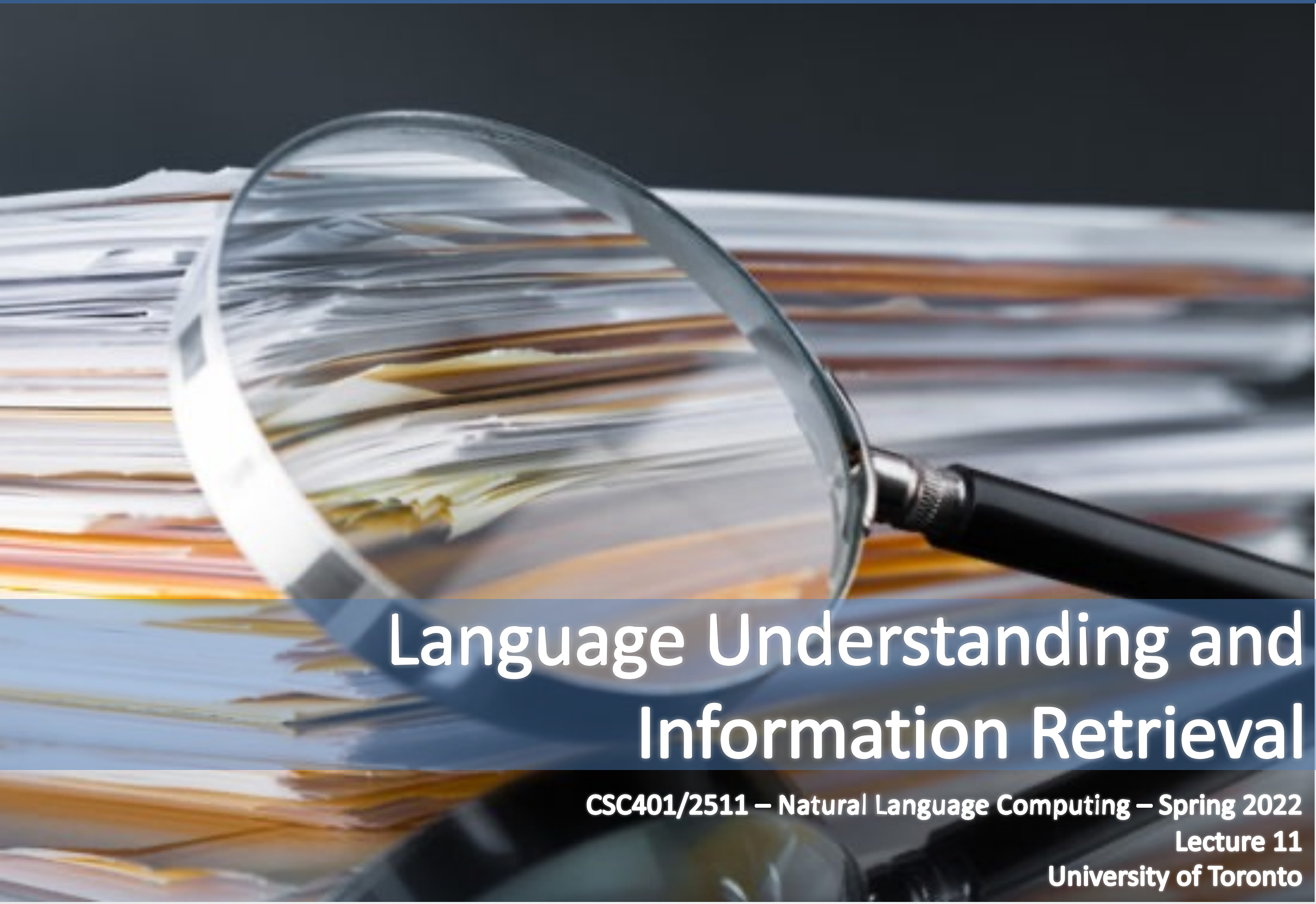


Language Understanding and Information Retrieval

CSC401/2511 – Natural Language Computing – Spring 2022

Lecture 11

University of Toronto

Agenda

- March 23 (today): Natural languages understanding
- March 28 Monday: Information retrieval

Understanding “what the human means” so I can decide how to reply

NATURAL LANGUAGE UNDERSTANDING

Natural Language Understanding

- Understanding natural language is a very broad research direction.
- NLP researchers have considered several related tasks:
  - Detect the sentiment
  - Decide if a sentence entails the other
  - Detect the stance
  - Examine if AI models can use some common sense.
  - … Many others

Detect the sentiment

- Is each row a positive or a negative comment?
- These examples come from the SST2 dataset in the GLUE benchmark – a binary **classification** problem.

Sentence	Label
that 's far too **tragic** to merit such superficial treatment	negative
equals the original and in some ways even **betters** it	positive
the plot is **nothing but boilerplate** clichés from start to finish	negative
gorgeous and deceptively minimalist	positive

Socher, Richard, et al. "Recursive deep models for semantic compositionality over a sentiment treebank." *Proceedings of EMNLP*. 2013.

Detect the entailment

- Do the premise entail the hypothesis?
- These examples come from the MNLI dataset – a 3-class **classification** problem.

Premise	Hypothesis	Label
How do you know? **All this is their information** again.	**This information belongs to them.**	entail
and it is nice talking to you all righty	I talk to you every day.	neutral
Fun for **adults and children**.	Fun for **only children**.	contradict

Williams, Adina, Nikita Nangia, and Samuel R. Bowman. "The Multi-genre NLI corpus." (2018).

Detect the stance

- In this thread, what is the stance of each tweet?
- An example from SemEval 2017 task 8:

u1: We understand there are two gunmen and up to a dozen hostages inside the café under siege at Sydney.. ISIS flags remain on display #7News [support]

> **u2:** @u1 not ISIS flags [deny]
>
> **u3:** @u1 sorry – how do you know it's an ISIS flag? Can you actually confirm that? [query]
>
> > **u4:** @u3 no she can't cos it's actually not [deny]
>
> **u5:** @u1 More on situation at <location> <link> [comment]
>
> **u6:** @u1 Have you actually confirmed its an ISIS flag or are you talking shit [query]

Derczynski L, Bontcheva K, Liakata M, Procter R, Hoi GW, Zubiaga A. SemEval-2017 Task 8: RumourEval: Determining rumour veracity and support for rumours. arXiv preprint arXiv:1704.05972. 2017 Apr 20.

Reasoning with common sense

Here is an example from the **Winograd Schema Challenge**:

- The trophy doesn’t fit into the brown suitcase because it is too small. What is too small?
- The trophy doesn’t fit into the brown suitcase because it is too large. What is too large?

Aside: WSC is designed as a **pronoun resolution** problem.

Levesque, Hector, Ernest Davis, and Leora Morgenstern. "The Winograd Schema Challenge." *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*. 2012.

GLUE and SuperGLUE

- Just like the ImageNet for CV, there are some leaderboards in NLP:
- General Language Understanding Evaluation (**GLUE**) is a popular suite of datasets.
- And a more recent suite, **SuperGLUE**.
- Some datasets we mentioned just now are included, e.g., SST2 in GLUE, and WSC in SuperGLUE

Mind the corner cases!

