Deep Data Integration

Wang-Chiew Tan
Meta AI
wangchiew@fb.com

Disclaimer: All opinions presented in this talk are my own.

Data Integration

- The data integration problem:
  - provide uniform access to disparate data sources
- The user sees only one data source

Data Integration

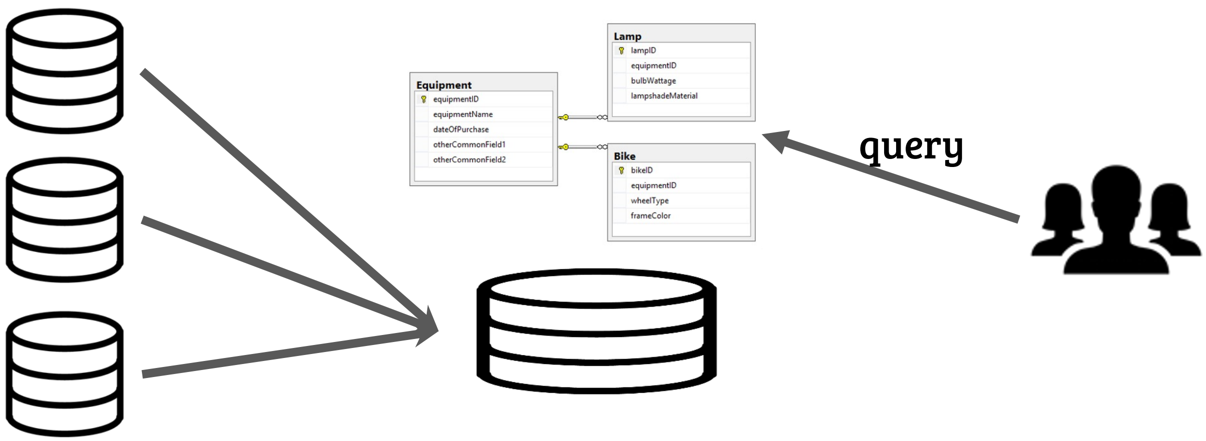
- The data integration problem:
  - provide uniform access to disparate data sources
- The user sees only one data source
- Traditionally, two approaches:
  - Virtual Data Integration
  - Data Warehouse
- Today:
  - Data Lake

Virtual Data Integration

Data sources Global schema

- Data reside at their original locations
- Global schema ⇒ uniform view of underlying data sources

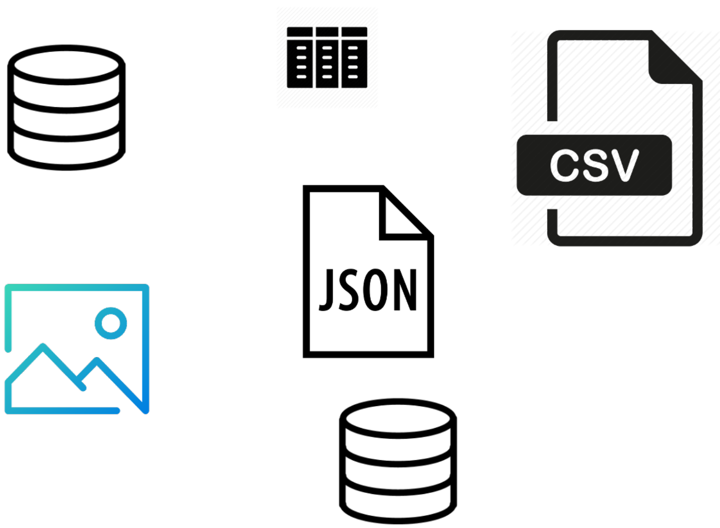
Data Warehouse



Data sources Data warehouse

- Data is consolidated at the warehouse
- Warehouse ⇒ uniform view of underlying data sources

Data Lake



- Massive collection of raw data
- May not have a schema
- May have different types
- May be in different locations

- How can we query the data lake?

The Data Integration Ecosystem

- **Data Discovery:** What are the relevant data sources?
- **Data Extraction**: How to identify and extract relevant information from sources?
- **Schema Matching/Schema Mapping**: How are data in different sources are potentially related? How to specify the relationship between the source and global/warehouse schema?
- **Entity Matching**: How to identify identical entities in different sources?
- **Data Cleaning**: How to manage missing or erroneous data?
- :

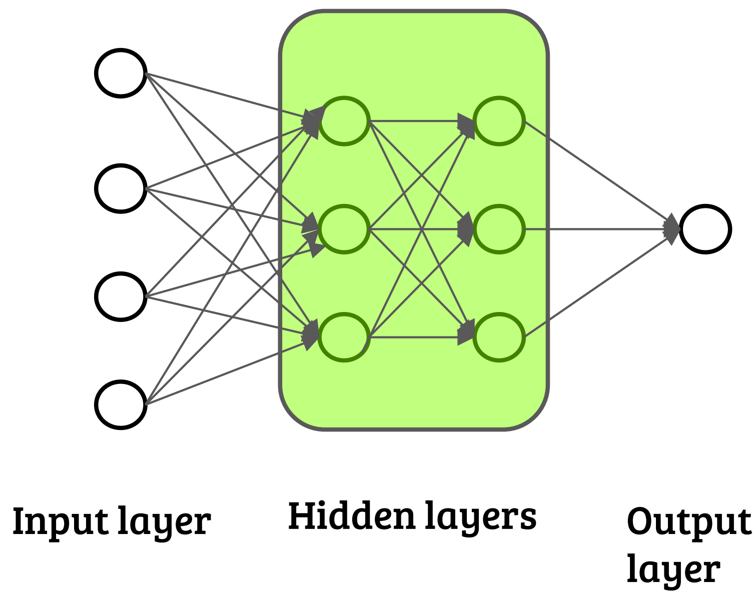
Outline

- Data Integration and Data Preparation
- Deep Learning
- Case Study: Entity Matching with Pre-trained Language Models
- Challenges and Opportunities

