Zero-shot and Few-shot Learning with Knowledge Graphs: A Comprehensive Survey

Jiaoyan Chen*, Yuxia Geng, Zhuo Chen, Jeff Z. Pan, Yuan He, Wen Zhang, Ian Horrocks and Huajun Chen*

***Abstract*—Machine learning especially deep neural networks have achieved great success but many of them often rely on a number of labeled samples for supervision. As sufficient labeled training data are not always ready due to e.g., continuously emerging prediction targets and costly sample annotation in real world applications, machine learning with sample shortage is now being widely investigated. Among all these studies, many prefer to utilize auxiliary information including those in the form of Knowledge Graph (KG) to reduce the reliance on labeled samples. In this survey, we have comprehensively reviewed over 90 papers about KG-aware research for two major sample shortage settings — zero-shot learning (ZSL) where some classes to be predicted have no labeled samples, and few-shot learning (FSL) where some classes to be predicted have only a small number of labeled samples that are available. We first introduce KGs used in ZSL and FSL as well as their construction methods, and then systematically categorize and summarize KG-aware ZSL and FSL methods, dividing them into different paradigms such as the mapping-based, the data augmentation, the propagation-based and the optimization-based. We next present different applications, including not only KG augmented prediction tasks such as image classification, question answering, text classification and knowledge extraction, but also KG completion tasks, and some typical evaluation resources for each task. We eventually discuss some challenges and open problems from different perspectives.**

***Index Terms*—Knowledge Graph, Zero-shot Learning, Few-shot Learning, Sample Shortage, Inductive Knowledge Graph Completion.**

I. Introduction

Machine learning (ML) especially deep learning is playing an increasingly important role in artificial intelligence (AI), and has achieved great success in many domains and applications in the past decades. For example, Convolutional Neural Networks (CNNs) can often achieve even higher accuracy than human beings in image classification and visual object recognition, leading to the fast development of applications such as self-driving vehicles, face recognition, handwriting recognition, image retrieval and remote sensing image processing; Recurrent Neural Networks (RNNs) and Transformer-based models are successful in sequence learning and natural language understanding, which boost applications such as machine translation, speech recognition and chatbots; Graph Neural Networks (GNNs) have been widely applied to prediction tasks involving graph structured data in domains such as social networks, chemistry and biology.

However, the high performance of most ML models relies on a number of labeled samples for (semi-)supervised learning, while such labeled samples are often costly or not efficient enough to collect in real-word applications. Even when labeled samples can be collected, re-training a complex model from scratch when new prediction targets (e.g., classification labels) emerge is unacceptable in many contexts where real-time is required or enough computation resource is inaccessible. All these situations will lead to *sample shortage* in ML. In the paper, we review two major sample shortage settings: *zero-shot learning* (ZSL) and *few-shot learning* (FSL). ZSL is formally defined as predicting new classes (labels) that have never appeared in training, where the new classes are named as *unseen classes* while the classes that have samples in training are named as *seen classes* [1, 2, 3]. FSL is to predict new classes for which only a small number of labeled samples are given [4][5]. For convenience, we also call such new classes with insufficient labeled samples as unseen classes, and the other classes that have a large number of samples used in training as seen classes. Specially, when the unseen class has only one labeled sample, FSL becomes *one-shot learning* [5].

ZSL has attracted wide attention in the past decade with quite a few solutions proposed [6, 7, 8]. One common solution is transferring knowledge which could be samples, features (data representations) and model parameters from seen classes to unseen classes so as to avoid only learning features from labeled samples and training new models from the scratch [9]. For example, in zero-shot image classification, image features that have been already learned by CNNs such as ResNet from images of seen classes are often directly re-used to build classifiers for unseen classes. The key challenge is selecting the right knowledge to transfer and adaptively combining these transferred knowledge for a new prediction task. To this end, ZSL methods often utilize auxiliary information that describes inter-class relationships. When ZSL was originally investigated for visual object recognition and image classification, the methods mainly use attributes that describe objects' visual characteristics (a.k.a. class attributes) [2, 3]. Next, class textual information such as class name and sentence description is widely studied due to its high accessibility [10, 11]. In recent five years, Knowledge Graph (KG), which is a widely used representation of structured knowledge, such as those in RDF[1] triples in form of <Subject, Predicate, Object>, has attracted wide attention, and some KG-augmented ZSL methods have