A commercial sentiment analysis model (G) failed many **"corner-case" tests**:

- **M**in **F**unctional **T**est (MFT)
- **Inv**ariance test (INV)
- **Dir**ectional perturbation test (DIR)

Test case	Expected	Predicted	Pass?
A Testing **Negation** with ***MFT*** — Labels: negative, positive, neutral			
Template: I {NEGATION} {POS_VERB} the {THING}.			
I can't say I recommend the food.	neg	pos	X
I didn't love the flight.	neg	neutral	X
...			
Failure rate = 76.4%			
B Testing **NER** with ***INV*** — Same pred. (inv) after removals / additions			
@AmericanAir thank you we got on a different flight to [Chicago → Dallas].	inv	pos / neutral	X
@VirginAmerica I can't lose my luggage, moving to [Brazil → Turkey] soon, ugh.	inv	neutral / neg	X
...			
Failure rate = 20.8%			
C Testing **Vocabulary** with ***DIR*** — Sentiment monotonic decreasing (↓)			
@AmericanAir service wasn't great. You are lame.	↓	neg / neutral	X
@JetBlue why won't YOU help them?! Ugh. I dread you.	↓	neg / neutral	X
...			
Failure rate = 34.6%			

Ribeiro, Marco Tulio, et al. "Beyond accuracy: Behavioral testing of NLP models with CheckList." ACL 2020

Popular methods for NLU

- NLU are usually formulated as **machine learning classification problems**, so ML models can in general apply.
- As of early 2022, the best DNN-based models perform better than "human baselines" on the GLUE leaderboard.
  - "Human baselines" means: the agreement between randomly selected English speakers and the experts who annotated the test sets.
  - The DNN-based models are usually **fine-tuned** from **language models** (e.g., RoBERTa).

Aside: Better NLU Systems

- How to develop better NLU systems?
- Here are some possible answers:
  - More high-quality data.
  - Better data representations.
  - Larger models.
  - Better model structures.
- In the next a few slides, we will look at the solutions reflecting the avenues.

GPT-3: More high-quality data

- CommonCrawl text data from 2016 to 2019.
  - 45TB compressed text data are noisy.
  - Filtered down to 570GB.

Filter steps:

1. Automatically predict document quality, then resample the high-quality documents.
2. (Fuzzy) deduplication.
3. Remove the documents contained in evaluation datasets.

Note that GPT-3 also used huge models.
More about "large models" later.

Brown, Tom, et al. "Language models are few-shot learners." *Advances in neural information processing systems* 33 (2020): 1877-1901.

ERNIE: Knowledge-enhanced LM

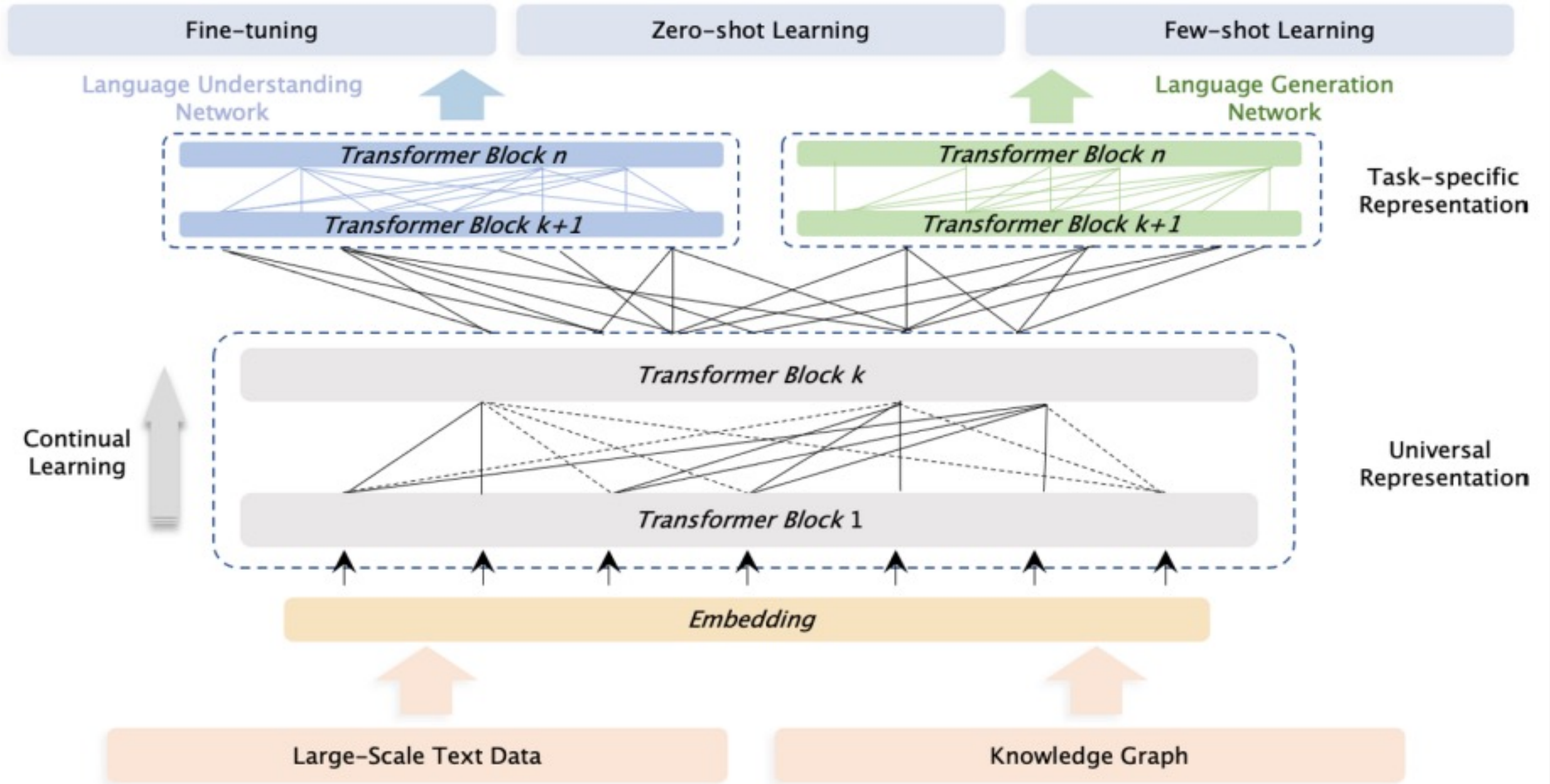


Figure 1: The framework of ERNIE 3.0.

ALBERT: A Light (and larger) BERT

- Larger models might be more expressive, but there are memory bottlenecks.
  - Too much parameters won't fit a GPU card.
- Two techniques to reduce # parameters:
  - Divide large vocabulary embedding matrix into two small matrices.
  - Share parameters across the layers.
- Result? "We are able to scale up to much larger ALBERT configurations but still have fewer parameters than BERT-large."

Lan, Zhenzhong, et al. "Albert: A lite bert for self-supervised learning of language representations." *arXiv preprint arXiv:1909.11942* (2019).

DeBERTa: Enhanced structures

- Remember that in BERT: $v_w = v_{content} + v_{position}$
- DeBERTa uses two separate vectors, $v_{content}\ (H)$ and $v_{position}(P)$ separately.
- For tokens at locations i and j, 4 types of attentions are computed:
  - Content-to-content $H_i H_j^T$
  - Content-to-position $H_i P_{j,i}^T$
  - Position-to-content $P_{i,j} H_j^T$
  - Position-to-position $P_{i,j} P_{j,i}^T$

He, Pengcheng, et al. "Deberta: Decoding-enhanced bert with disentangled attention." *arXiv preprint arXiv:2006.03654* (2020).

