CITS3005 Knowledge Representation

Lecture 11: Inductive Knowledge

The University of Western Australia

2023

Induction and Learning

While deductive knowledge is characterised by precise logical consequences, inductively acquiring knowledge involves generalising patterns from a given set of input observations, which can then be used to generate novel but potentially imprecise predictions.

Knowledge Acquisition

Knowledge Extraction

,

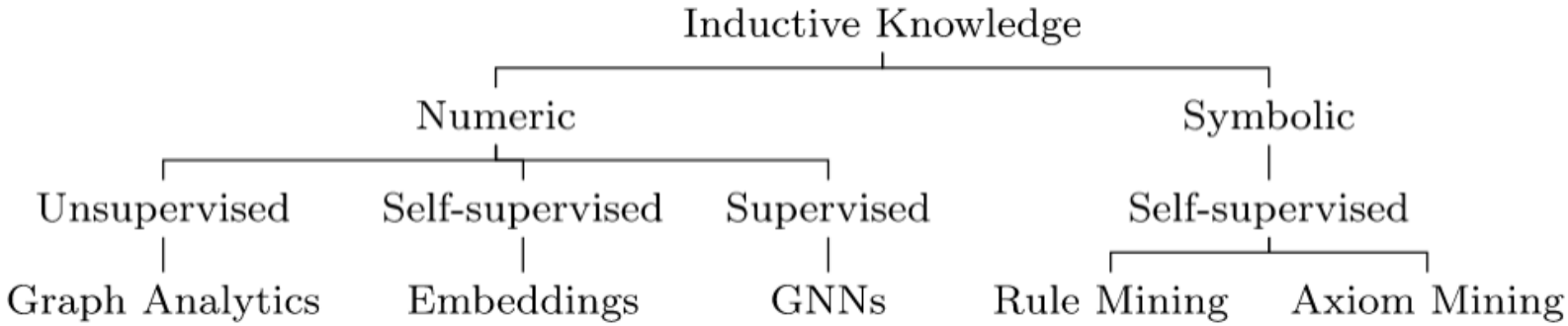
- Texts, natural langauge
- Structured formats, databases, spreadsheets, data stream
- Unstructured formats, audio, video, signal processing
- Experience

- Knowledge graph completion, link prediction
- Rule mining, hypothesis mining
- Causal networks
- Question Answering

Inductive Techniques

There are many approaches to induction over knowledge graphs including:

- unsupervised methods and graph analytics, which detects communities or clusters in a graph.
- knowledge graph embeddings can use self-supervision to learn a low-dimensional numeric model of a knowledge graph.
- supervised learning methods such as graph neural networks can use the structure of graphs.
- symbolic learning can learn symbolic models and logical formulae from a graph in a self-supervised manner.



