbert Badaro, Mohammed Saeed, and Paolo Papotti. 2022. Data Ambiguity Profiling for the Generation of Training Examples. Accepted In ICDE 2023.
- (Wang et al, 2021) Fei Wang, Kexuan Sun, Muhao Chen, Jay Pujara, and Pedro Szekely. 2021. Retrieving Complex Tables with Multi-Granular Graph Representation Learning. In SIGIR. ACM, 1472–1482.
- (Wang et al., 2021b) [34] Zhiruo Wang, Haoyu Dong, Ran Jia, Jia Li, Zhiyi Fu, Shi Han, and Dongmei Zhang. 2021. TUTA: Tree-based Transformers for Generally Structured Table Pre-training. In ACM SIGKDD. 1780–1790.
- Xiaoyu Yang and Xiaodan Zhu. 2021. Exploring Decomposition for Table-based Fact Verification. In EMNLP 2021. ACL, Punta Cana, Dominican Republic, 1045– 1052. https://aclanthology.org/2021.findings-emnlp.90
- (Yin et al., 2020) Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. TaBERT: Pretraining for Joint Understanding of Textual and Tabular Data. In ACL. ACL, Online, 8413–8426. https://doi.org/10.18653/v1/2020.acl-main.745
- (Yu et al, 2021) Tao Yu, Chien-Sheng Wu, Xi Victoria Lin, bailin wang, Yi Chern Tan, Xinyi Yang, Dragomir Radev, richard socher, and Caiming Xiong. 2021. GraPPa: GrammarAugmented Pre-Training for Table Semantic Parsing. In International Conference on Learning Representations.
- (Yu et al., 2018) Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. 2018. Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. In EMNLP. ACL, 3911–3921.
- Li, Yuliang, et al. "Deep entity matching with pre-trained language models." Proceedings of the VLDB Endowment 14.1 (2020): 50-60.
- (Zhong et al., 2017) Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2SQL: Generating structured queries from natural language using reinforcement learning. arXiv preprint arXiv:1709.00103 (2017).
- (Cheng et al., 2021), Zhoujun Cheng, Haoyu Dong, Fan Cheng, Ran Jia, Pengfei Wu, Shi Han, and Dongmei Zhang. 2021. Fortap: Using formulae for numerical- reasoning-aware table pretraining. arXiv preprint arXiv:2109.07323.
- (Hu et al., 2019) Kevin Hu, Snehalkumar’Neil’S Gaikwad, Madelon Hulsebos, Michiel A Bakker, Emanuel Zgraggen, César Hidalgo, Tim Kraska, Guoliang Li, Arvind Satyanarayan, and Ça˘gatay Demiralp. 2019. VizNet: Towards a large-scale visualization learning and benchmarking repository. CHI

References

- (Dong et al., 2019) Haoyu Dong, Shijie Liu, Zhouyu Fu, Shi Han, and Dongmei Zhang. 2019. Semantic structure extraction for spreadsheet tables with a multi-task learning architecture. In Workshop on Document Intelligence at NeurIPS 2019
- (Jauhar et al., 2016) Sujay Kumar Jauhar, Peter Turney, and Eduard Hovy. 2016. TabMCQ: A dataset of general knowledge tables and multiple-choice questions. arXiv preprint arXiv:1602.03960.
- (Pasupat and Liang, 2015) Panupong Pasupat and Percy Liang. "Compositional semantic parsing on semi-structured tables." arXiv preprint arXiv:1508.00305 (2015).
- (Kwiatkowski et al., 2019) Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. TACL, 7:453–466.
- (Chen et al., 2021) Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Yang Wang, and William W. Cohen. 2021. Open question answering over tables and text. In ICLR
- (Chen et al., 2020b) Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. 2020b. HybridQA: A dataset of multihop question answering over tabular and textual data. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1026–1036
- (Robertson et al., 1995) Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at trec-3. Nist Special Publication Sp, 109:109.
- (Treviso et al, 2022) Treviso, M. et al, “Efficient Methods for Natural Language Processing: A Survey”, arXiv e-prints 2022.
- (Yang et al 2009) Xiaoyan Yang, Cecilia M. Procopiuc, Divesh Srivastava: Summarizing Relational Databases. Proc. VLDB Endow. 2(1): 634-645 (2009)
- (Zaheer et al, 2020) Big Bird: Transformers for Longer Sequences. NeurIPS 2020