A Neural Network (NN)

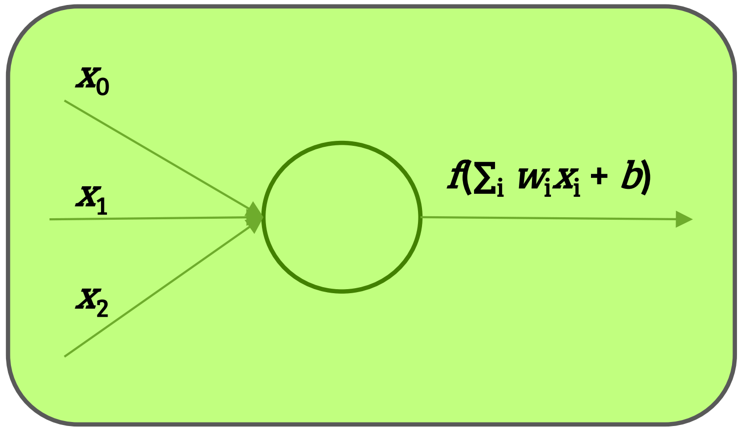
neuron

Deep = many many hidden layers



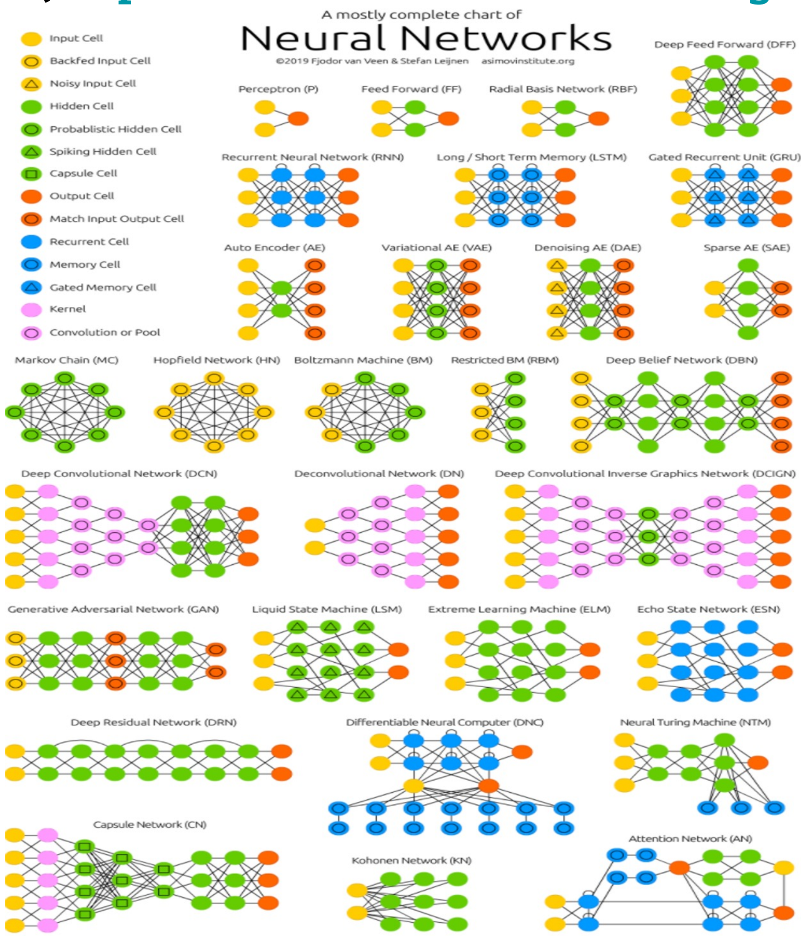
- (1) an input layer – a numerical representation of data, (2) one or more hidden layers, (3) an output layer
- Input: a numerical representation of data
- Output: the answer

A Neuron

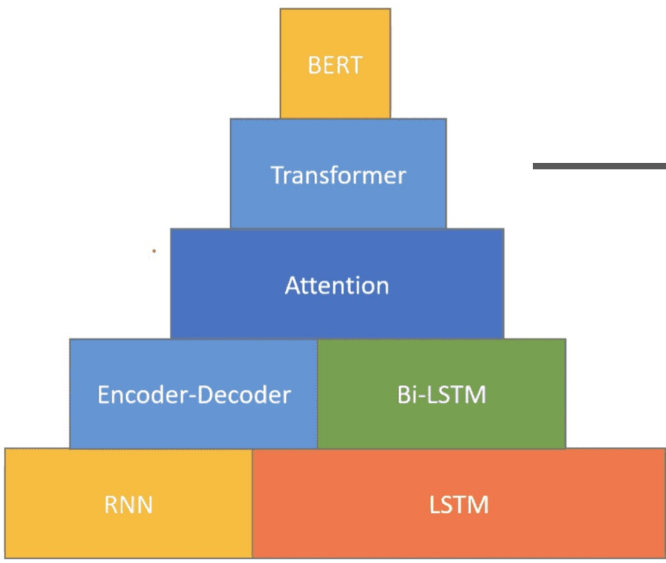


- Each neuron passes information as defined above
  - w = weight, b = bias, f = activation function
- The learning process tunes w and b:
  - compare predicted output with actual output
  - adjust w and b in all layers to minimize a loss function (e.g., mean squared error) through back propagation