Debate: Form vs. Meaning

- Recently, researchers debated around the claim "DNNs understand language".
- ML classifiers may reach nontrivial accuracies by relying only on **shortcuts**:
- Here is an example of natural languages inference task:

S1: You have access to the facts.
S2: The facts are accessible to you.
Label: Entailment
Correct reason: S1 entails S2.
Shortcut: They both contain the, to, and you

Debate: Form vs. Meaning

- Many NLU models failed to generalize on more challenging datasets.
- Here is an example from a challenge dataset, **HANS**:
  - Rely on meaning -> correct result
  - Rely on shortcuts -> incorrect result

S1: The doctor was paid by the actor.
S2: The doctor paid the actor.
Label: Contradict
Shortcut: They both contain the, doctor, paid, the, actor.

DNNs failed badly!

McCoy, R. Thomas, Ellie Pavlick, and Tal Linzen. "Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference." *arXiv preprint arXiv:1902.01007* (2019).

Debate: Form vs. Meaning

- There are evidence supporting counter-arguments. Here is an example showing RoBERTa learns some "deeper" features.
- Example features:

Feature name	Feature description
Surface feature: Length	Is sentence longer than n=3 words?
Linguistic feature: Syntactic category	Does sentence have an adjective?

Warstadt, Alex, et al. "Learning which features matter: RoBERTa acquires a preference for linguistic generalizations (eventually)." *EMNLP 2020*.

Debate: Form vs. Meaning

- How to test if a model relies on surface (S) vs. linguistic (L) feature?
- Train on an **ambiguous** dataset , which consists of only two types of samples:
  - S=0, L=0 (Label: 0)
  - S=1, L=1 (Label: 1)

Model relying on S or L can make correct predictions.

- Use a **disambiguating** test set, which consists of only two types of samples:
  - S=0, L=1
  - S=1, L=0

Model relying on S will predict 0. L: 1.
Model relying on S will predict 1. L: 0.

Warstadt, Alex, et al. "Learning which features matter: RoBERTa acquires a preference for linguistic generalizations (eventually).“ *EMNLP 2020*.

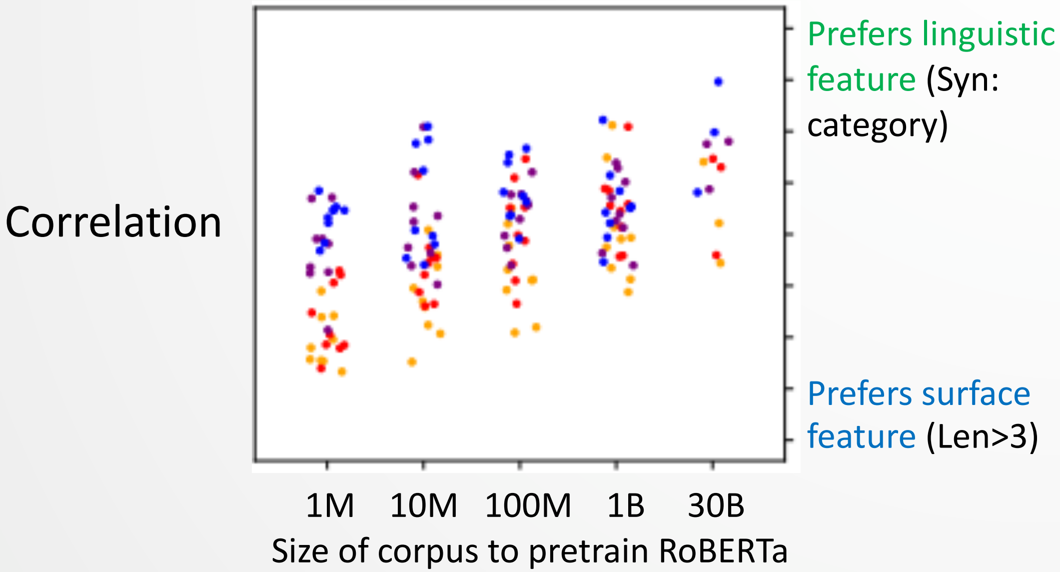
Debate: Form vs. Meaning

- Let model perform binary classification (0 vs 1). Compute the (Matthew's) correlation between model's predictions and the L features.
- If the model prefers L: correlation goes towards 1.
- If the model prefers S: correlation goes towards -1.

Warstadt, Alex, et al. "Learning which features matter: RoBERTa acquires a preference for linguistic generalizations (eventually).“ *EMNLP 2020*.

Debate: Form vs. Meaning

- RoBERTa trained with more training data prefers more “linguistic” features.

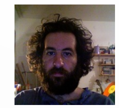


The colors reflect different constituents in the training datasets.

This is 1 of the 20 configurations. Please refer to the original paper for details:
Warstadt, Alex, et al. "Learning which features matter: RoBERTa acquires a preference for linguistic generalizations (eventually).“ *EMNLP 2020*.

Debate: Form vs. Meaning

DNN models can compute some statistics – this is sufficient to solve those NLU machine learning problems, but they still do not understand the content.



The tasks that DNN models are trained on – cloze, next-token-prediction, etc. -- provide supervision signals that teach the DNN models to capture some semantics.

For details of this debate, please refer to:
Wolf, T. "Learning Meaning in Natural Language Processing – The Semantics Mega-Thread" *Blog post:* https://medium.com/huggingface/learning-meaning-in-natural-language-processing-the-semantics-mega-thread-9c0332dfe28e

Probing the DNNs

How much shortcut do the DNNs capture?

- We need to probe the DNNs to see **what**, and **how much**, do the DNNs understand.
- Here, “probe” means “carefully examine”.
- Probing is a fast-developing field, with many ongoing research projects.
- The most popular probing method is **diagnostic classification**.

Probing as diagnostic classification

Steps for setting up a diagnostic classification:

- Collect a specified diagnostic dataset.
- Train a small, auxiliary model (e.g., LogReg) on the neural model (e.g., BERT) representations.
- High LogReg accuracy → good representations*.

Diagnostic dataset: text → Pre-trained model → Diagnostic dataset: label

* There are some debates. See e.g., [Hewitt & Liang., (2019)]

Aside: Probing BERT

- This is an application to show “which layer does BERT do what”.
- Tenney etal., (2019) considered a collection of probing tasks, including:
  - Part-of-speech (POS)
  - Dependencies (Deps.)
  - Semantic role labeling (SRL)
  - Coreference resolution (Coref.)
  - Semantic proto-roles (SPR)

Tenney, I., Das, D., & Pavlick, E. (2019). BERT Rediscovers the Classical NLP Pipeline. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4593–4601. https://doi.org/10.18653/v1/P19-1452

Aside: Probing BERT

Set up a probing classifier P as:

- Input: Weighted sum of representations across l layers (i.e., $1, 2, \dots, l$).
- Output: labels of the probing task τ.

The probing classifier produces two numbers:

- $Score\left(P^{(l)}\right)$: F1 score of probing classification.
- $s^{(l)}$: The weight assigned to layer l.

