Knowledge-aware Zero-shot Learning (K-ZSL): Concepts, Methods and Resources

Jiaoyan Chen[1], Yuxia Geng[2], Yufeng Huang[2] and Huajun Chen[2]

1. Department of Computer Science, University of Oxford, UK
2. College of Computer Science and Technology, Zhejiang University, China

https://china-uk-zsl.github.io/kg-zsl-tutorial-iswc-2022/

Tutorial of The 21th International Semantic Web Conference (October 23, 2022, Virtual)

Schedule

Length	Content	Speaker
	Part I - Introduction and Background	
25 mins	ZSL definitions and concepts	Jiaoyan Chen
15 mins	An introduction to KGs	Jiaoyan Chen
15 mins	A brief review on knowledge-aware ZSL	Jiaoyan Chen
10 mins	**Break**	
	Part II - KG-aware ZSL Method	
20 mins	Mapping-based paradigm	Yuxia Geng
20 mins	Data augmentation-based paradigm	Yuxia Geng
20 mins	Propagation-based paradigm	Yuxia Geng
20 mins	Feature fusion paradigm	Jiaoyan Chen
20 mins	Explanation and ZSL	Jiaoyan Chen
10 mins	**Break**	
	Part III - Hands-on: Resources and Benchmarking	
15 mins	A brief review on evaluation resources	Jiaoyan Chen
15 mins	Benchmarking zero-shot image classification	Yuxia Geng
15 mins	Benchmarking zero-shot KG completion	Yuxia Geng
10 mins	Resource demonstration	Yufeng Huang
5 mins	Summary and discussion	Huajun Chen

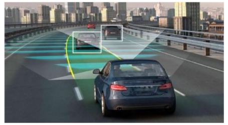
Part I – Introduction and Background

&1

Zero-shot Learning (ZSL) Definitions and Concepts

Deep Learning

- Deep learning is playing a great role
  - Computer vision (CV), natural language understanding (NLP), data science, knowledge engineering and the Semantic Web, etc.
  - A lot of intelligent applications



Self-driving

Chatting bot

Machine Translation

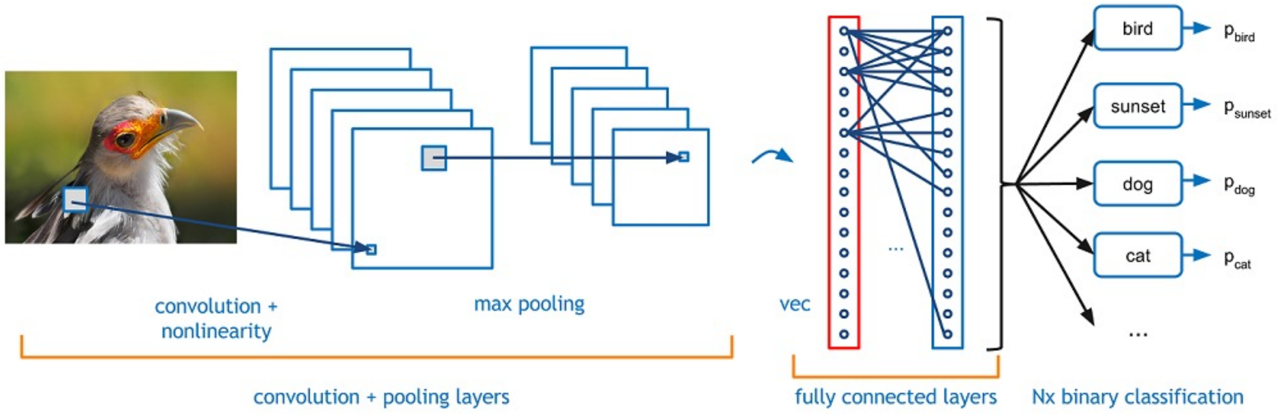
Remote sensing and mapping

Urban Computation

Intelligent Finance

Deep Learning

- Example: Convolutional Neural Networks (CNNs)



(Output: CNN Features)

Image Source: https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

Supervised Learning & Sample Shortage

- Many deep models such as CNNs rely on (semi-)supervised learning with labeled training samples
- However, high quality labeled samples are not always available in many scenarios:
  - Target classes change over time (e.g., new classes emerge)
  - Cannot afford the labour for annotation (e.g., the number of target classes is very large; the annotation is expensive and time consuming)
  - Target classes are rare (e.g., flower of rare breeds)
  - Security and privacy reasons
  - …
- Lack of time and/or computation for re-training

Supervised Learning & Sample Shortage

- Sample shortage has been widely investigated
  - Relevant research problems (or challenges)
    - Domain adaptation
    - Concept drift
    - Long-tailed recognition
    - Few-shot and zero-shot learning
    - …
  - Relevant methods
    - Transfer learning
    - Distant supervision
    - Active learning
    - Meta-learning (learn to learn)
    - Pre-training
    - …

ZSL in Image Classification

- Typical supervised image classification
  - See right example from the CIFAR-10 dataset

ZSL in Image Classification

- Typical supervised image classification
- Early sample shortage research problems:
  - **One-shot learning** which aims to classify objects with just one labeled image [Li et al. 2006];
  - **Few-shot learning**
    - A “few” labeled images

Fei-Fei, Li, Robert Fergus, and Pietro Perona. "One-shot learning of object categories." *IEEE transactions on pattern analysis and machine intelligence* 28.4 (2006): 594-611.

ZSL in Image Classification

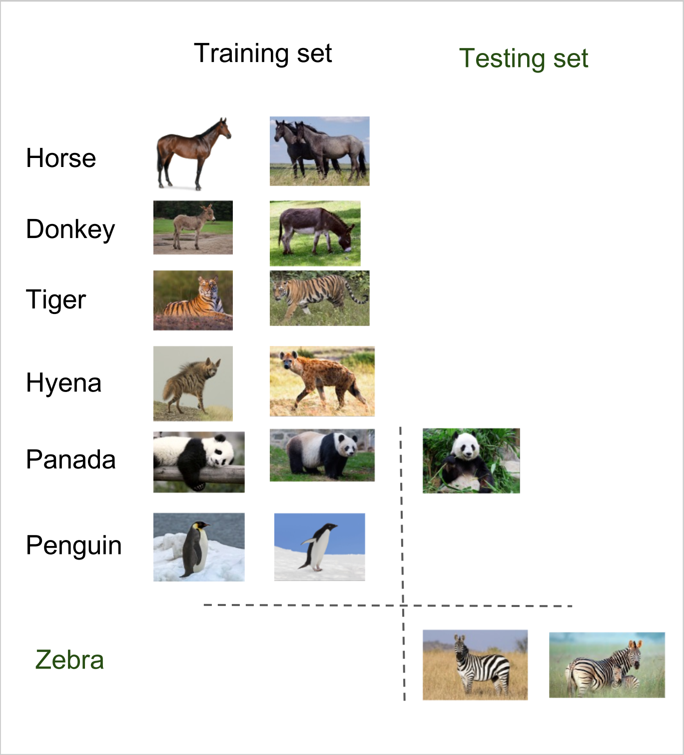
- Early sample shortage concepts:
  - One-shot learning which aims to classify objects with just one labeled image [Li et al. 2006];
  - Few-shot learning
- **Zero-shot Learning**
  - Classify objects with NO labeled images
  - [Palatucci et al. 2009], [Lampert et al. 2009], etc.