The Network Zoo (https://www.asimovinstitute.org/neural-network-zoo/)



Transformers



A gentle introduction to BERT model – Anand Srivastava
https://inblog.in/A-gentle-introduction-to-BERT-Model-JfGFFXb97v

NeurIPS2017

Attention Is All You Need

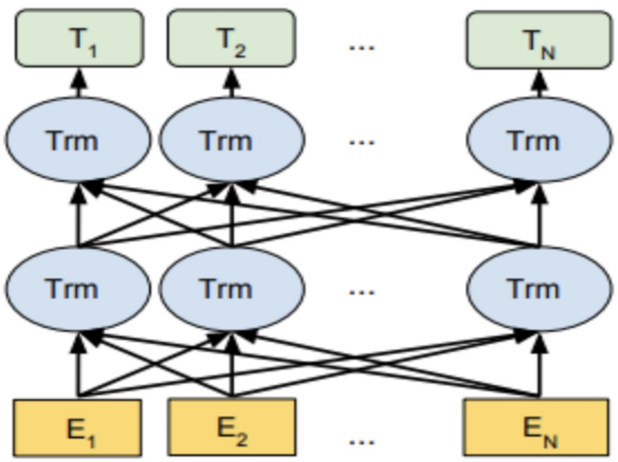
Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

NeurIPS 2017

Transformers

- Self-Attention
  - Calculates vector representation of a token based on its relation to all neighboring tokens ➔ contextualized embeddings
    - “The river **bank** was covered with flowers”
    - ”The **bank** issued a financial statement”
- Multi-head attention
  - Contextualized embeddings for different relations (e.g., subj-verb, subj-adj relations)
- Positional embeddings
  - Self-attention is position invariant
  - Positional embeddings used to indicate relative word positions

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [Devlin+ NAACL 2019]



- Takes entire sequence of tokens as input simultaneously

- Pre-training/fine-tuning paradigm
- Pre-trained on two unsupervised tasks simultaneously
  - Masked Language Model
  - Next Sentence Prediction
- Trained on large BookCorpus and English Wikipedia datasets
- Fine-tuning (later)

Transformers War

BART [Lewis ACL2020 (BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension)]

BERT (DistillBert, BERT_{base}, BERT_{large})

[Conneau+ ACL2020 (Unsupervised Cross-lingual Representation Learning at Scale)] **XLM-R**

XLNet [Yang+ NeurIPs2019 (XLNet: Generalized Autoregressive Pretraining for Language Understanding)]

[Lan+ ICLR2020 (ALBERT: A Lite BERT for Self-supervised Learning of Language Representations)] **Albert**

T5 [Raffel+ JMLR2019 (Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer)]

GPT-3 (GPT2, GPT)
[Brown+ NeurIPs2020 (Language Models are Few Shot Learners)]

DeBERTa [He+ arXiv2020 (DeBERTa: Decoding-enhanced BERT with Disentangled Attention)]

Entity Matching (EM)

- Given two data sources, find all pairs of entities, one from each data source, that refer to the same entity
- One of the most prevalent problems in data integration
- Important for deduplication, KB construction, data search
- Work as early as [Felligi & Sunter J. American Statistical Assoc.1969 (A Theory for Record Linkage)]
- The name itself needs entity resolution! [Gurajada+ CIKM2019 (Learning-Based Methods with Human-in-the-Loop for Entity Resolution)]

Entity resolution

Record linkage

Reference reconciliation

Duplicate detection

Ditto: Deep Entity Matching with Pre-trained Language Models

[Yuliang Li, Jinfeng Li, Yoshihiko Suhara, AnHai Doan, T.   VLDB2021]

- **Input:** Two collections of data entries (tables, JSON files, text, ...)
- **Output:** all entry pairs that refer to the same entity (products, businesses, ...)

Table A:

title	manf./modelno	price
instant immersion spanish deluxe 2.0	topics entertainment	49.99
adventure workshop 4th-6th grade 7th edition	encore software	19.99
sharp printing calculator	sharp el1192bl	37.63

Table B:

title	price
instant immers spanish dlux 2	36.11
encore inc adventure workshop 4th-6th grade 8th edition	17.1
new-sharp shr-el1192bl two-color printing calculator 12-digit lcd black red	56.0

Two Phases of Entity Matching

- **Blocking**
  - Reduce the number of pairwise comparisons (otherwise O(N^2))
  - Simple heuristics, e.g., two entries must share at least 1 token
- **Matching:**
  - Decide whether each candidate pair is a real match
  - Rules, Crowdsourcing, classic ML, Deep Learning, etc.

title	manf./modelno	price
instant immersion spanish deluxe 2.0	topics entertainment	49.99
adventure workshop 4th-6th grade 7th edition	encore software	19.99
sharp printing calculator	sharp el1192bl	37.63

title	price
instant immers spanish dlux 2	36.11
encore inc adventure workshop 4th-6th grade 8th edition	17.1
new-sharp shr-el1192bl two-color printing calculator 12-digit lcd black red	56.0

Entity Matching is Challenging

title	manf./modelno	price
instant immersion spanish deluxe 2.0	topics entertainment	49.99
adventure workshop 4th-6th grade 7th edition	encore software	19.99
sharp printing calculator	sharp el1192bl	37.63

title	price
instant immers spanish dlux 2	36.11
encore inc adventure workshop 4th-6th grade 8th edition	17.1
new-sharp shr-el1192bl two-color printing calculator 12-digit lcd black red	56.0