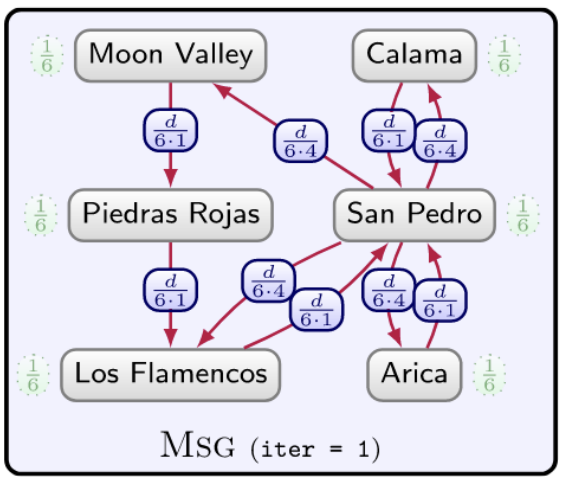
Graph Analytics

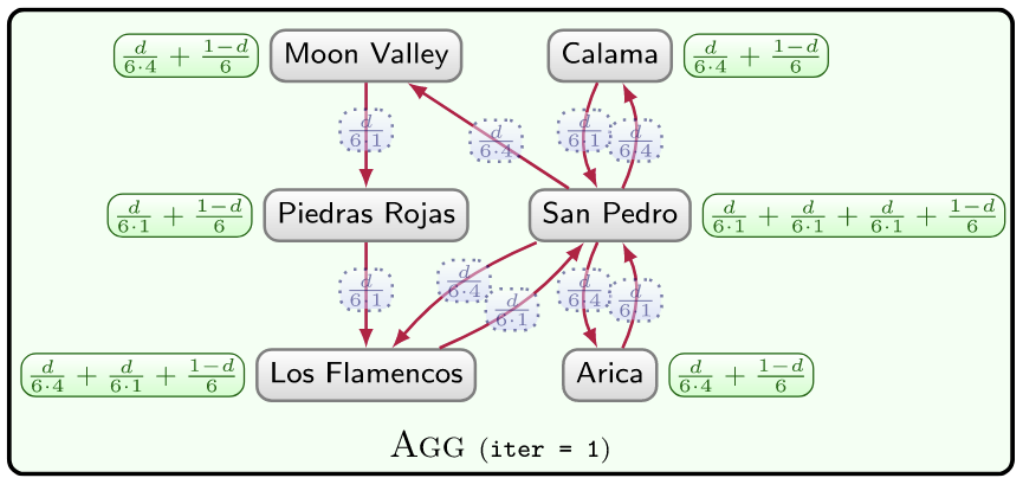
Graph analytics is the application of analytical processes to graph data. i.e., how the nodes of the graph are connected. Techniques include:

- Centrality: aims to identify the most important or central nodes or edges of a graph. Centrality measures include degree, betweenness, closeness, Eigenvector, PageRank, HITS.
- Community detection: aims to identify communities in a graph, or sub-graphs that are more densely connected.
- Connectivity: aims to estimate how well-connected the graph is.
- Node (or vertex) similarity: aims to find nodes that are similar to other nodes. Node similarity metrics may be computed using structural equivalence, random walks, diffusion kernels.
- Path finding: aims to find paths in a graph, typically between pairs of nodes given as input.

Example: PageRank

The PageRank algorithm was developed by Larry Page and Sergey Brin for determining the popularity of web pages, based on links, or the centrality of nodes.
In the tourist example, PageRank calculates the probability of a tourist randomly following the routes shown in the graph being at a particular place after a given number of "hops".
The calculation can be efficiently approximated at large scales.





Machine Learning Approaches

In the context of knowledge graphs, machine learning can either be used for directly refining a knowledge graph or for downstream tasks, such as recommendation, information extraction, question answering.
To encode graphs, nodes and edges as numeric vectors, we need embeddings!

Terminiology

- Supervised learning: training materials comprise input-output value pairs as examples
- Unsupervised learning: training materials comprise only input examples
- Other variants: semi-supervised, reinforcement learning

Data for Superivied Learning

Training examples can be split in three (80-10-10):

- training data are used to train the model
- validation data are used to optimise hyper-parameters and monitor progress
- test data are used only for final evaluation

More Machine Learning

Epochs and batches

- We can go through the training data many times: each go-through is an epoch
- we go through the training examples in groups: each group is called a batch
- Each example creates a loss: numeric difference between the actual and the "correct" result

Underfitting is when the modelled has not trained long enough, overfitting is when the model has trained too long.

Evaluation Measures

Results without ranking:

- accuracy (A): ratio of correct results
- precision (P), recall (R), $F1 = 2PR/(P+R)$,...

Ranked results:

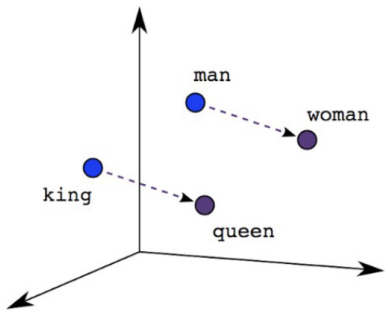
- Hit@n: number of correct results in the top n, e.g., Hit10
- Mean Rank: average rank of the correct results
- Mean Reciprocal Rank (MRR): average inverse rank of the higest-ranked correct result for each query,

Eg: the best correct results for three queries have ranks 3, 1, 28 so $MRR = (1/3 + 1/1 + 1/28)/3 = 1.37$.