Palatucci, Mark, et al. "Zero-shot learning with semantic output codes." *Advances in neural information processing systems* 22 (2009).

Lampert, Christoph H., Hannes Nickisch, and Stefan Harmeling. "Learning to detect unseen object classes by between-class attribute transfer." *2009 IEEE conference on computer vision and pattern recognition*. IEEE, 2009.

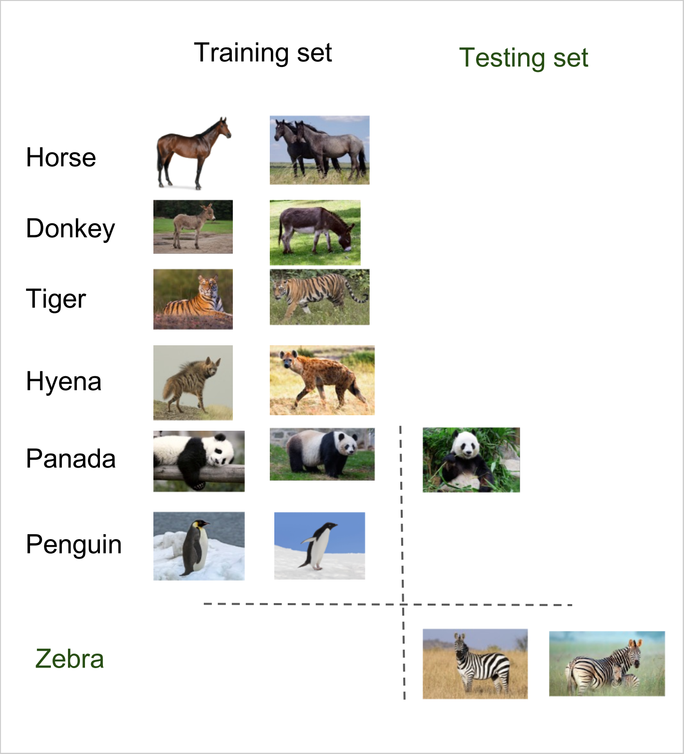
ZSL in Image Classification

- **Zero-shot Learning**
  - Classify objects with NO labeled images
  - [Palatucci et al. 2009], [Lampert et al. 2009], etc.
  - **Example** on the right



ZSL in Image Classification

- **Zero-shot Learning**
  - Classify objects with NO labeled images
  - [Palatucci et al. 2009], [Lampert et al. 2009], etc.
  - **Example** on the right
- Key concepts
  - **Seen classes**
    - with labeled images for training, e.g., Panada
  - **Unseen classes**
    - With images to predict but without labeled images for training e.g., Zebra
  - **Standard ZSL**
    - Predict images of unseen classes in testing
  - **Generalized ZSL**
    - Predict images of both seen classes and unseen classes in testing



ZSL Tasks in NLP

- Text classification
  - E.g., online topic classification with **emerging topics** (unseen classes that have never appeared in model training)
  - E.g., Clinical coding with **new clinical concepts** e.g., from ICD-9 to ICD-10
  - E.g., User attribute (e.g., profession) prediction with **new attribute values**

Dong H. et al. Automated clinical coding: what, why, and where we are?. npj Digital Medicine.

ZSL Tasks in NLP

- Text classification
  - E.g., online topic classification with **emerging topics** (unseen classes that have never appeared in model training)
  - E.g., Clinical coding with **unseen clinical codes**
  - E.g., User attribute (e.g., profession) prediction with **new attribute values**

- Knowledge extraction from text (a.k.a. Open Information Extraction)
  - E.g., relation classification with **unseen relations** (i.e., those relations that have no relation mentions appeared in training)
  - Entity linking with **unseen entities**
  - Event extraction

ZSL Tasks in NLP

- Text classification
  - E.g., online topic classification with **emerging topics** (unseen classes that have never appeared in model training)
  - E.g., Clinical coding with **unseen clinical codes**
  - E.g., User attribute (e.g., profession) prediction with **new attribute values**
- Knowledge extraction from text (a.k.a. Open Information Extraction)
  - E.g., relation classification with **unseen relations** (i.e., those relations that have no relation mentions appeared in training)
  - Entity linking with **unseen entities**
  - Event extraction
- Others:
  - **Unseen answers** in question answering
  - **Unseen tokens** in word embedding (out-of-vocabulary problem)
  - Etc.

ZSL Tasks in VQA

- Visual Question Answering (VQA)
  - ZSL definition 1: **Unseen answers**
  - ZSL definition 2: **Unseen words** in the question and/or answer

Question: Where might a person dress like this?

Answer: Office

- Out-of-knowledge VQA
  - Require reasoning over external knowledge to give the answer

Question: What vehicle uses this item?

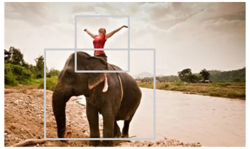
Answer: Firetruck

“firetruck is some truck that needs water supply from fire hydrant”

ZSL Tasks in Scene Graph

- Scene graph extraction
  - Extract triples (a.k.a. relations in this domain) from an image
  - **Unseen triples**
    - Training with images with *girl ride animal* and *woman ride elephant*
    - Testing with images with dog *ride bike*

(girl, ride, horse)



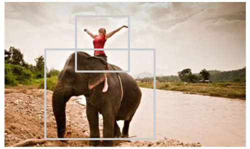
(woman, ride, elephant)

(dog, ride, bike)

ZSL in Scene Graph

- Scene graph extraction
  - Extract triples (a.k.a. relations in this domain) from an image
  - **Unseen triples**
    - Training with examples about *(girl, ride, animal)* and *(woman, ride, elephant)*
    - Testing with images with *(dog, ride, bike)*
  - **Unseen predicates** or **unseen object types**
    - Training with images with *ride*
    - Testing with images with *sit on*

(girl, ride, horse)



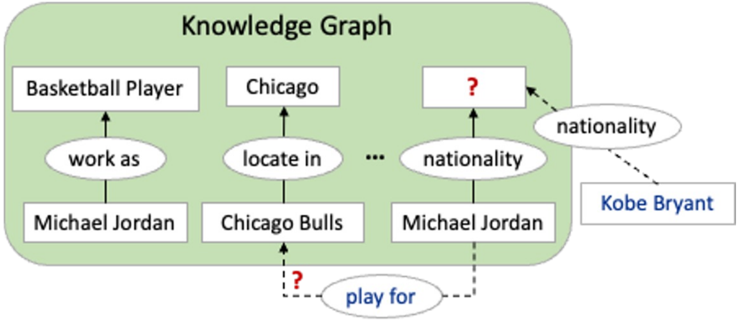
(woman, ride, elephant)

(dog, ride, bike)

(child, sit on, bike)

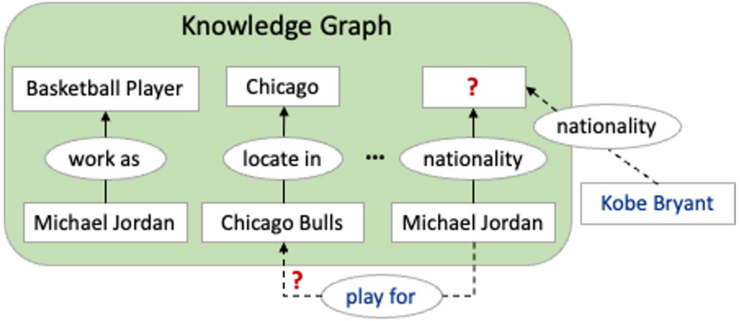
ZSL in KG Refinement

- Knowledge graph (KG) link prediction
  - Predict facts (triples) in a KG
  - **Emerging relations or entities** that have never appeared in the original KG triples used for learning the embeddings
    - E.g., the new relation play for and the new entity Kobe Bryant in the right example
    - Sometimes known as inductive KG completion