State-of-the-art EM solutions fail to match/non-match in all these 3 cases!
(as of April 2020)

Challenges

- Observations:
  - Language understanding is an important component of EM
  - What to pay attention to for each record
  - Dirty data

title	manf./modelno	price
instant immersion spanish deluxe 2.0	topics entertainment	49.99
adventure workshop 4th-6th grade 7th edition	encore software	19.99
sharp printing calculator	sharp el1192bl	37.63

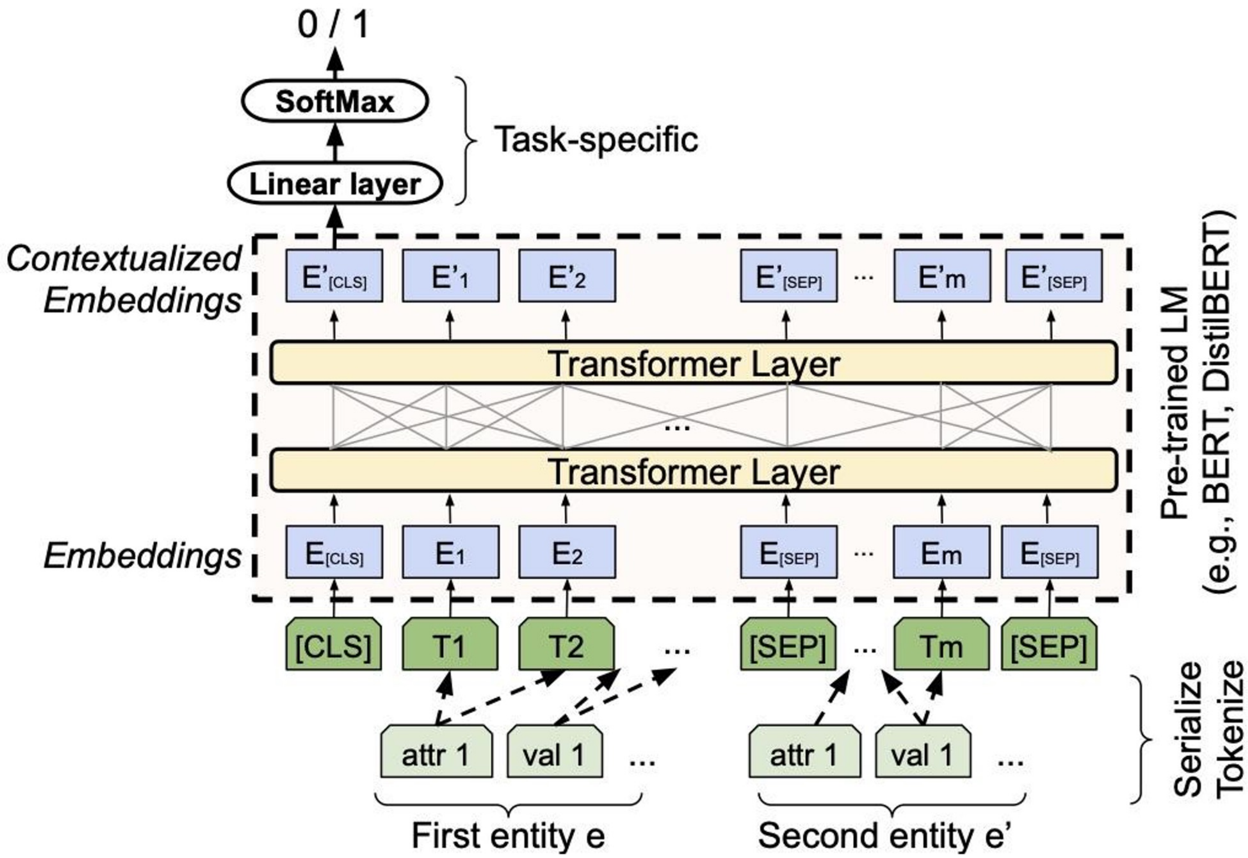
title	price
instant immers spanish dlux 2	36.11
encore inc adventure workshop 4th-6th grade 8th edition	17.1
new-sharp shr-el1192bl two-color printing calculator 12-digit lcd black red	56.0

Fine-tuning Pre-trained Language Models

- Pre-trained LM are already trained on a large dataset
- Strong baselines for several NLP tasks
- "Cheaper" to fine-tune a pre-trained LM with labeled data for your needs than to pre-train a model from scratch

- Train some layers, freeze the others
- E.g., Freeze all layers, attach new layers, train the weights of the new layers

Ditto's Model Architecture



Serialization

- **Serialize each entity:**

```
[COL] title [VAL] instant immers spanish dlux 2
[COL] manf./modelno [VAL] NULL [COL] price [VAL] 36.11
```