Tenney, I., Das, D., & Pavlick, E. (2019). BERT Rediscovers the Classical NLP Pipeline. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4593–4601. https://doi.org/10.18653/v1/P19-1452

Aside: Probing BERT

Measure two summary statistics:

- Center-of-gravity

$$\bar{E}[l] = \sum_{l=0}^{L} l \cdot s_\tau^{(l)}$$

The probe "attends to" here the most.

- Expected layer

$$\bar{E}_\Delta[l] = \frac{\sum_{l=1}^{L} l \cdot \Delta_\tau^{(l)}}{\sum_{l=1}^{L} \Delta_\tau^{(l)}}$$

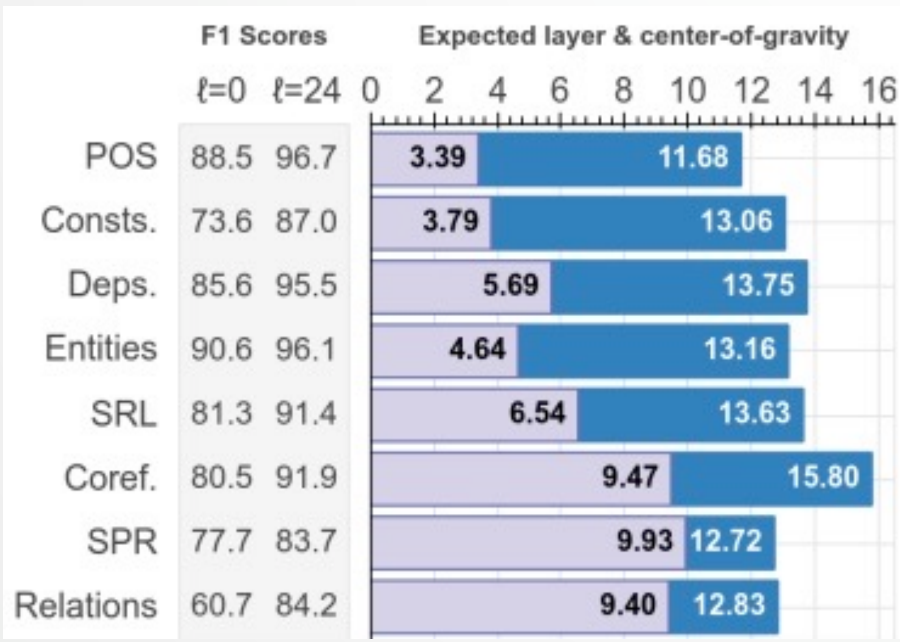
This layer is *expected* to improve *Score* the most.

Where $\Delta_\tau^{(l)} = Score\left(P_\tau^{(l)}\right) - Score\left(P_\tau^{(l-1)}\right)$.

Tenney, I., Das, D., & Pavlick, E. (2019). BERT Rediscovers the Classical NLP Pipeline. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4593–4601. https://doi.org/10.18653/v1/P19-1452

Aside: Probing BERT

The "lower" layers of BERT are more "syntactic";
The "higher" layers … more "semantic".



Statistics on BERT-large
Blue: center-of-gravity
Purple: expected layer

As if BERT re-discovers the traditional NLP pipeline!

Tenney, I., Das, D., & Pavlick, E. (2019). BERT Rediscovers the Classical NLP Pipeline. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4593–4601. https://doi.org/10.18653/v1/P19-1452

Comparing probing configurations

The probing configurations do not occur standalone.

- By probing, we mostly want to **compare**.
- Following is a hypothetical example ($R_B > R_A$):

Config A	Config B
Task: Predict tense	Task: Predict tense
Encoder: BERT (layer 2)	Encoder: BERT (layer 4)
Probe: LogReg	Probe: LogReg
Performance R_A : Accuracy=80%	Performance R_B : Accuracy=81%

- If B *really* outperforms A, how likely can people reproduce this finding?
  - Can measure with **statistical power**.

Statistical power

Empirically, power can be estimated via simulation:

- Repeat N times:
  - Sample e.g., 90% of data.
  - On these data, set up configurations A and B.
  - Do a statistical test (e.g., to test if $R_B > R_A$)
- Among the N simulations, N_{rej} gave $R_B > R_A$
  - i.e., rejected the null hypothesis.
- Then the power is $\frac{N_{rej}}{N}$.
- Usually, we require power to be at least 0.80.

Aside: Test set sizes matter!

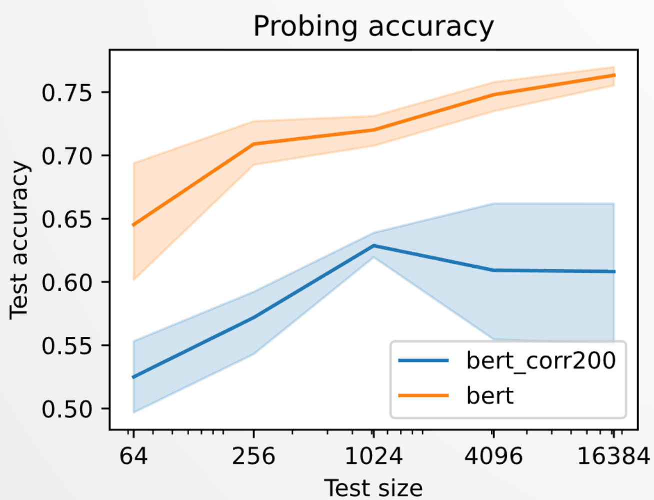
- Newsletters like to boast e.g., "AI outperformed humans on GLUE", but we should be careful, but…
- **Some leaderboard datasets are too small.**
- The results from several hundreds of data sample might not have sufficient **reliability**.

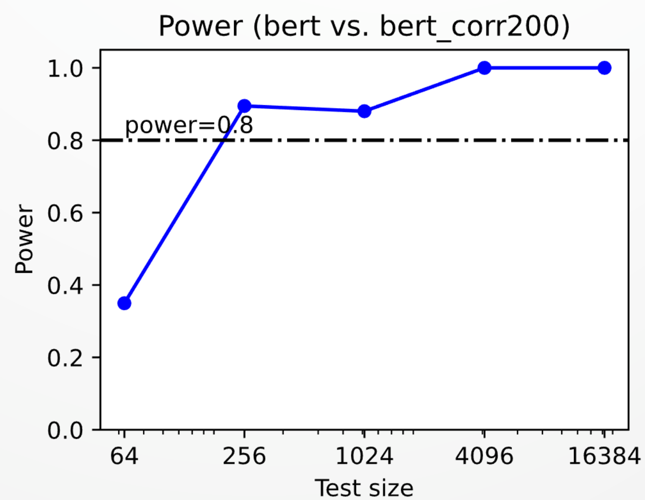
Paper 1: On **100** sentences, our model outperforms BERT by 1% accuracy.
Paper 2: On **100,000** sentences, our model outperforms BERT by 1% accuracy.

Which one of the two papers are more reliable?

Aside: sizes of probing datasets

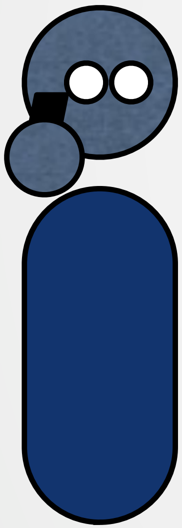
- An example: BERT outperform BERT_corr200, but only when test size >= 256 do the results have sufficient **statistical power**.