Jiaoyan Chen (jiaoyan.chen@manchester.ac.uk) is from The University of Manchester and University of Oxford, UK. Yuan He and Ian Horrocks are from University of Oxford, UK. Yuxia Geng, Zhuo Chen, Wen Zhang and Huajun Chen are from Zhejiang University, China. Jeff Z. Pan is from The University of Edinburgh, UK.

* denotes corresponding authors.

[1]Resource Description Framework, https://www.w3.org/TR/rdf11-concepts/

even achieved the state-of-the-art performance on many tasks [12, 13, 14, 15].

FSL, which started to attract wide attention around when one-shot learning was proposed [5], has a longer history and even more studies than ZSL [16]. Since the unseen classes have some labeled samples although their sizes are quite small, techniques of *meta learning* (a.k.a. *learn to learn*) [17] have been widely applied [18]. Meta learning is usually applied by either reducing the parameter searching space in training using meta parameters such as more optimized initial parameter settings, or transforming a classification problem to a metric learning problem where a testing sample is matched with the unseen classes based on their few-shot samples and meta learned mappings. KGs have been utilized to optimise such meta learning-based methods; for example, Sui et al. [19] retrieve relevant knowledge from the NELL KG [20] to construct task-relevant relation networks as mapping functions for addressing few-shot text classification. Meanwhile, the aforementioned idea of knowledge transfer can also be adopted for addressing FSL, where KG as auxiliary information is becoming increasingly popular in recent years [21, 22, 23, 24]. For example, Chen et al. [22] transfer the feature learned by a CNN from flight delay forecasting tasks with a lot of historical records to a new forecasting task with limited historical records, by exploiting a KG with different kinds of flight related knowledge about e.g., airports and airlines; Peng et al. [23] extract a KG from WordNet for representing class hierarchies and then used this KG to augment knowledge transfer for few-shot image classification.

Motivation and Contribution. Since KG has become a very popular form for representing knowledge and graph structured data, acting as the foundation of many successful AI and information systems [25], it is thus reasonable to use KGs to augment both ZSL and FSL as discussed above. A few papers have been published on KG-aware zero-shot and few-shot learning especially in recent five years, and this research topic is becoming more and more popular. It is worth mentioning that this topic includes not only using KGs to augment ZSL and FSL but also addressing prediction tasks of the KG itself where ZSL and FSL methods are applied and extended for the KG context. By the middle of December in 2021, we have collected 50 papers on KG-aware ZSL and 46 papers on KG-aware FSL. To systematically categorize and compare all the proposed methods, and to present an overall picture of this promising field, a comprehensive survey is now in urgent need. In this paper, we *(i)* introduce KGs and their construction methods for ZSL and FSL, *(ii) categorize, analyze and compare* different kinds of KG-aware ZSL and FSL methods (see Figure 1 for an overview of the paradigms and categories), *(iii)* present ZSL and FSL tasks as well as their evaluation resources in various domains including computer vision (CV), NLP and KG completion, and *(iv)* discuss the existing challenges and potential future directions. This survey is suitable for all AI researchers, especially those who are to enter the domain of ML with sample shortage, those who have already been working on this topic but are interested in solutions utilizing knowledge representation and reasoning, and those who are working on KG and semantic techniques.

Related Literature Reviews. There have been several papers that have literature reviews relevant to zero-shot and few-shot learning, but they are all quite different from this survey.