Special tokens for start of attribute names/values

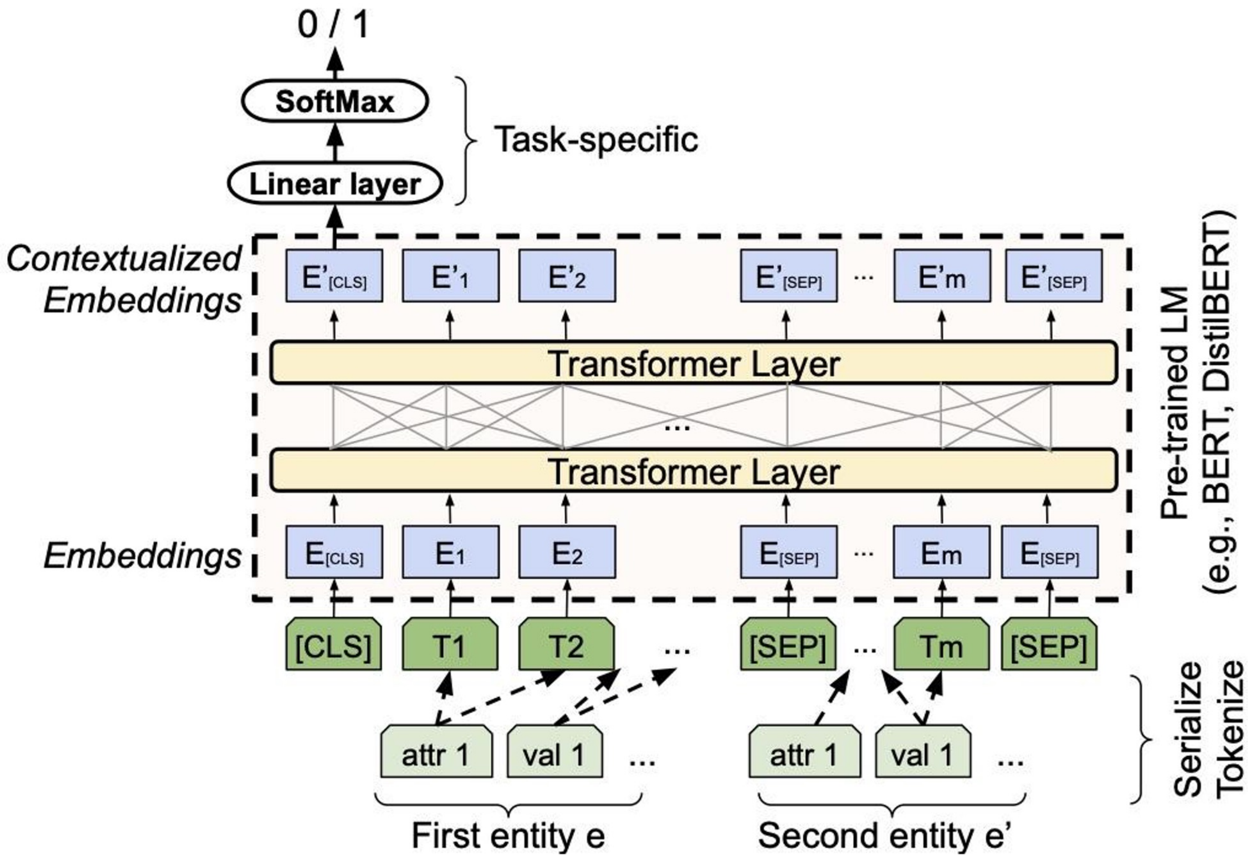
- **Apply LM (e.g., BERT) for sequence pair classification!**

```
[CLS] serialize(e) [SEP] serialize(e') [SEP]
```

First Entity

Second Entity

Ditto’s Model Architecture



RoBERTa for better performance and DistilBERT for fast training / prediction

Optimizations in Ditto

- **Injecting Domain-Knowledge:**
  - allow the user to specify information that is more important (e.g., PID)
  - e.g., "... new-sharp [ID] shr-el1192bl [/ID] two-color ..."
- **Span typing:** Use spacy or regex to identify and assign entity types

Entity Type	Types of Important Spans
Publications, Movies, Music	Persons (e.g., Authors), Year, Publisher
Organizations, Employers	Last 4-digit of phone, Street number
Products	Product ID, Brand, Configurations (num.)

- **Span normalization:** Normalize spans (e.g., numbers, years) into the same formats

Optimizations in Ditto

- **Summarization:**
  - Transformers have a max sequence length (e.g., 512)
  - Keep only the essential information → keep tokens of **high TF-IDF**
- **Data Augmentation:**
  - Allows the model to learn "harder" by **modifying the training data**
  - e.g., Dropping a span, delete an attribute, swapping two attributes, ...

  - MixDA: performs a convex interpolation on original and augmented text to generate a new one

Experiments

- Benchmark 1: **ER-Magellan**
  - 13 datasets
  - 3 domains: *publications, products, and businesses*
  - 3 categories: *Structured, Dirty, and Textual*
- Benchmark 2: **WDC Product Matching**
  - >200K of product pairs
  - 4 product categories: *computers, cameras, shoes, and watches*
  - *small (1/20), medium (1/8), large (1/2), and xlarge (1/1)*
- Baseline: **DeepMatcher (DM)**, the SOTA deep learning model for matching
  - We compare the F1 score and the training time
- **Also ran on a real company matching dataset**

Experiments: ER-Magellan datasets (w/ RoBERTa)

Datasets	Size	Ditto	DeepMatcher
Structured/Amazon-Google	11,460	75.58	69.30
Structured/Beer	450	94.37	78.80
Structured/DBLP-ACM	12,363	98.99	98.40
Structured/DBLP-GoogleScholar	28,707	95.60	94.70
Structured/Fodors-Zagats	946	100.00	100.00
Structured/iTunes-Amazon	539	97.06	91.20
Structured/Walmart-Amazon	10,242	86.76	71.90
Dirty/DBLP-ACM	12,363	99.03	98.10
Dirty/DBLP-GoogleScholar	28,707	95.75	93.80
Dirty/iTunes-Amazon	539	95.65	79.40
Dirty/Walmart-Amazon	10,242	85.69	53.80
Textual/Abt-Buy	9,575	89.33	62.80
Textual/Company	112,632	93.69	92.70