ZSL in KG Refinement

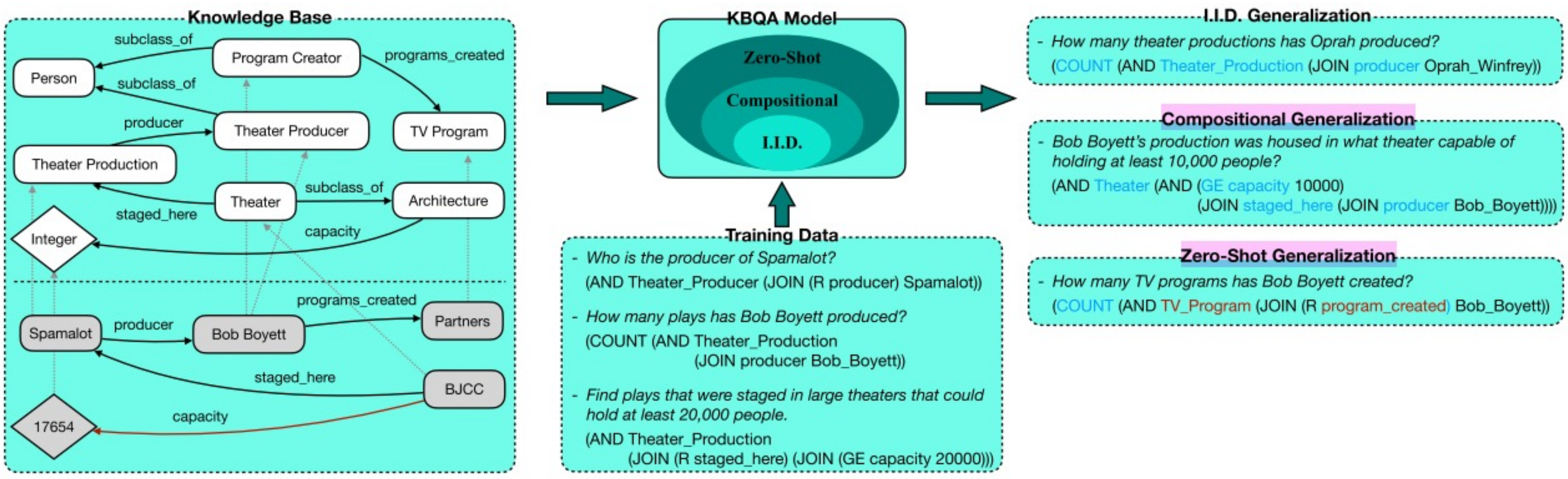
- Knowledge graph (KG) link prediction
  - Predict facts (triples) in a KG
  - **Emerging relations or entities** that have never appeared in the original KG triples used for learning the embeddings
    - E.g., the new relation play for and the new entity Kobe Bryant in the right example
    - Sometimes known as inductive KG completion



- Other relevant tasks
  - Entity typing with emerging types
  - Populating KG by tabular data e.g., table column annotation with unseen column types

ZSL in KG Question Answering

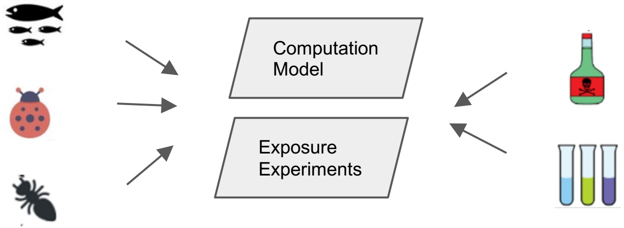
- Transforming natural language questions to formal queries (e.g., s-expressions and SPARQL queries) with unseen schema items (e.g., relations) and/or unseen schema iterm combinations with logical operations



Example from Gu, Yu, et al. "Beyond IID: three levels of generalization for question answering on knowledge bases." *Proceedings of the Web Conference 2021*. 2021.

ZSL in Bioinformatics & Molecule Learning

- Bioinformatics and Molecule Learning
  - E.g., Protein function prediction with unseen functions (classes)
  - E.g., Ecotoxicological effect prediction with new chemical components or species that have no experimental risk assessment observations for model training
    - Risk assessment experiments for different species and chemical components are costly and suffer from ethic issues; the current observations (e.g. the ECOTOX database) are quite incomplete



Myklebust, Erik B., et al. "Prediction of adverse biological effects of chemicals using knowledge graph embeddings." Semantic Web Preprint (2022): 1-40.

Knowledge-aware ZSL & Side information

- **No samples for the unseen classes!**

- A mainstream solution: using **Side information** (a.k.a. **external or auxiliary knowledge**) to bridge the seen classes and the unseen classes, thus enabling the model transfer
    - Mapping function e.g., side information → the class's model parameters
    - Generation model e.g., generating samples conditioned on the side information
    - Graph propagation e.g., transfer classes' model parameters via propagation over graph-structured side information
    - ... (more method details will be introduced later)

Side information

- **Side information** (a.k.a. **external or auxiliary knowledge**) to bridge the seen classes and the unseen classes, thus enabling the model transfer
  - Textual description:

"***Zebras** are white animals with black stripes, they have larger, rounder ears than horses ...*"

Sometimes simple name information also contains important semantics, e.g., two relations "has office in" and "has headquarter in".

Side information

- **Side information** (a.k.a. **external or auxiliary knowledge**) to bridge the seen classes and the unseen classes, thus enabling the model transfer

  - Textual description:

"***Zebras*** *are white animals with black stripes, they have larger, rounder ears than horses ...*"

  - Attributes e.g., visual annotations:

black: yes
white: yes
brown: no
stripes: yes
water: no
…

black: yes
white: yes
brown: yes
stripes: yes
water: no
…

Annotations could be associated with binary values for existence or real value for degree

Side information

- **Side information** (a.k.a. **external or auxiliary knowledge**) to bridge the seen classes and the unseen classes, thus enabling the model transfer

  - Textual description:

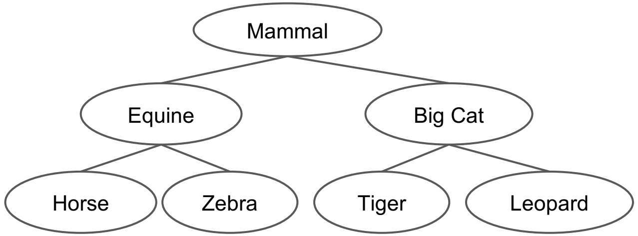
*“**Zebras** are white animals with black stripes, they have larger, rounder ears than horses ...”*

  - Attributes e.g., visual annotations:

black: yes
white: yes
brown: no
stripes: yes
water: no
…

black: yes
white: yes
brown: yes
stripes: yes
water: no
…

  - Graph structured relationships e.g., taxonomy:



A simplified demonstration of the animal taxonomy tree

Side information

- **Side information** (a.k.a. **external or auxiliary knowledge**) to bridge the seen classes and the unseen classes, thus enabling the model transfer
  - Textual description:

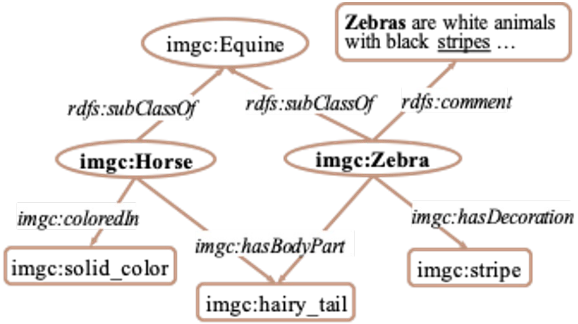
*“**Zebras** are white animals with black stripes, they have larger, rounder ears than horses ...”*

  - Attributes e.g., visual annotations:

black: yes
white: yes
brown: no
stripes: yes
water: no
…

black: yes
white: yes
brown: yes
stripes: yes
water: no
…

  - Graph structured relationships e.g., taxonomy
  - Knowledge Graphs (relational facts, categories, literals, etc.):



Side information

- **Side information** (a.k.a. **external or auxiliary knowledge**) to bridge the seen classes and the unseen classes, thus enabling the model transfer

  - Textual description:

"***Zebras*** *are white animals with black stripes, they have larger, rounder ears than horses ...*"

  - Attributes e.g., visual annotations:

black: yes
white: yes
brown: no
stripes: yes
water: no
…

black: yes
white: yes
brown: yes
stripes: yes
water: no
…

  - Graph structured relationships e.g., taxonomy
  - Knowledge Graphs (relational facts, categories, literals, etc.):
  - Logical relationships & rules:

"$Zebra \sqsubseteq Equine \sqcap \exists hasTexture.Stripes \sqcap \exists hasHabitat.Meadow$ ... "
"$hasUncle \equiv hasParent \circ hasBrother$"

In Description Logics

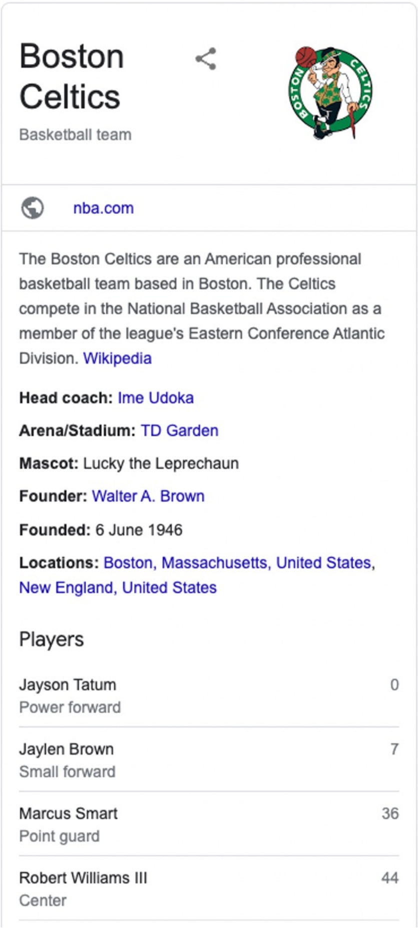
&2

An Introduction to Knowledge Graphs

The Term of “Knowledge Graph”

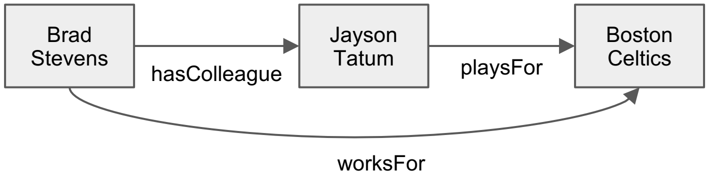
- The Knowledge Graph is a knowledge base used by **Google** and its services to enhance its search engine's results with knowledge gathered from a variety of sources.
  - Proposed around 2012
  - Knowledge ≈ **Instances + Facts**
  - KG ≈ **Linked Structured Data** (can be regarded as a multi-relational graph)

example

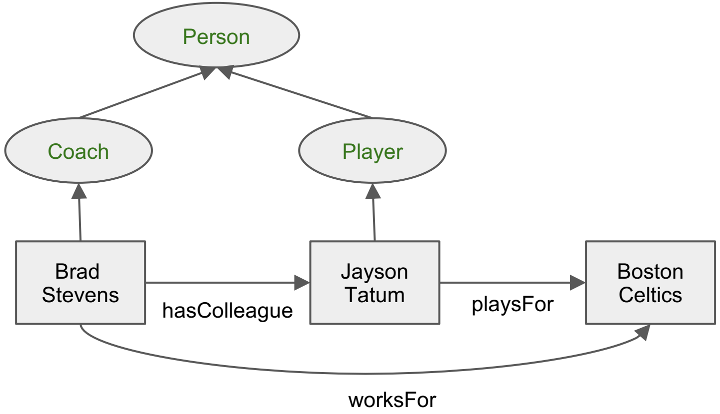


A Semantic Web Perspective

- **RDF** (Resource Description Framework)
  - Triple: <Subject, Predicate, Object>
  - Representing facts (data):
    - E.g., <Jayson Tatum, playsFor, Boston Celtics>



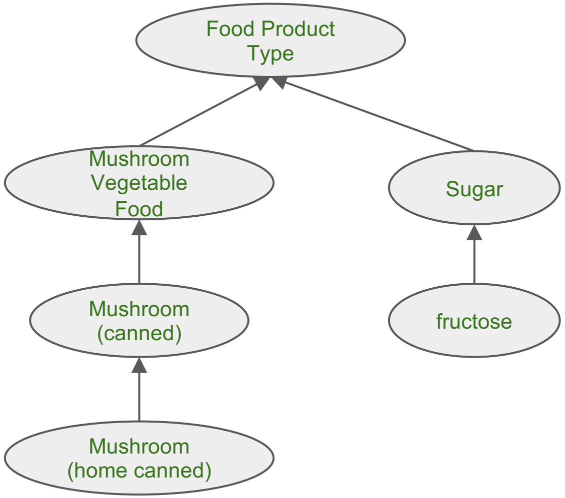
A Semantic Web Perspective



- RDF (Resource Description Framework)
  - Triple: <Subject, Predicate, Object>
  - Representing facts (data):
    - E.g., <Jayson Tatum, playsFor, Boston Celtics>
- **RDF Schema**
  - Meta data (schema) of instances and facts
    - E.g., class (concept), property, property domain and range

A Semantic Web Perspective

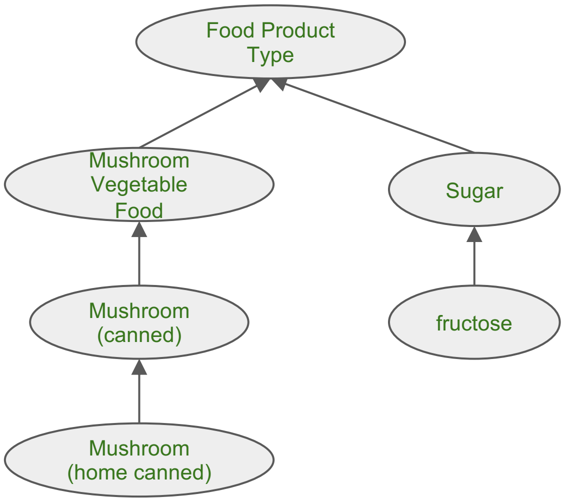
- RDF (Resource Description Framework)
  - Triple: <Subject, Predicate, Object>
  - Representing facts:
    - E.g., <Jayson Tatum, playsFor, Boston Celtics>
- RDF Schema
  - Meta data (schema) of instances and facts
    - E.g., class (concept), property, property domain and range
- **Web Ontology Language** (OWL)
  - Schema, constraints and logical relationships
    - E.g., ‘food material’ ≡ ‘environmental material’ *and* (‘has role’ *some* ‘food’)
    - E.g., the max. cardinality of “playsFor” is 1
  - Taxonomies and vocabularies
    - Formal, explicit, shared, conceptualization