Zhu, Z., Wang, J., Li, B., & Rudzicz, F. (2022). On the data requirements of probing. *Findings of the Association of Computational Linguistics*.

NLU summary



- The language understanding problem (NLU)
- Several tasks to evaluate the NLU abilities.
- Methods for building NLU systems.
- The "form vs meaning" debate.
- Recent approaches: probing.

Retrieve useful information from knowledge base

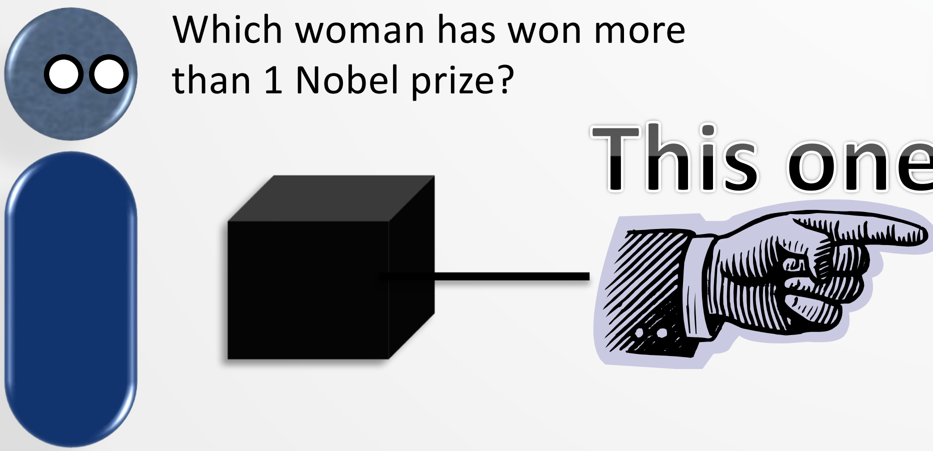
INFORMATION RETRIEVAL

UNIVERSITY OF TORONTO

Information retrieval systems

Retrieving information is also a very, very big problem.
Here is a simplification:

> Given a query, search for the most relevant document among a knowledge base.



(Marie Curie)

Information retrieval systems

Given a query, search for the most relevant document among a knowledge base.

- Addressing the task requires handling some problems:
  - How to represent the query?
  - How to store a knowledge base?
  - How to search efficiently and accurately?
- The problems are closely related. We will look at some popular approaches.

IR scenario 1: SQL

Structured Query Language (SQL) query

- How to represent the query?
  SQL queries.
- How to store a knowledge base?
  Tabular entries with predefined schemas.
- How to search efficiently and accurately?
  Compile and execute the SQL queries.

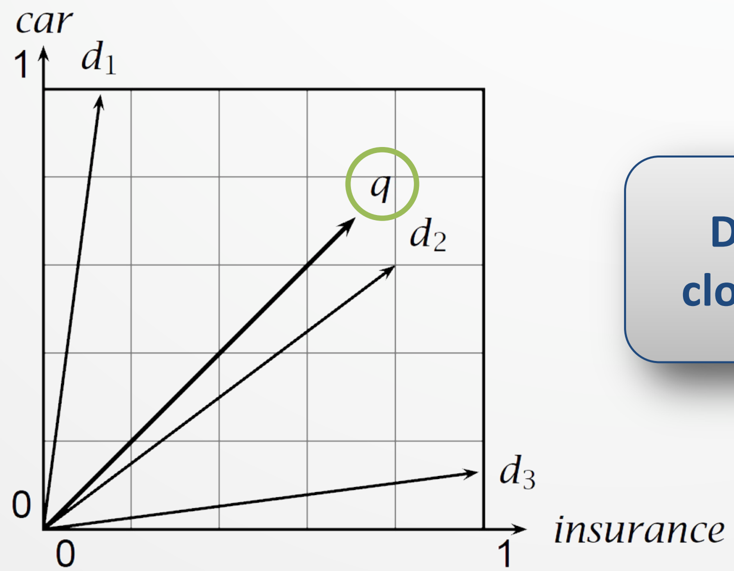
IR scenario 2: Doc2Vec

Find the documents that match the given keywords.

- How to represent the query?
  Query is just another text-based document.
- How to store a knowledge base?
  A collection of documents – actually, **vectorize** them, so it's easy to compute a similarity score.
- How to search efficiently and accurately?
  Compute the **similarity score** between the query and each document. Return the document with the highest similarity score.

Similarity score

- If the query and the available documents can be represented by vectors, we can determine **similarity** according to their **cosine distance**.
  - Vectors that are **near** each other (within a certain **angular radius**) are considered relevant.



Document d_2 is closest to query q.

UNIVERSITY OF TORONTO

Vectorization: *tf.idf*

- *tf.idf* is a traditional method to vectorize the documents.
- It starts by weighting *words* in the *documents*.

  - **Term frequency, tf_{ij}:** number of occurrences of word w_i in document d_j.
  - **Document frequency, df_i:** number of documents in which w_i appears.
  - **Collection frequency, cf_i:** total occurrences of w_i in the collection.

Term frequency

- **Higher** values of $\boldsymbol{tf_{ij}}$ (for contentful words) suggest that word w_i is a **good** indicator of the content of document d_j.
  - When considering the relevance of a document d_j to a keyword w_i, tf_{ij} should be **maximized**.

- We often **dampen** tf_{ij} to temper these comparisons.
  - $tf_{dampen} = 1 + \log(tf)$, if $tf > 0$.

Document frequency

- The **document frequency**, df_i, is the number of documents in which w_i appears.
  - **Meaningful** words may occur repeatedly in a related document, but **functional** (or less meaningful) words may be distributed evenly over all documents.

Word	Collection frequency	Document frequency
kernel	10,440	3997
try	10,422	8760

  - E.g., *kernel* occurs about as often as *try* in total, but it occurs in fewer documents – it is a more **specific** concept.

Inverse document frequency

- Very specific words, w_i, would give **smaller** values of df_i.
- To maximize specificity, the **inverse document frequency** is
$$idf_i = \log\left(\frac{D}{df_i}\right)$$
where D is the total number of documents and we scale with log, as before.
- This measure gives **full** weight to words that occur in 1 document, and **zero** weight to words that occur in all documents.

UNIVERSITY OF TORONTO

tf.idf

- We combine the **term frequency** and the **inverse document frequency** to give us a joint measure of **relatedness** between words and documents:

$$tf.idf(w_i, d_j) = \begin{cases} \left(1 + \log(tf_{ij})\right) \log \frac{D}{df_i} & \text{if } tf_{ij} \geq 1 \\ 0 & \text{if } tf_{ij} = 0 \end{cases}$$

- The j^{th} document is therefore represented by a vector:

$$[tf.idf(w_1, d_j), \\ tf.idf(w_2, d_j), \\ \dots, \\ tf.idf(w_{|W|}, d_j)]$$

Latent semantic indexing

- **Co-occurrence**: *n.* when two or more terms occur in the same documents more often than by chance.
  - Note: this is *not* the same as collocations

- Consider the following:

	Term 1	Term 2	Term 3	Term 4
Query	natural	language		
Document 1	natural	language	NLP	embedding
Document 2			NLP	embedding

  - Document 2 appears to be **related** to the query although it contains **none** of the query terms.
    - The query and document 2 are **semantically related**.