Ditto consistently outperforms DM

More robust to noisy, small, and text-heavy data

Up to 32% F1 improvement (9.43% in average)

Experiments: WDC product datasets (w/ DistillBERT for faster training)

	Ditto	DeepMatcher	Size
Small (1/20)			
computers	**80.76**	70.55	2834
cameras	**80.89**	68.59	1886
watches	**85.12**	66.32	2255
shoes	**75.89**	73.86	2063
all	**84.36**	76.34	9038
Medium (1/8)			
computers	**88.62**	77.82	8094
cameras	**88.09**	76.53	5255
watches	**91.12**	79.31	6413
shoes	**82.66**	79.48	5805
all	**88.61**	79.94	25567

	Ditto	DeepMatcher	Size
Large (1/2)			
computers	**91.70**	89.55	33359
cameras	**91.23**	87.19	20036
watches	**95.69**	91.28	27027
shoes	**88.07**	90.39	22989
all	**93.05**	89.24	103411
xLarge (1/1)			
computers	**95.45**	90.8	68461
cameras	**93.78**	89.21	42277
watches	**96.53**	93.45	61569
shoes	**90.11**	92.61	42429
all	**94.08**	90.16	214736

Ditto already outperforms DeepMatcher when given only 1/2 of training data!

Ablation Analysis

Ditto w. DA only

Ditto w. DK only

No optimization

	Ditto	Ditto (DA)	Ditto (DK)	Baseline
Structured	88.48	87.98	88.20	85.99
Dirty	91.33	91.00	90.41	88.39
Textual	87.52	86.97	87.26	61.37
WDC_small	83.67	84.36	82.13	81.08
WDC_xlarge	94.11	94.08	91.78	91.63

- All 3 optimizations are effective
- DK is more effective on the ER-Magellan datasets
- DA is more effective on the WDC datasets

Case study: company matching

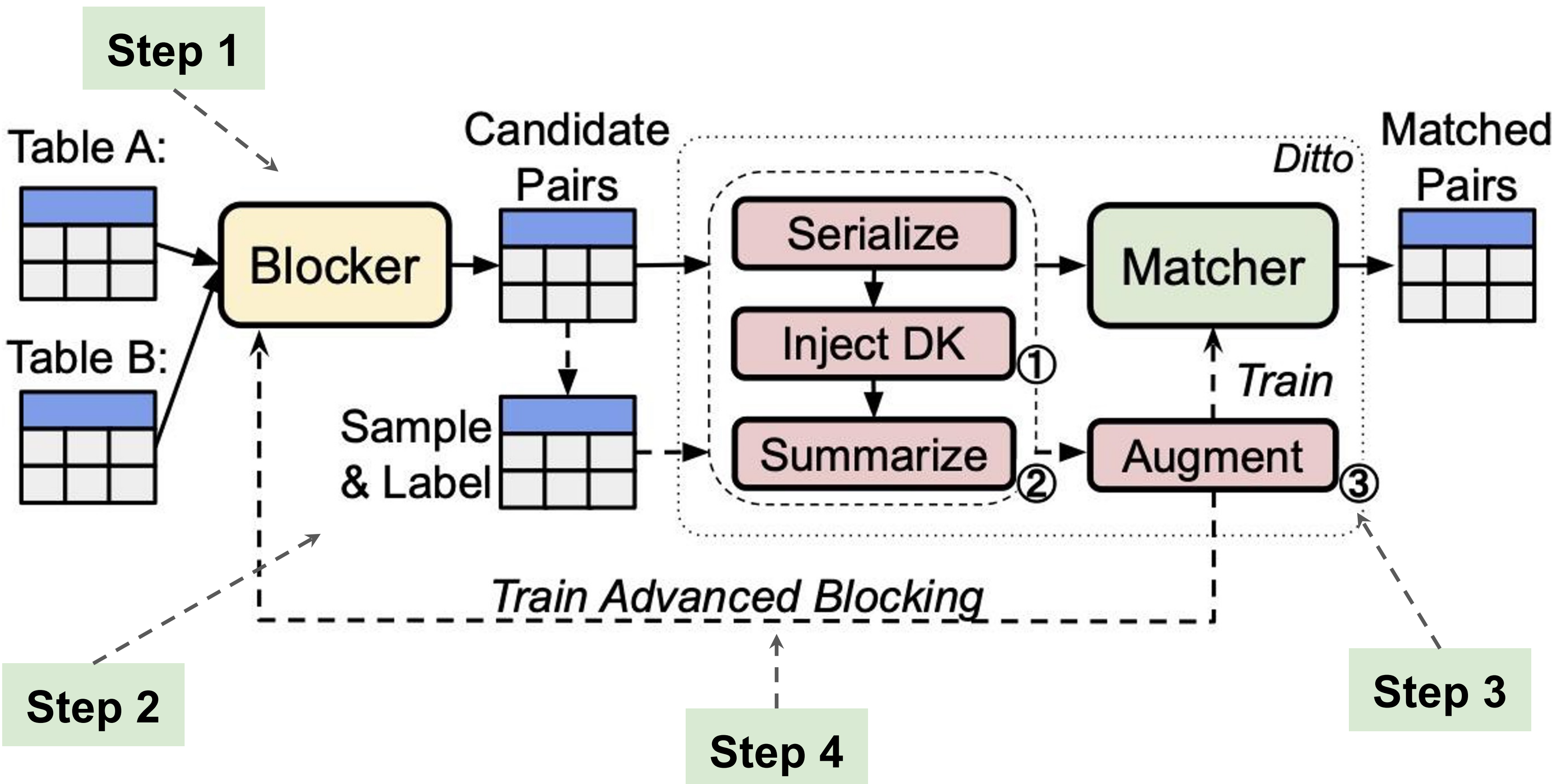
- **Given two tables A and B of companies, find record pairs that refer to the same company**

name	addr	city, state, zip	phone
M-Theory Group	6171 W Century Blvd # 350	Los Angeles, CA 90045-5336	+1.877.682.4555
M-THEORY CONSULTING GROUP, LLC	6171 W. CENTURY BLVD.	LOS ANGELES, CA 90045	2137858058