A segment of the hierarchical classes of the food ontology FoodOn

A Semantic Web Perspective

- RDF (Resource Description Framework)
  - Triple: <Subject, Predicate, Object>
  - Representing facts:
    - E.g., <Jayson Tatum, playsFor, Boston Celtics>
- RDF Schema
  - Meta data (schema) of instances and facts
    - E.g., class, property domain and range
- **Web Ontology Language** (OWL)
  - Schema, constraints and logical relationships
    - E.g., 'food material' ≡ 'environmental material' *and* ('has role' *some* 'food')
    - E.g., the max. cardinality of "playsFor" is 1
  - Taxonomies and vocabularies
    - Formal, explicit, shared, conceptualization

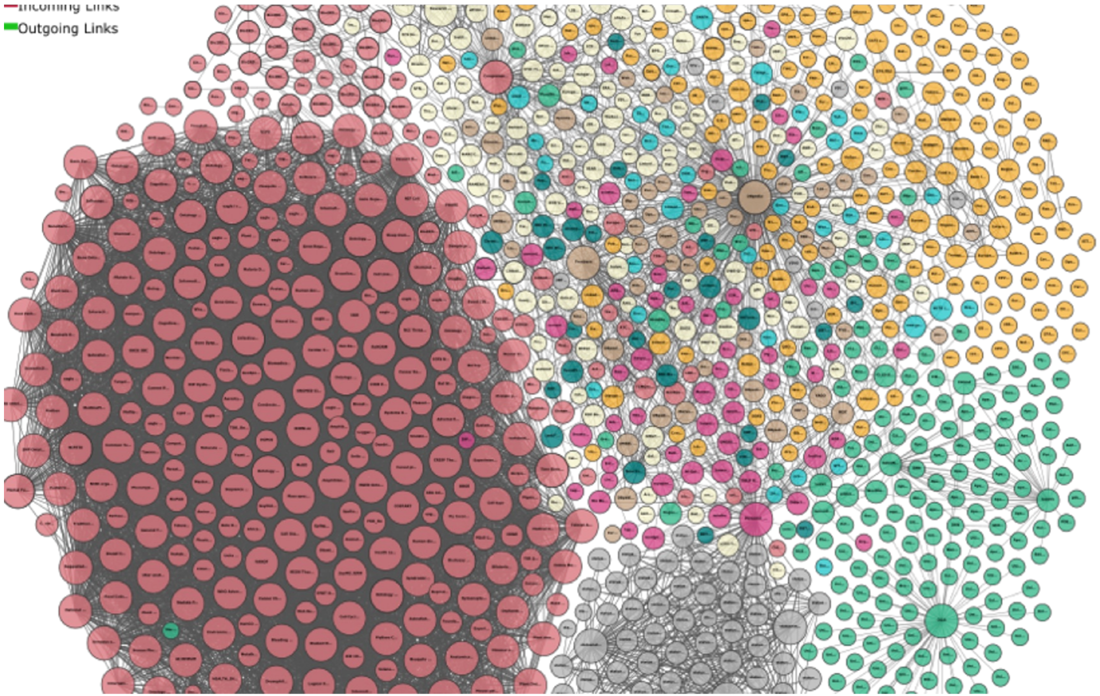


A segment of the hierarchical classes of the food ontology FoodOn

What is KG?
RDF facts? RDF facts + schema? Ontology?

Linked Data Perspective

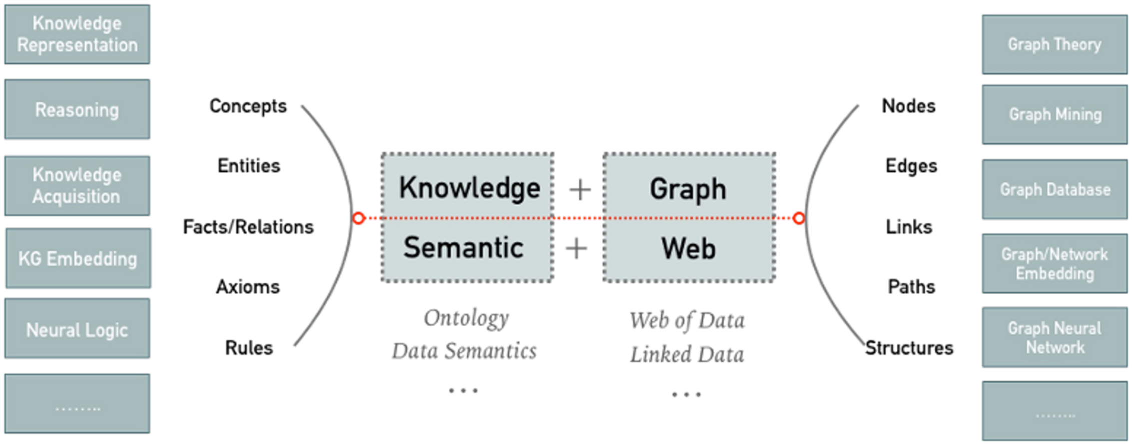
- Linking data with knowledge representation and semantic technologies
  - URI
  - RDF
  - SPARQL
  - Etc.



Linked Open Data

AI Perspective: Knowledge + Graph

- KG is more expressive than pure graph with knowledge representation
  - Support graph algorithms e.g., graph embeddings and graph neural networks
  - Support reasoning with formal logics



Why use a KG?

- Intuitive (e.g., no “foreign keys”)
- Data + schema (ontology)
- IRI/URI not strings (explicit)
- Flexible & extensible
- Rule language
  - Location + capital → location
  - Parent + brother → uncle
- Other kinds of query
  - Navigation
    - Similarity & Locality

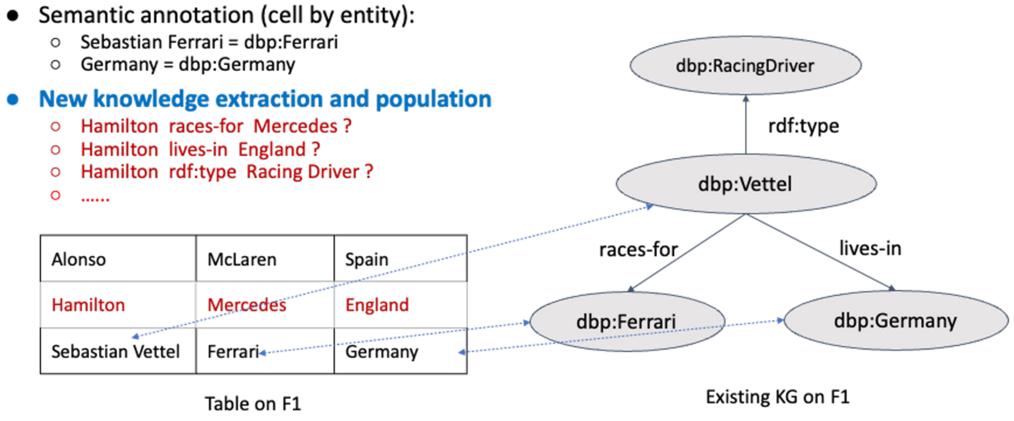
(Slide from Ian Horrocks)