- The two survey papers [7] and [16] systematically review the ZSL methods by 2019 and the FSL methods by 2020, respectively, mainly from the perspective of problem setting (e.g., whether the unlabeled testing samples are used or not in training), ML theory (e.g., which prediction error to reduce), and methodology (e.g., data focused, model focused and learning algorithm focused). However, they do not consider the categorization and deep analysis from the perspective of auxiliary information, and failed to collect most KG-aware methods.
- The very recently released paper [26] reviews both ZSL and FSL methods that use or aim at structured data. Structured data, however, is more general than KG with a much larger scope, and thus [26] collects only a small part of the papers on KG-aware ZSL and FSL research. It includes 19 papers about KG-aware ZSL and 21 papers about KG-aware FSL, while this survey has 50 papers and 46 papers, accordingly. This survey also has a more fine-grained method categorization, and additional technical analysis on KGs and their construction for zero-shot and few-shot learning. Meanwhile, [26] focuses more on addressing problems in structured data by ZSL and FSL methods, but less on augmenting ZSL and FSL methods.
- The paper [8] is our previous survey and perspective paper published in IJCAI 2021 Survey Track. It briefly categorizes different external knowledge used in ZSL with incomplete reviews on KG-aware ZSL papers, and it does not cover FSL.
- The benchmarking paper [27] was published in 2018. It reviews around 10 ZSL methods that mainly utilize class attribute and text information as the auxiliary information, focusing on their evaluation and result comparison on image classification task. This paper covers neither state-of-the-art ZSL methods proposed in recent 3 years nor KG-aware ZSL methods. Similarly, the survey paper [6] reviews ZSL papers published before 2018, mainly focusing on ZSL studies on CV tasks.

Paper Organization. The remainder of this survey is organized as follows. Section II introduces the preliminary, including the definitions and annotations of ZSL and FSL, and an overall view of the auxiliary information. Section III introduces the definition and scope of KGs, as well as the KG construction for ZSL and FSL. Section IV reviews KG-aware ZSL methods which are categorized into four paradigms: mapping-based, data augmentation, knowledge propagation and feature fusions. For each paradigm, we further introduce different categories and their corresponding methods. Section V is similar to Section IV but reviews KG-ware FSL methods, and compares KG-aware FSL and ZSL in the end. Section VI introduces the development and resources of KG-aware ZSL and FSL in different tasks across CV, NLP and KG completion. Section VII discusses the existing challenges and the future directions. Section VIII concludes this paper.

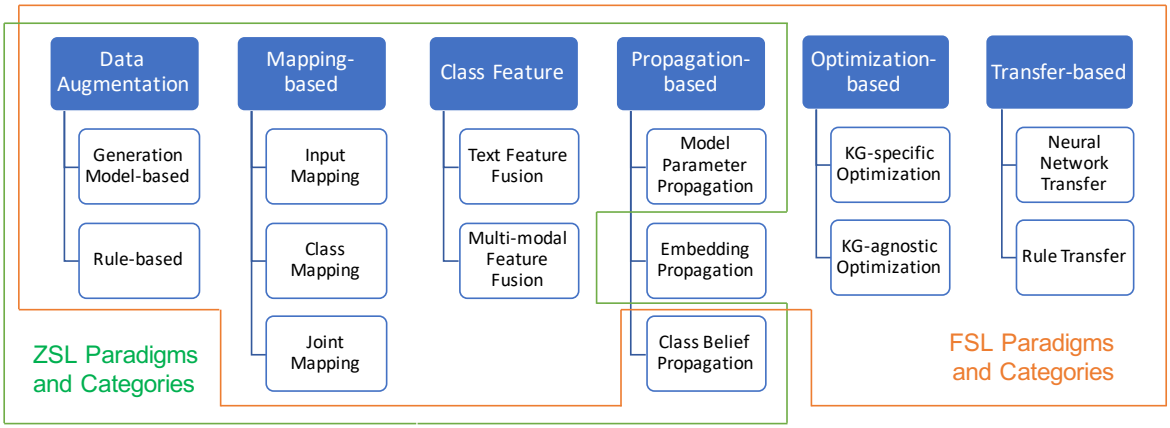


Fig. 1: Paradigms and categories of KG-aware zero-shot and few-shot learning methods