Same

- **Ditto matches two tables of 789K and 412K entries with 96.5% F1**

The Complete Pipeline with Ditto



Entity Matching & Deep Learning

- Concurrent work on applying pre-trained LM to EM. Technique is identical to Ditto's baseline [Brunner, Stockinger EDBT20 (Entity Matching with Transformer Architectures - A Step Forward in Data Integration)]
- RNN based [Mudgal+ SIGMOD18 (Deep Learning for Entity Matching), Ebraheem+ VLDB18 (Distributed representations of tuples for entity resolution)]
- Hierarchical-based Deep Learning EM solution [Zhao, He WWW2019 (Auto-EM: End-to-end Fuzzy Entity-Matching using Pre-trained Deep Models and Transfer Learning)]
- Mitigate data hungry DL based EM solutions:
  - Transfer Learning + Active Learning [Kasai+ACL19 (Low-resource Deep Entity Resolution with Transfer and Active Learning)]
  - Data Augmentation [Miao+SIGMOD21 (Rotom: A Meta-Learned Data Augmentation Framework for EM, Data Cleaning, Text Classification, and Beyond)]
- Contrastive DNN approach [Wang+ ICDM20 (CorDEL: A Contrastive Deep Learning Approach for Entity Linkage)]
- Transformer based Deep Learning models for EM [Tracz+ ACLWorkshop20 (BERT-based similarity learning for product matching)]
  - Bert-based similarity learning for product matching
- The Four Generations of Entity Resolution [Papadakis+ 21 Morgan&Claypool publishers]
- :

Deep Learning & other Data Integration Tasks

- Information extraction:
  - Named Entity Recognition [Li+ TKDE20 (A survey of DL methods for NER)]
  - Relation Extraction [Nayak+ ArXiv21 (Deep Neural approaches to relation triplets extraction)]
  - Opinion Mining [Irsoy, Cardie EMNLP14 (Opinion Mining with Deep Recurrent NN)] [Miao+ WWW20 (Snippext: Semi-supervised Opinion Mining with Augmented Data)]
  - Sentiment Analysis [Zhang, Wang, Liu Wiley18 (Deep Learning for Sentiment Analysis: A survey)]

Deep Learning & other Data Integration Tasks

- Table understanding [Deng+VLDB20 (TURL: Table Understanding through Representation Learning)] [Hulsebos+SIGKDD19 (Sherlock: A Deep Learning Approach to Semantic Data Type Detection.)] [Zhang+VLDB20 (Sato: Contextual Semantic Type Detection in Tables)] [Trabelsi+ arXiv20 (Semantic Labeling Using a Deep Contextualized Language Model)] [Herzig+ ACL20. (Tapas: Weakly supervised table parsing via pre-training)] [Yin+ ACL20. (Tabert: Pretraining for joint understanding of textual and tabular data)] [Lockard+arXiv21 (TCN: Table Convolutional Network for Web Table Interpretation)] [Wang+arXiv 20. (Structure-aware Pre-training for Table Understanding with Tree-based Transformers)]
- Data curation/preparation
  - [Thirumuruganathan+EDBT20 (Data Curation with Deep Learning)]
  - [Tang+arXiv21 (RPT: Relational Pre-trained Transformer Is Almost All You Need towards Democratizing Data Preparation)]
- Querying Tables/Text [Thorne+VLDB21 (to appear) (From Natural Language Processing to Neural Databases)] [Yin+ACL20 Tabert: Pretraining for joint understanding of textual and tabular data]
- :

Effectiveness of Deep Learning in Data Integration

- Suitable for tasks where rules are difficult to specify, features are hard to engineer
  - Many data integration problems are like this
  - Variations and nuances in language, heterogeneity in content and structure, dirty data, context
- Robust to data imperfections
  - Can deal with missing or wrong values, missing meta-data, heterogeneous data

Effectiveness of Deep Learning in Data Integration

- Immense language understanding
  - Pre-training:
    - Lower layers capture lexical structure.
    - Higher layers capture more semantic properties of a language
    - Deeper layers track longer-distance linguistic dependencies
    - BERT representations capture linguistic information in a compositional way that mimics classical, tree-like structures
      [Clark+ BlackBoxNLP19 (What does Bert look at? An Analysis of BERT's attention] [Jawahar, Sagot, Seddah ACL19 (What does BERT learn about the Structure of Language)] [Tenney, Das, Pavlick ACL19 (Bert Rediscovers the Classical NLP Pipeline] [Jiang+ TACL20. (How Can We Know What Language Models Know?)] [Roberts, Raffel, Shazeer EMNLP20 (How Much Knowledge Can You Pack Into the Parameters of a Language Model?)]
    - Difference between "Sharp TV" vs "Sharp resolution"
    - Similarity between "Stop hair loss" vs "Prevents thinning hair"