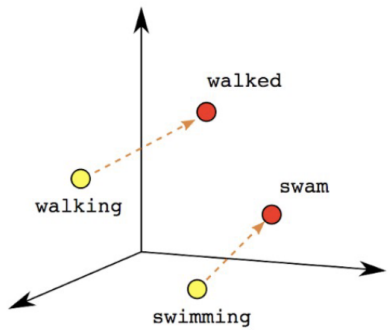
Word Embeddings

How can we represent the meaning of words?

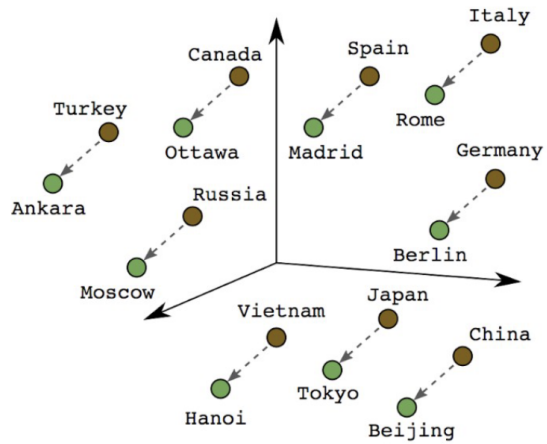
- By designation (e.g., textual descriptions in a dictionary)
- As nodes in a network (e.g., in a knowledge graph like DBPedia)
- Formally (e.g., adding axioms to a RDFS vocabulary or OWL ontology)
- As vectors in a latent semantic space



Male-Female



Verb Tense

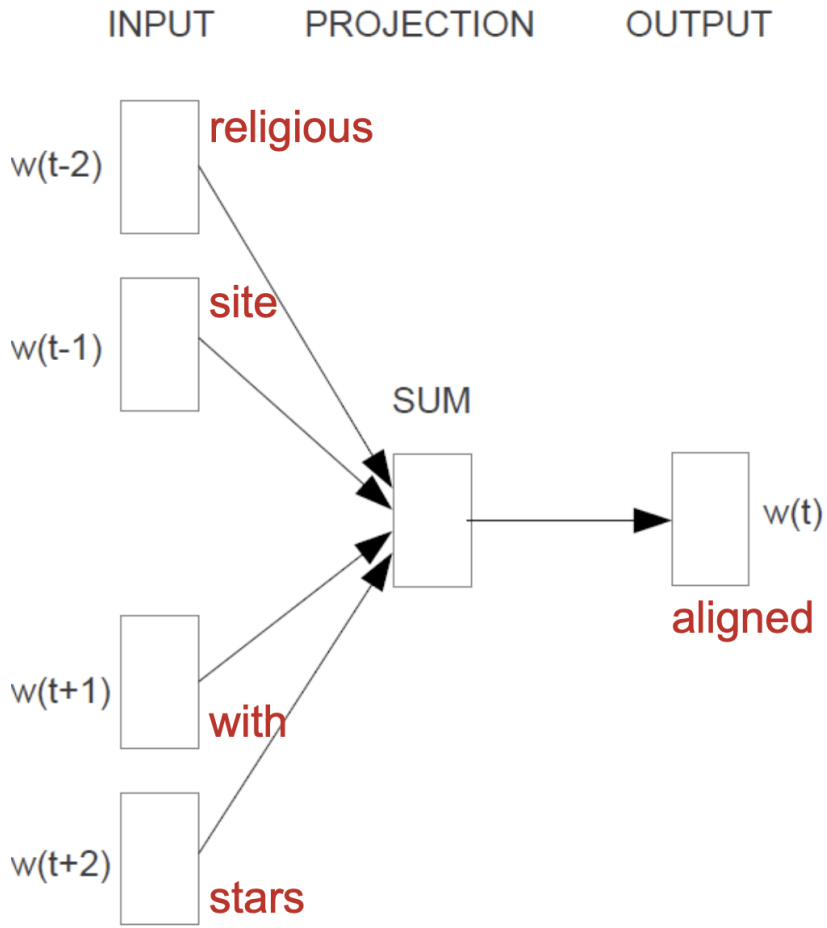


Country-Capital

Training Embeddings

CBOW (Continuous Bag of Words):