KG Construction

- Crowdsourcing and Encyclopedias
  - DBpedia, Wikidata, Zhishi.me, LinkedGeoData …
- Tabular data
  - DBs, Web Tables, Excel Sheets, CSV files …
- **The Web mining**
  - NELL (Never-ending Language Learning)
- **Natural language understanding**
  - Open Information Extraction
- **Knowledge Integration, Refinement and Curation**
- **Personalized (or customized) KG Construction**

KG Construction from Tabular Data

- **Tabular data**
  - DBs by e.g., OBDA techniques
  - Web Tables, Excel Sheets, CSV files … by e.g., table to KG matching



A toy example on matching tables with KG and new knowledge extraction

AD: SemTab 2022 (Semantic Web Challenge on Tabular Data to Knowledge Graph Matching) in conjunction with ISWC'22

KG Applications

- Search engines (e.g., Google KG)
- Search, browse and recommendation in e-Commerce (e.g., Amazon Product KG)
- Personal assistants (e.g., Apple Siri, Amazon Alex)
- Clinical AI
- Smart city & IoT
- Machine Learning and Neural-symbolic Integration
- …

&3

A Brief Review on Knowledge-aware ZSL

Jiaoyan, Chen, Geng Yuxia, Chen Zhuo, Horrocks Ian, Pan Jeff Z., Chen Huajun. "Knowledge-aware Zero-Shot Learning: Survey and Perspective." IJCAI 2021 Survey Track.

Jiaoyan, Chen, Geng Yuxia, et al. "Low-resource Learning with Knowledge Graphs: A Comprehensive Survey." arXiv preprint arXiv:2112.10006 (2021).

Problem and Setting

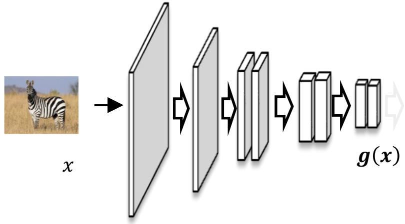
- Given the input x and the output (class) y , the general task is to learn a function (model) $f: x \rightarrow y$

- In the standard ZSL setting
  - The training data for learning f is denoted $D_{tr} = \{(x, y) | x \in \mathcal{X}_{tr}, y \in \mathcal{Y}_s\}$
  - The test data for evaluating f is denoted as $D_{te} = \{(x, y) | x \in \mathcal{X}_{te}, y \in \mathcal{Y}_u\}$
  - $\mathcal{Y}_s \cap \mathcal{Y}_u = \emptyset$

- In the generalised ZSL setting
  - $D_{te} = \{(x, y) | x \in \mathcal{X}_{te}, y \in \mathcal{Y}_u \cup \mathcal{Y}_s\}$

Problem and Setting

- x and y are often encoded
  - Denoted as $\boldsymbol{g}(\boldsymbol{x})$ and $\boldsymbol{h}(\boldsymbol{y})$ respectively
  - E.g., in image classification, g could be a Convolutional Neural Network, h could be simple dummy encoding



Convolutional and pooling layers

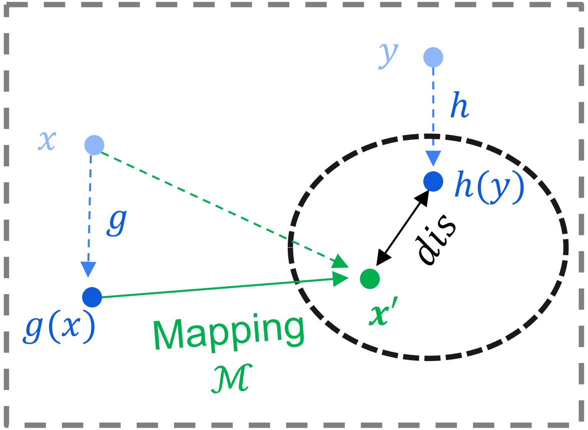
y	$\boldsymbol{h}(\boldsymbol{y})$
Zebra	[1, 0, 0, 0]
Horse	[0, 1, 0, 0]
Tiger	[0, 0, 1, 0]
Panda	[0, 0, 0, 1]

  - E.g., in KG link prediction which predict the relation given two entities, g and h could be the KG entity and relation embeddings

Problem and Setting

- How to utilize the external knowledge?
  - Considered in class encoding $\boldsymbol{h}(\boldsymbol{y})$
    - E.g., word embedding for text, and dummy encoding for attributes
  - D_{tr} could be optionally used for learning parameters of the prediction model components (e.g., $\boldsymbol{g}$, $\boldsymbol{h}$, classifier)
  - We categorize the methods into four kinds:
    - Mapping-based paradigm
    - Propagation-based paradigm
    - Data augmentation paradigm
    - Class feature paradigm

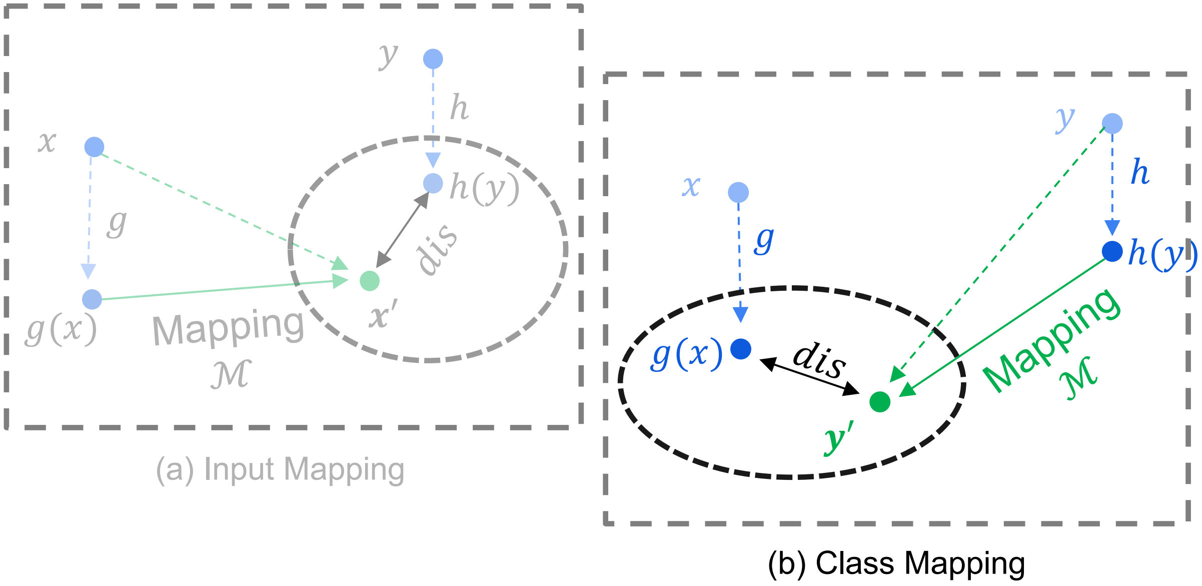
Mapping-based Paradigm



(a) Input Mapping

- The mapping function $\mathcal{M}$ maps the input x or $g(x)$ to its class's vector encoding $h(y)$
- Training: the function $\mathcal{M}$ is learned from D_{tr} by minimizing the distance between x' and $h(y)$
- Prediction: a testing sample is mapped and compared with the vectors of candidate classes $\mathcal{Y}_u$ (or $\mathcal{Y}_u \cup \mathcal{Y}_s$)