Effectiveness of Deep Learning in Data Integration

- Immense ability to learn from examples. Attention is key

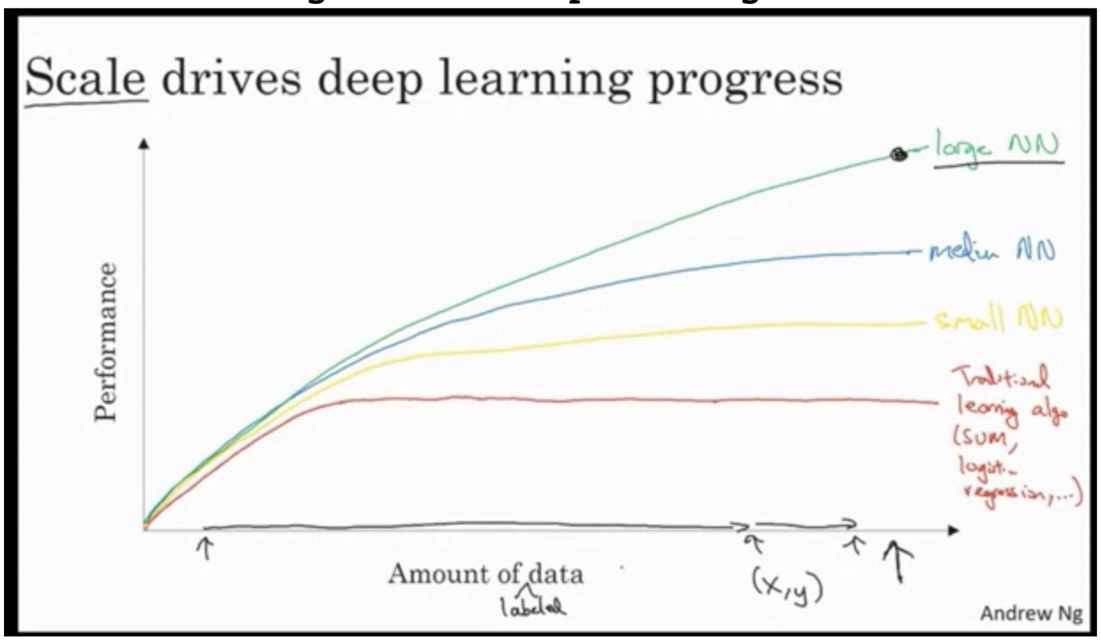
title	manf./modelno	price
instant immersion spanish deluxe 2.0	topics entertainment	49.99
adventure workshop 4th-6th grade 7th edition	encore software	19.99
sharp printing calculator	sharp el1192bl	37.63

title	price
instant immers spanish dlux 2	36.11
encore inc adventure workshop 4th-6th grade 8th edition	17.1
new-sharp shr-el1192bl two-color printing calculator 12-digit lcd black red	56.0

What is the catch?

- Data hungry
  - Quality of a DL model is directly dependent on its training data

From Andrew Ng's NN and Deep Learning course



- Traditional ML models' performance plateaus with more training data
- Larger NN tends to perform better with more training data

What is the catch?

- Data hungry
  - Quality of a DL model is directly dependent on its training data
  - The more training data, the better.

quality

- Quality training data is expensive to obtain
  - Often a significant data integration problem

Disadvantages of using Deep Learning for Data Integration

- Data hungry
  - Quality of a DL model is directly dependent on its training data
  - The more quality training data, the better
  - Quality training data is expensive to obtain
  - Fairness/Bias in training data
- Requires high performance hardware
- Longer latency. Expensive to deploy
- Complex: lots of hyperparameters (BERT-base 110M, BERT-large 340M)
- Opaque

Challenges and Opportunities

- Benchmarks for DI tasks
  - Comprehensive benchmarks for data cleaning, table understanding, entity matching etc.
  - E.g., EM: include numerical heavy data, different types of dirty data and include metrics for measuring fairness/biasness in data
- Techniques to mitigate data hungry DL solutions:
  - Data Augmentation: generate additional training data fairly
  - Transfer learning, Active Learning, Weak supervision

Challenges and Opportunities

- Model Explainability:
  - Explain the results of your DI tasks
  - Generate rules for the DI task which are also explainable
  - Explain a model's decision. E.g., LIME: Local Interpretable Model Agnostic Explanations
    - Generate explanations for why and why-not questions
- Querying heterogeneous heterogeneous data (different structure, different modalities)
  - Query data "outside the box"
    - Structured data/text/images/audio/video in a virtual DI setting

Andrew Ng on MLOps: From Model-centric to Data-centric AI (March 2021)

"When a system isn't performing well, many teams instinctually try to improve the code. But for many practical applications, it's more effective instead to focus on improving the data"

"If Google has BERT then OpenAI has GPT-3. But, these fancy models take up only 20% of a business problem. What differentiates a good deployment is the quality of data; everyone can get their hands on pre-trained models or licensed APIs."

Can we integrate data for social good?

- World today:
    - Content: text/images/audio/video
- Can we integrate data to understand the world for a variety of purposes?
    - Understand the origins of content
    - Understand the entities and relationships between entities in the content, and related content
    - Understand the meaning or intent of content

Acknowledgements

Deep thanks to:

Yuliang Li, Jinfeng Li, Yoshihiko Suhara (Megagon Labs)
AnHai Doan (University of Wisconsin)

Some slides on Entity Matching are originally from Yuliang Li

Alon Halevy (Meta AI)