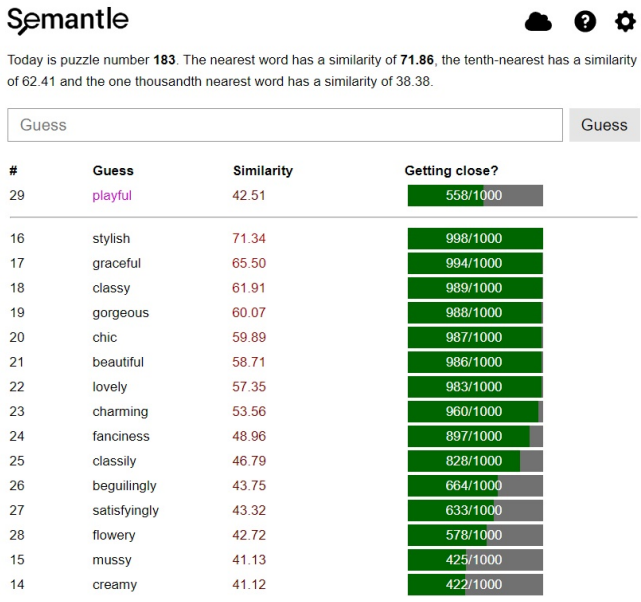
- part of word2vec
- neural network with one hidden layer
- trained on large corpus of NL text (1.6 billion words)
- input examples: sentences with one word missing
- expected output: the missing word
- the weights in the neural network are used as word vectors
- Also: Skip-gram, GloVe, FastText,



What are embeddings good for?

Extremely powerful and widely used, but what does it mean?

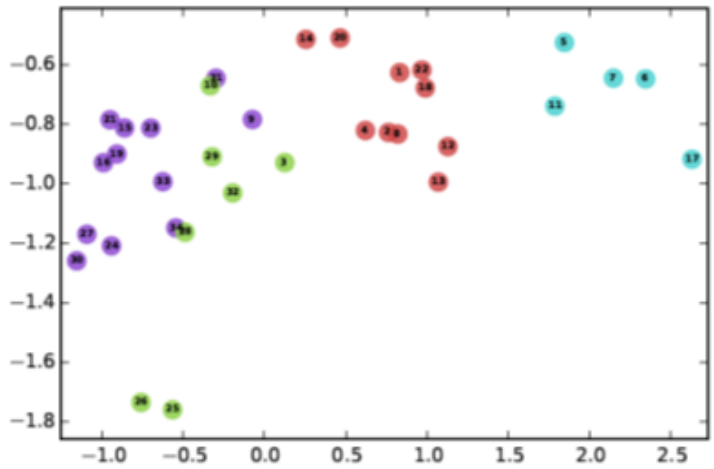
- The distributional hypothesis: *words that occur in the same contexts tend to have similar meanings*
- so, word similarity can be measured in terms of vector similarity?
- this is not true in general:
  synonyms will often appear close to the same words
  but so will many antonyms ("love", "hate")
- syntagmatic similarity: the words are able to combine in sentences with the same other words
- paradigmatic similarity: the words can be substituted with one another



Graph Embeddings

Graph embeddings are similar to word embeddings, but use vectors to represent nodes edges, and/or subgraphs.
Early embeddings like DeepWalk used *random walks* to create words or nodes, and then applied word embedding approaches to get node embeddings.

Input



Output

What are Graph Embeddings good for

Graph completion and validation:

- node classification: given a node which type should it have?
- link prediction: given a node and a edge, what should be at the end?
- relation prediction: given two nodes, which edge type should link them?
- triple classification: given two nodes and an edge, is the triple correct?

Graph (or sub-graph) classification:

- what type of entity/situation/event does the graph represent?
- which class does the graph represent?