UNIVERSITY OF TORONTO

Singular value decomposition (SVD)

- An SVD projection is computed by decomposing the term-by-document matrix $A_{t \times d}$ into the product of three matrices:

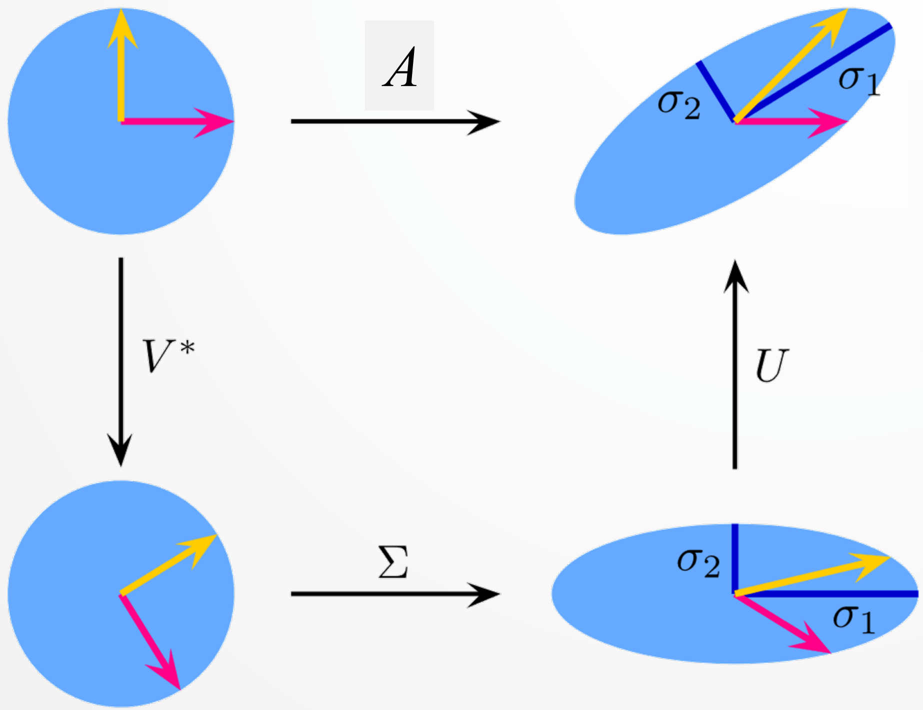
  $T_{t \times n}$, $S_{n \times n}$, and $D_{d \times n}$

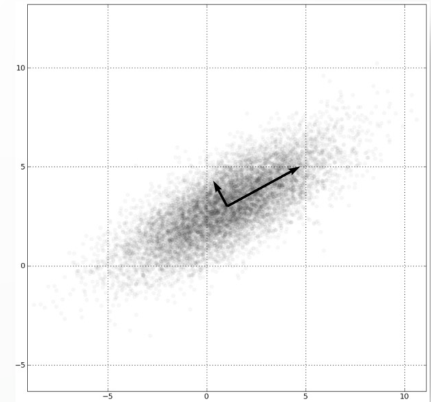
  where t is the number of words (terms),
  d is the number of documents, and
  $n = \min(t, d)$.

- Specifically,

$$\boxed{A_{t \times d} = T_{t \times n} S_{n \times n} (D_{d \times n})^{\intercal}}$$

Singular value decomposition (SVD)





$$A = U \cdot \Sigma \cdot V^*$$

UNIVERSITY OF TORONTO

SVD example

$$A_{t\times d} = T_{t\times n} S_{n\times n} (D_{d\times n})^{\intercal}$$

$A =$

	d_1	d_2	d_3	d_4	d_5	d_6
natural	1	0	1	0	0	0
language	0	1	0	0	0	0
processing	1	1	0	0	0	0
car	1	0	0	1	1	0
truck	0	0	0	1	0	1

$T =$

nat.	-0.44	-0.30	0.57	0.58	0.25
lang.	-0.13	-0.33	-0.59	0	0.73
proc.	-0.48	-0.51	-0.37	0	-0.61
car	-0.70	0.35	0.15	-0.58	0.16
truck	-0.26	0.65	-0.41	0.58	-0.09

$S =$

2.16	0	0	0	0
0	1.59	0	0	0
0	0	1.28	0	0
0	0	0	1	0
0	0	0	0	0.39

$D^{\intercal} =$

d_1	d_2	d_3	d_4	d_5	d_6
-0.75	-0.28	-0.20	-0.45	-0.33	-0.12
-0.29	-0.53	-0.19	0.63	0.22	0.41
0.28	-0.75	0.45	-0.20	0.12	-0.33
0	0	0.58	0	-0.58	0.58
-0.53	0.29	0.63	0.19	0.41	-0.22

- What do these matrices mean?

SVD example

$A =$

	d_1	d_2	d_3	d_4	d_5	d_6
natural	1	0	1	0	0	0
language	0	1	0	0	0	0
processing	1	1	0	0	0	0
car	1	0	0	1	1	0
truck	0	0	0	1	0	1

- A is the matrix of term frequencies, tf_{ij}.
  - E.g., *natural* occurs once in d_1 and once in d_3.

UNIVERSITY OF TORONTO

SVD example

- Matrices T and D represent **terms** and **documents**, respectively in this ***new*** space.
  - E.g., the first row of T corresponds to the first row of A, and so on.

$T =$

nat..	-0.44	-0.30	0.57	0.58	0.25
lang.	-0.13	-0.33	-0.59	0	0.73
proc.	-0.48	-0.51	-0.37	0	-0.61
car	-0.70	0.35	0.15	-0.58	0.16
truck	-0.26	0.65	-0.41	0.58	-0.09

$D^{\intercal} =$

$\boldsymbol{d_1}$	$\boldsymbol{d_2}$	$\boldsymbol{d_3}$	$\boldsymbol{d_4}$	$\boldsymbol{d_5}$	$\boldsymbol{d_6}$
-0.75	-0.28	-0.20	-0.45	-0.33	-0.12
-0.29	-0.53	-0.19	0.63	0.22	0.41
0.28	-0.75	0.45	-0.20	0.12	-0.33
0	0	0.58	0	-0.58	0.58
-0.53	0.29	0.63	0.19	0.41	-0.22

- T and D are **orthonormal**, so all columns are orthogonal to each other and $T^{\intercal}T = D^{\intercal}D = I$.

SVD example

- The matrix S contains the **singular values** of A in descending order.
  - The i^{th} singular value indicates the amount of variation on the i^{th} axis.

$S =$

2.16	0	0	0	0
0	1.59	0	0	0
0	0	1.28	0	0
0	0	0	1	0
0	0	0	0	0.39

SVD example

- By restricting T, S, and D to their first $k < n$ columns, their product gives us $\hat{A}$, a 'best least squares' approximation of A.