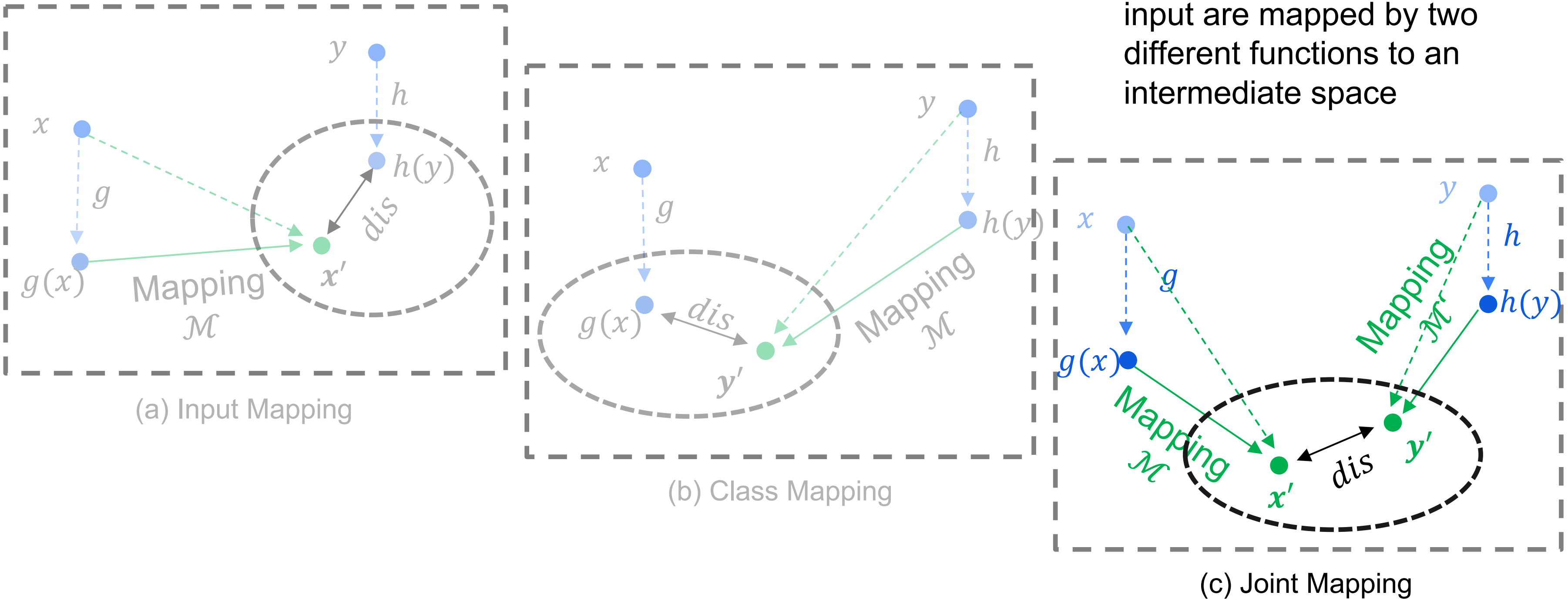
Mapping-based Paradigm



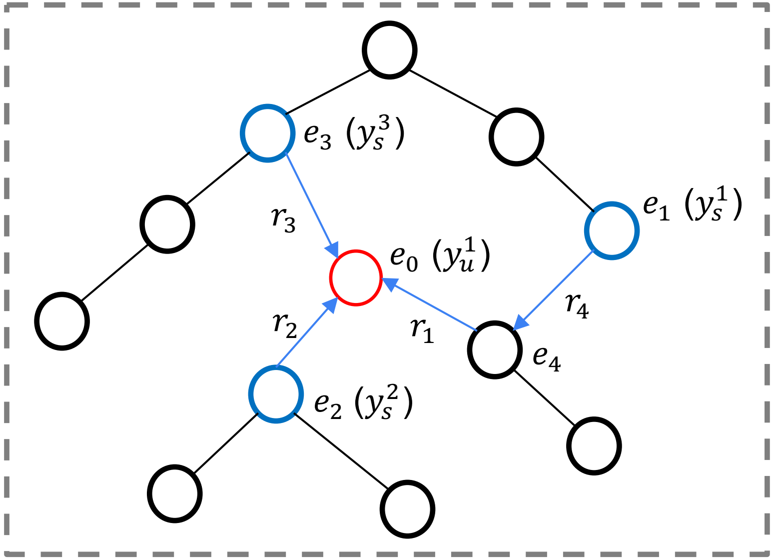
- The class (external knowledge) y or its encoding $h(y)$ is mapped to the vector of its sample $g(x)$

Mapping-based Paradigm

- Both the class label and the input are mapped by two different functions to an intermediate space

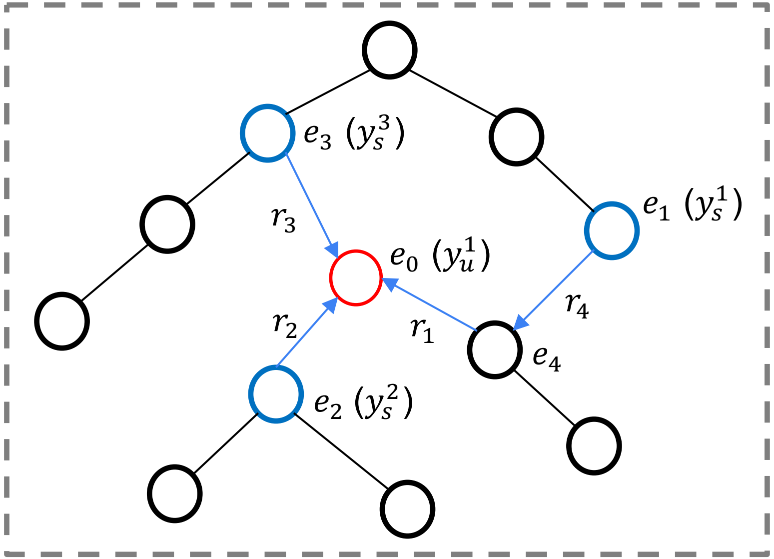


Propagation-based Paradigm



- **Model parameter propagation**
  - Classes are aligned with graph nodes (e.g., e_3 -- y_s^3)), while the graph is built with the external knowledge
  - The parameters of the models of the seen classes e.g., y_s^3, which are learned from D_{tr} , are regarded as node features
  - Model parameters are propagated to unseen classes (e.g., y_u^1) to estimate their model parameters via e.g., Graph Convolutional Networks

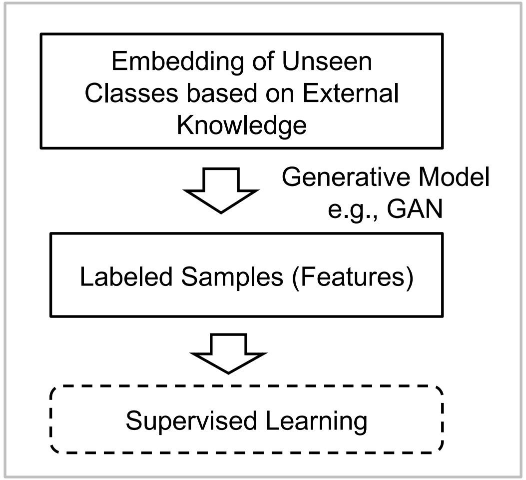
Propagation-based Paradigm



- **Class belief propagation**
  - Usually for zero-shot multi-label classification e.g., a scene image with multiple objects to recognize
  - Classes are aligned with graph nodes (e.g., e_3 -- y_s^3)), while the graph is built with the external knowledge
  - The scores of seen classes, predicted by models trained by D_{tr}, are propagated to estimate the scores of unseen classes