Input to deep networks:

- perhaps in combination with text, images, ...
- deep multi-stream networks
- early or late fusion of streams
- context for LLMs and vector databases

Transaltional Embeddings (TransE)

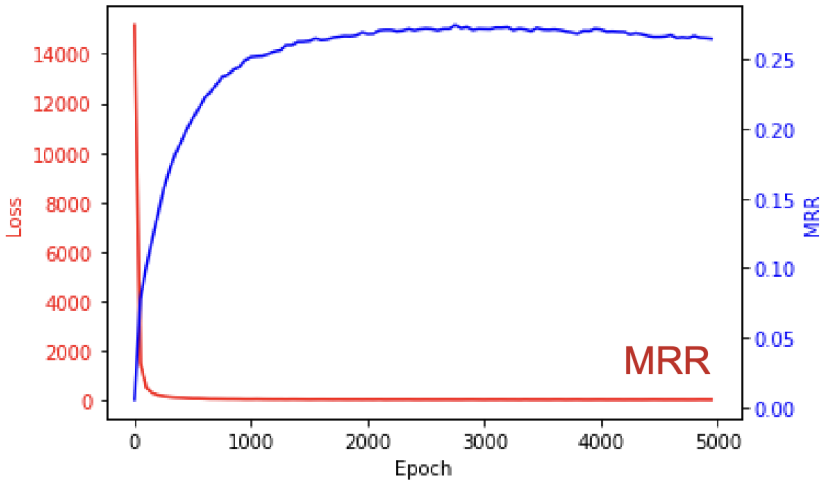
The translational property:

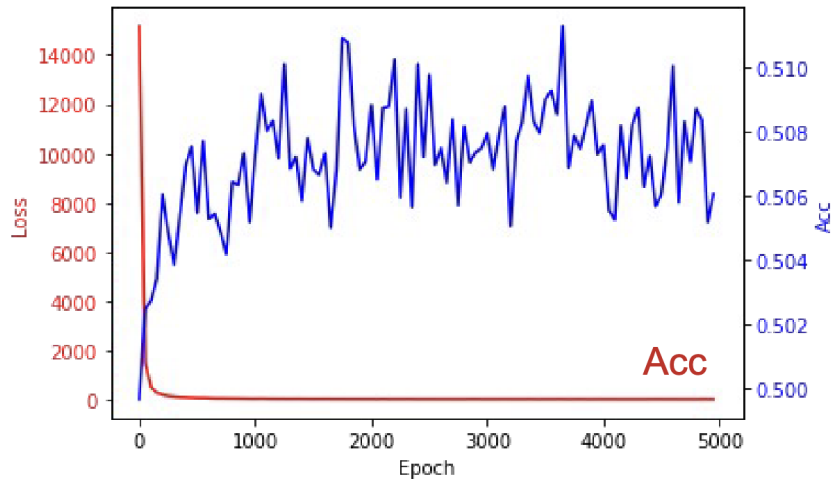
$$if\ (h, r, t) \in KG,\ then\ [h] + [r] \simeq [t]$$

Approach:
start out with random vectors for nodes and edges repeat:

- for each example $(h, r, t) \in KG$, generate a corrupted (h', r, t') that is not in KG (because either h' or t' is changed)
- adjust vectors to
  - minimise $dist([h] + [r], [t])$
  - maximise $dist([h'] + [r], [t'])$
  - loss per example is the difference between $dist([h] + [r], [t])$ and $dist([h'] + [r], [t'])$

Datasets and Training





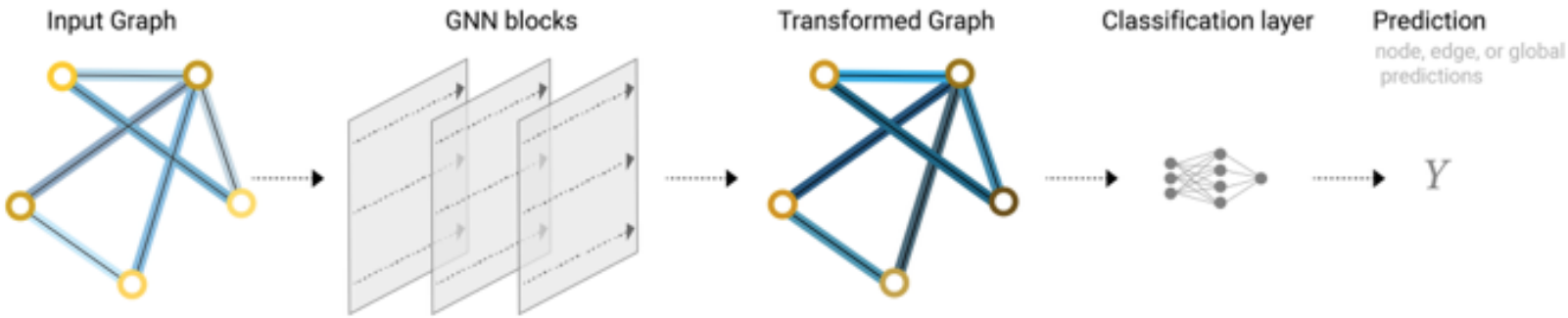
DATA SET	WN	FB15K	FB1M
ENTITIES	40,943	14,951	1×10^6
RELATIONSHIPS	18	1,345	23,382
TRAIN. EX.	141,442	483,142	17.5×10^6
VALID EX.	5,000	50,000	50,000
TEST EX.	5,000	59,071	177,404