$T =$

cosm.	-0.44	-0.30	0.57	0.58	0.25
astro.	-0.13	-0.33	-0.59	0	0.73
moon	-0.48	-0.51	-0.37	0	-0.61
car	-0.70	0.35	0.15	-0.58	0.16
truck	-0.26	0.65	-0.41	0.58	-0.09

$S =$

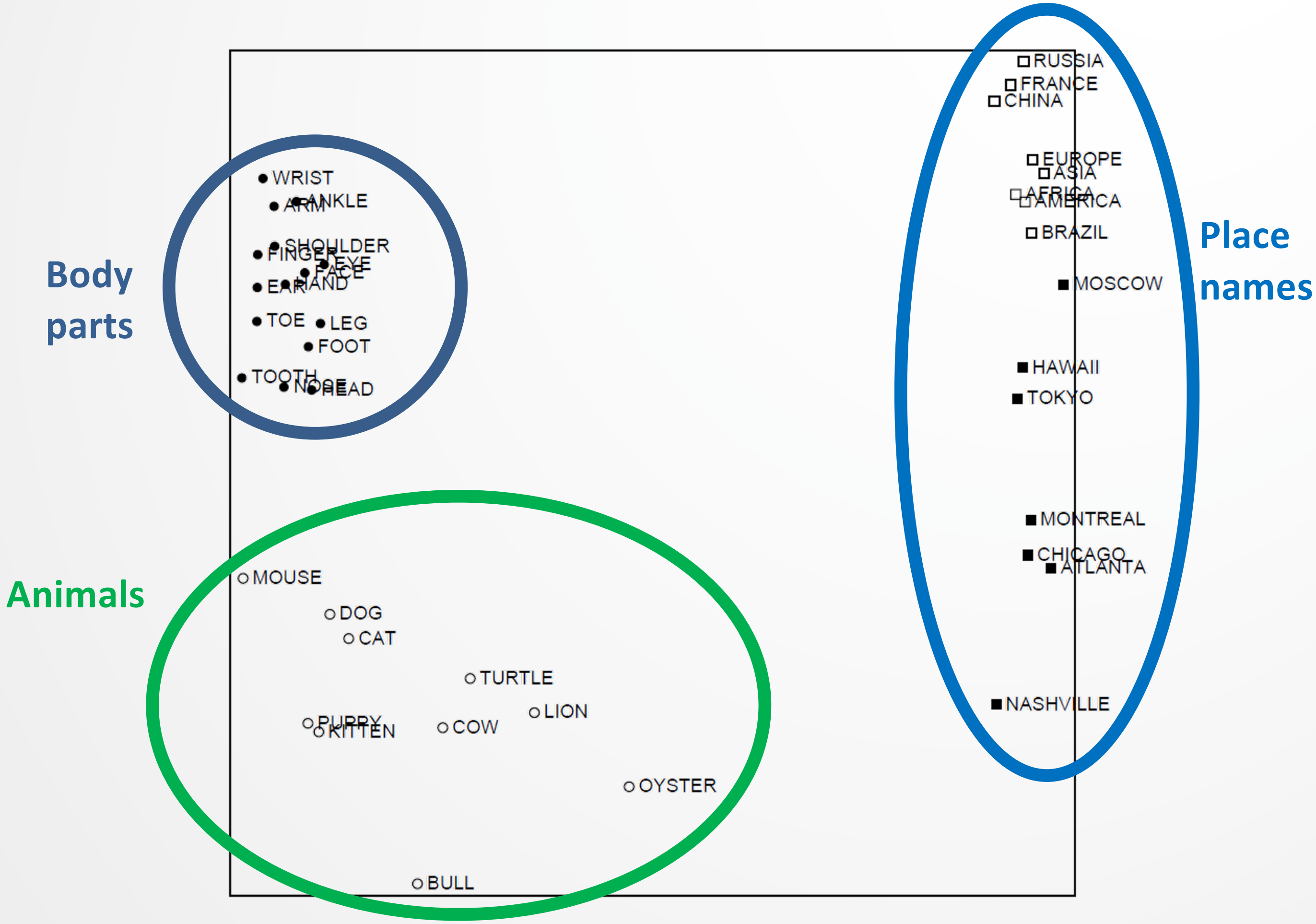
2.16	0	0	0	0
0	1.59	0	0	0
0	0	1.28	0	0
0	0	0	1	0
0	0	0	0	0.39

$D^\intercal =$

d_1	d_2	d_3	d_4	d_5	d_6
-0.75	-0.28	-0.20	-0.45	-0.33	-0.12
-0.29	-0.53	-0.19	0.63	0.22	0.41
0.28	-0.75	0.45	-0.20	0.12	-0.33
0	0	0.58	0	-0.58	0.58
-0.53	0.29	0.63	0.19	0.41	-0.22

UNIVERSITY OF TORONTO

SVD in practice



Rohde *et al.* (2006) An Improved Model of Semantic Similarity Based on Lexical Co-Occurrence. *Communications of the ACM* **8**:627-633.

Neural embeddings for documents

- We can use neural embeddings for words *and* documents
  - Use term-document matrix, but swap out SVD for NNs.
  - Small amounts of **labeled** data can be used to fine-tune.

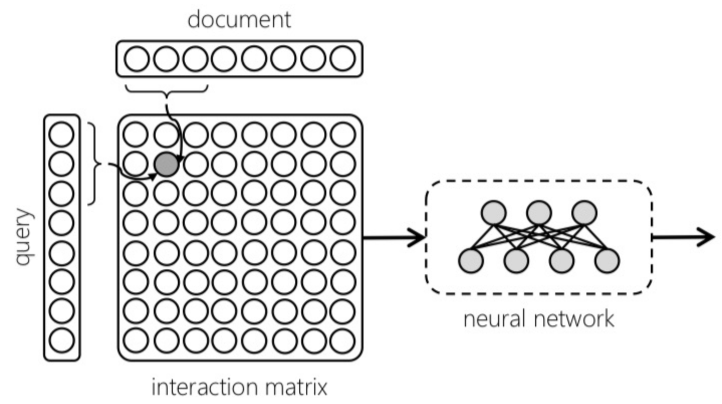


Figure 21: Schematic view of an interaction matrix generated by comparing windows of text from the query and the document. A deep neural network—such as a CNN—operates over the interaction matrix to find patterns of matches that suggest relevance of the document to the query.

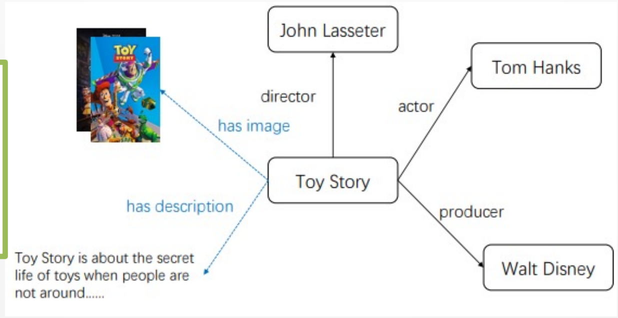
Mitra B, Craswell N. (2017) Neural Models for Information Retrieval. http://arxiv.org/abs/1705.01509

Zhang Y, Rahman MM, Braylan A, *et al.* (2016) Neural Information Retrieval: A Literature Review.

Structured vs. Unstructured

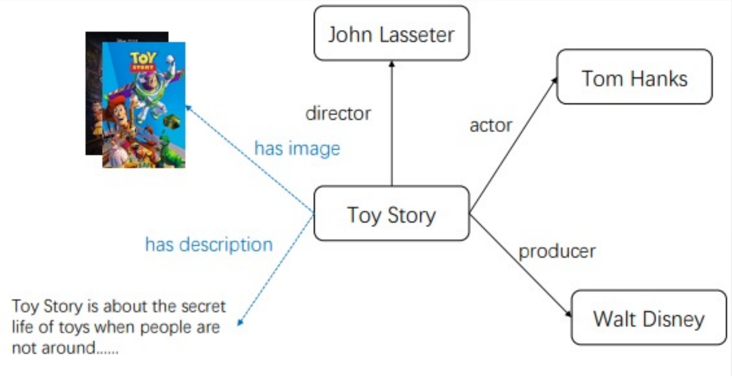
- Plain texts are **unstructured**.
- Many modern IR systems use **structured** data.
  - E.g., docs vectorized to the same dimensions.
  - E.g., relational data.
- Structured data are easier to save and use.