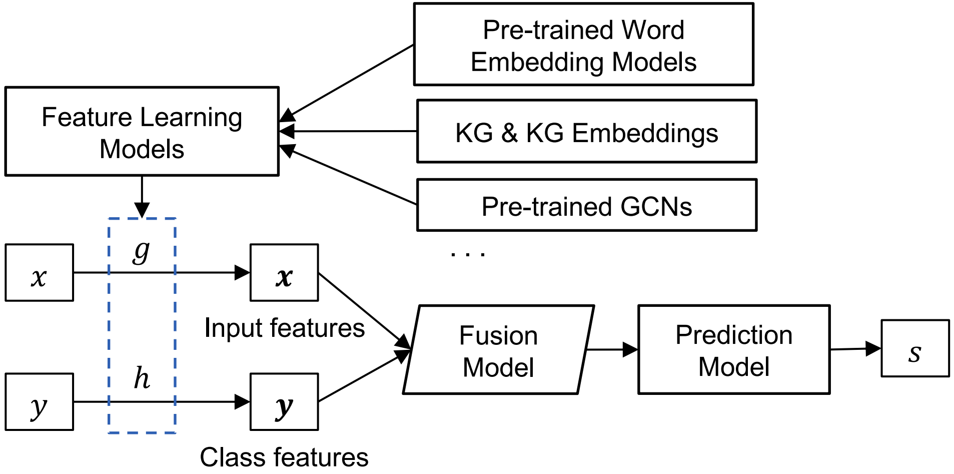
Data Augmentation Paradigm

- Automatically generate training data for unseen classes
  - Rules e.g., inferring additional facts for KG link prediction with unseen relations
  - Generative models



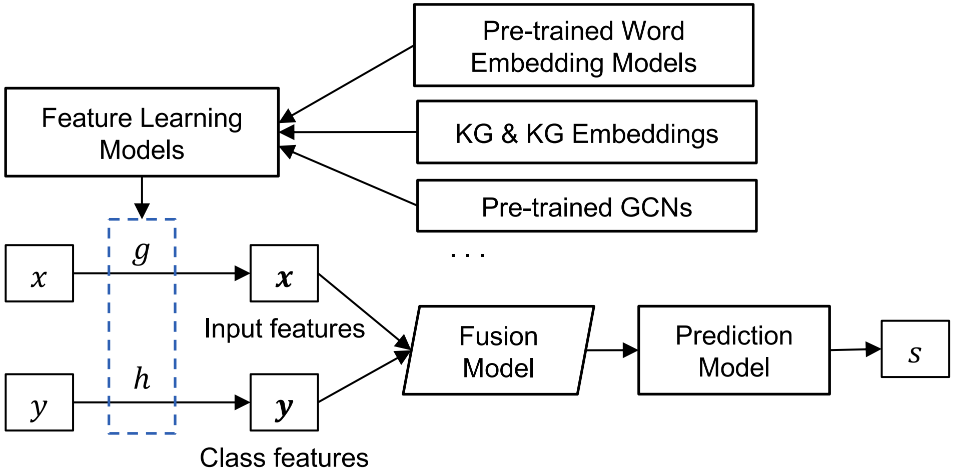
Class Feature Paradigm

- Class encoding $g(x)$ and input encoding $h(y)$ are fed into one model
- The model predicts a score which indicates whether the input x matches the class y



Class Feature Paradigm

- Class encoding $g(x)$ and input encoding $h(y)$ are fed into one model
- The model predicts a score which indicates whether the input x matches the class y
- Typical case: text class feature
    - E.g., in zero-shot KG link prediction, where unseen entities or relations are described by text; the text embeddings can be directly used to predict the triple



A Summary of External Knowledge (Side Information)

Category	Description	Embedding	Semantic Richness	Summary
Text	Unstructured text that describes the classes, such as class names, phrases, sentences and documents	Word embedding, text feature learning	Weak	**Very easy to access**; words are often **ambiguous**; long text is usually **noisy**

Rely on feature extraction by e.g., TF-IDF, and joint text feature learning with the classifier

A Summary of External Knowledge

Category	Description	Embedding	Semantic Richness	Summary
Text	Unstructured text that describes the classes, such as class names, phrases, sentences and documents	Word embedding, text feature learning	Weak	**Very easy to access**; words are often **ambiguous**; long text is usually **noisy**
Attribute	Semi-structured class properties with categorical, Boolean or real values, such as annotations that describe object visual characteristics	Vectors with binary or numeric values	Medium	Attributes by **manual annotation** are accurate but **very costly**

A Summary of External Knowledge

Category	Description	Embedding	Semantic Richness	Summary
Text	Unstructured text that describes the classes, such as class names, phrases, sentences and documents	Word embedding, text feature learning	Weak	**Very easy to access**; words are often **ambiguous**; long text is usually **noisy**
Attribute	Semi-structured class properties with categorical, Boolean or real values, such as annotations that describe object visual characteristics	Vectors with binary or numeric values	Medium	Attributes by **manual annotation** are accurate but **very costly**
Knowledge Graph	Multi-relation graph composed of entities aligned with the classes, other entities and their relationships such as the subsumption and the relational facts	KG Embedding methods e.g., GNNs and Translation-based	High	KGs can also encompass the **text and attribute** external knowledge; some **open KGs can be re-used**

A Summary of External Knowledge

Category	Description	Embedding	Semantic Richness	Summary
Text	Unstructured text that describes the classes, such as class names, phrases, sentences and documents	Word embedding, text feature learning	Weak	**Very easy to access**; words are often **ambiguous**; long text is usually **noisy**
Attribute	Semi-structured class properties with categorical, Boolean or real values, such as annotations that describe object visual characteristics	Vectors with binary or numeric values	Medium	Attributes by **manual annotation** are accurate but **very costly**
Knowledge Graph	Multi-relation graph composed of entities aligned with the classes, other entities and their relationships such as the subsumption and the relational facts	KG Embedding methods e.g., GNNs and Translation-based	High	KGs can also encompass the **text and attribute** external knowledge; some **open KGs can be re-used**
Ontology & Rule	Logical relationships between the classes (and other concepts), such as the subsumption, the quantification constraints and the composition	Ontology embedding e.g., OWL2Vec*, materialization	Very high	Ontologies **include KGs** (as ABoxes) and can encompass the text and attributes; construction of **logics relies on domain knowledge**

See [Chen et al. 2021 IJCAI] for citations of different external knowledge

Conclusion

- Sample shortage in deep learning, ZSL concepts and definitions, ZSL scenarios
- Knowledge graph (KG) and KG construction
- External knowledge (including KGs) in ZSL
- Four different paradigms for knowledge-aware ZSL

Thanks!

Contacts:

Jiaoyan Chen (jiaoyan.chen@cs.ox.ac.uk)
Yuxia Geng (gengyx@zju.edu.cn)