Graph Neural Networks

A GNN is an optimizable transformation on all attributes of the graph (nodes, edges, global-context) that preserves graph symmetries (permutation invariances).
GNNs adopt a "graph-in, graph-out" architecture meaning that these model types accept a graph as input, and progressively transform these embeddings, without changing the connectivity of the input graph.
They predict (induce) a knowledge graph from data:

GNN Architectures

GNNs which takes as input a directed graph where nodes and edges are associated with feature vectors, which is recursively updated based on the feature and state vectors of the neighbouring nodes, until convergence. This allows nodes to be classified, or edges to be predicted.



An end-to-end prediction task with a GNN model.

Symbolic Induction

Symbolic Induction creates explainable knowledgebut directly deriving logical terms.
This is done through:

- Rule mining: looking for correspondences with high confidence and support.
- Axiom mining: Learning alowable patterns from ontologies to propose new axioms.
- Hypothesis mining: is the task of finding axioms and rules that naturally describe the graph data. That is they entail a high proportion of positive edges, and do not entail many negative edges.

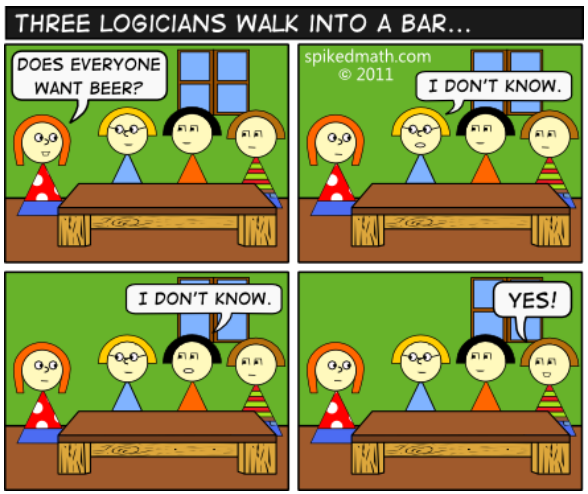
These become explainable rules of thumb.

Inductive information

The inductive process present a very different type of knowledge to the structured logic of the approaches we have repviously considered.

These methods provide insights and intuitions regarding logical structures but there remains a large challenge to integrate these approaches with deductive reasoning, and explainable AI.

Next week, course review!



TI

Three logicians walk into a bar. The bartender says "Would you all like a beer?" The first logician says "I don't know". The second logician says "i don't know" The third logician say "yes!"

This classic logic puzzle is a fun example of how logicians think. Let's break it down:

1. The bartender asks if they all want a beer, which means that if at least one of them wants a beer, they should all get one.
2. The first logician says "I don't know." This means that the first logician does not immediately see whether or not they all want a beer, because they don't know what the others will say.
3. The second logician also says "I don't know." This suggests that the second logician also can't immediately determine if they all want a beer or not.
4. The third logician, however, says "yes!" This is the key to the puzzle. If the third logician wants a beer, it means they have information from the previous two responses to conclude that at least one of the other logicians wants a beer. This is enough for the bartender to serve all three logicians a beer.

So, the third logician's "yes" response takes into account the uncertainty of the first two logicians and ensures that at least one of them wants a beer, prompting the bartender to serve beers to all three.