{“name”: “Toy Story”,
“director”: “John Lasseter”,
… }



Storing Structured Relational Data

- Saving each complex object as a database entry is one option.
- We can also store (or embed) the $\{R, S, T\}$ triplets. R is the **relation** (e.g., "has-director") between: the **source** S (e.g., "Toy Story") and the **target** T (e.g., "John Lasseter")

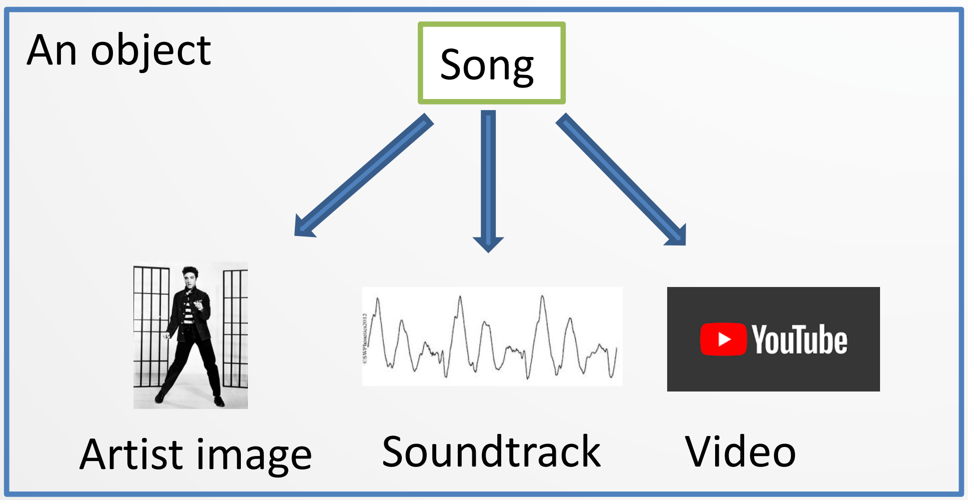


Sun, Rui, et al. "Multi-modal knowledge graphs for recommender systems." *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2020.

UNIVERSITY OF TORONTO

Multimodal vs. Plain text

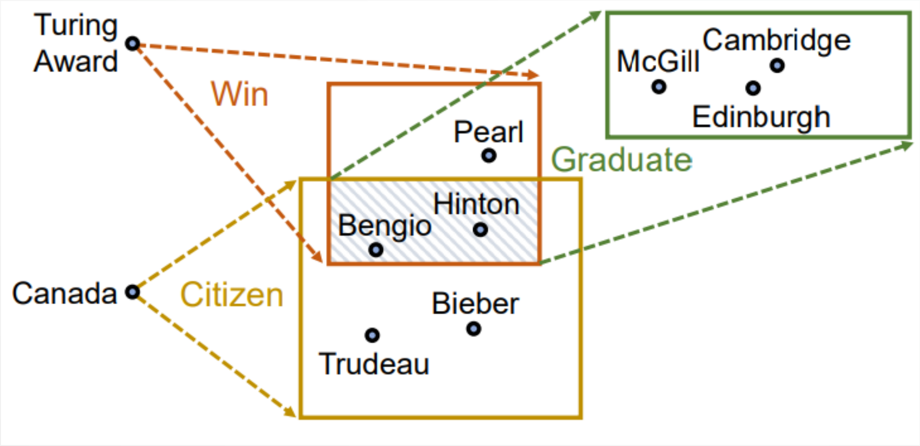
- Most modern IR systems are **multimodal**.
- The objects contain more than texts.
  - Images, sounds, even videos are stored too.
  - Choosing the right schemas is very important!



UNIVERSITY OF TORONTO

Handling complex queries

- Here is a question with 2 **conjunctive** queries: "Where did Canadian citizens with Turing Award graduate?"
- Idea: project each query into a **box** in the embedding space!



Ren, Hongyu, Weihua Hu, and Jure Leskovec. "Query2box: Reasoning over knowledge graphs in vector space using box embeddings." *ICLR 2020*

UNIVERSITY OF TORONTO

Aside: box embedding details

- Objective to learn the **box embeddings**:

$$-\log \sigma\big(\gamma - \mathrm{d}(\boldsymbol{v};\boldsymbol{q})\big) - \sum_{i=1}^{k} \frac{1}{k} \log \sigma\left(\mathrm{d}(\boldsymbol{v}';\boldsymbol{q}) - \gamma\right)$$

- This is a negative sampling loss, where:
  - γ is a fixed scalar margin.
  - $\boldsymbol{v}$ is an answer to the query $\boldsymbol{q}$.
  - $\boldsymbol{v}'$ is a non-answer to the query $\boldsymbol{q}$.
  - k is the number of negative entities.

Ren, Hongyu, Weihua Hu, and Jure Leskovec. "Query2box: Reasoning over knowledge graphs in vector space using box embeddings." *ICLR 2020*

UNIVERSITY OF TORONTO

Evaluating the retrieval

- Some commonly used metrics include:
  - Precision
  - Recall
  - F-score
  - Precision @ k

Precision and Recall

- **Precision**: $\frac{N_{\text{relevant \& retrieved}}}{N_{\text{retrieved}}}$
  - Among all retrieved documents, how many are relevant?
  - Precision in machine learning: $\frac{TP}{P}$
- **Recall**: $\frac{N_{\text{relevant \& retrieved}}}{N_{\text{relevant}}}$
  - Among all relevant documents, how many are retrieved?
  - Recall in machine learning: $\frac{TP}{T}$

F-score

- **F-score** is the weighted harmonic mean of precision and recall:

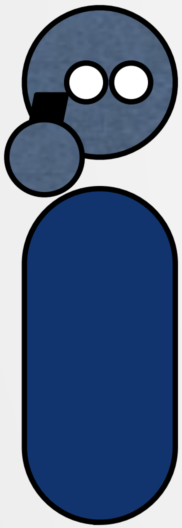
$$F = \frac{1}{\alpha \frac{1}{p} + (1 - \alpha) \frac{1}{\mathrm{r}}}$$

- Where p is precision, r is recall, and $\alpha \in [0,1]$.
- Notes:
  - When $\alpha = \frac{1}{2}$, we have $F_1 = \frac{2pr}{p+r}$
  - If either of precision or recall is 0 (i.e., true positive count $TP = 0$), then F is arbitrarily set to 0.

Precision at k (P@k)

- Modern IR systems usually do not just give one result.
  - Even if the 1^{st} result is not relevant, the 2^{nd}, etc. results could be relevant too.
- People sometimes measure the **precision at k (P@k)**:
  - Among the top k results, how many of them are relevant?
- **P@k** has some potential problems:
  - The 1^{st}, 2^{nd}, …, k^{th} locations have no differences.
  - If there are less than k relevant results, then even the best system can't get **P@k**=1.

Information Retrieval summary



- What information retrieval is.
- Doc2vec, latent semantic indexing.
- Neural IR systems.
- Structured data and complex queries.
- Evaluation of IR systems.