Annotation	Introduction
$\mathcal{D}_{tr}$, $\mathcal{X}_s$, $\mathcal{Y}_s$	Training sample set, inputs of the training samples, seen classes, respectively
$\mathcal{D}_{te}$, $\mathcal{X}_u$, $\mathcal{Y}_u$	Testing sample set, inputs of the testing samples, unseen classes, respectively
$\mathcal{D}_{few}$, $\mathcal{X}_{few}$	Few-shot sample set, inputs of the few-shot samples, respectively
$f : x \to y$	The target function mapping the input x to the output class y
$f' : (x, y) \to s$	The function that scores the matching degree between x and y
$g : x \to \boldsymbol{x}$	The encoding function of the input x
$h : y \to \boldsymbol{y}$	The encoding function of the class y

TABLE I: A list of the ZSL and FSL annotations in the paper.

II. Preliminary on Zero-shot and Few-shot Learning

Both ZSL and FSL have been applied in many different tasks, varying from image classification and visual question answering to text classification, knowledge extraction and KG completion. Although the exact ZSL and FSL problems may be different between papers, they can be expressed under one framework. In the session, we aim to present this framework with formal problem definitions and annotations, and at the same time introduce some background knowledge that are needed for understanding KG-aware ZSL and FSL. We start from ZSL, and then introduce FSL based on ZSL.

A. Zero-shot Learning

We first give a simple but generic definition towards ML classification, then formally define ZSL and introduce auxiliary information, and finally introduce the existing categorization of ZSL works.

Definition 1 (Supervised ML Classification): Given a set of labeled training samples $\mathcal{D}_{tr} = \{(x, y)|x \in \mathcal{X}, y \in \mathcal{Y}\}$, a classifier is trained to approximate a target function from the input x to the output label y, denoted as $f : x \to y$, such that f is able to correctly predict the labels of samples in a testing set $\mathcal{D}_{te} = \{(x, y)|x \in \mathcal{X}', y \in \mathcal{Y}\}$, where $\mathcal{X} \cap \mathcal{X}' = \emptyset$.

In image or text classification, x is an image or text while y is an image category or a text category. Sometimes, one input can be annotated by multiple labels, which is known as multi-label classification. In question answering, we refer to giving an answer or multiple answers to a natural language question w.r.t. a given textual context, where the label y corresponds to the answer. Visual question answering is similar but the context is an image or a video. In knowledge extraction, the task is usually to extract entities, relations or events from natural language text. It also includes entity or relation linking, which matches an entity or relation mention in text with a pre-defined entity or relation, and entity typing, which assigns a pre-defined class or multiple predefined classes to an entity mention in the text. Thus x is often a sentence or a document with an entity or relation mention, while y is a label corresponding to an entity, a relation, an event, or a class. In KG completion, which is to predict a missing RDF triple, x is often the two components of a triple while y is a label indicating the third component. Note in all these tasks, a candidate set is usually given for the output class y.

In supervised ML classification, given $\mathcal{D}_{tr}$, the trained classifier can only predict samples of classes that have appeared in the training stage (i.e., $\mathcal{Y}$), while ZSL aims to predict samples beyond $\mathcal{Y}$. Here is the formal definition.

Definition 2 (Zero-shot Learning): Given a training sample set $\mathcal{D}_{tr} = \{(x, y)|x \in \mathcal{X}_s, y \in \mathcal{Y}_s\}$ where $\mathcal{X}_s$ and $\mathcal{Y}_s$ are the training sample inputs and their classes, respectively, *Standard ZSL* aims to train f with $\mathcal{D}_{tr}$ for predicting on a testing set $\mathcal{D}_{te} = \{(x, y)|x \in \mathcal{X}_u, y \in \mathcal{Y}_u\}$ where $\mathcal{X}_u$ and $\mathcal{Y}_u$ are the testing sample inputs and their classes, respectively, with $\mathcal{Y}_u \cap \mathcal{Y}_s = \emptyset$. $\mathcal{Y}_s$ is called *seen classes* while $\mathcal{Y}_u$ is called *unseen classes*. When it is required to predict testing samples of both seen and unseen classes, i.e., $\mathcal{D}_{te} = \{(x, y)|x \in \mathcal{X}_u \cup \mathcal{X}'_s, y \in \mathcal{Y}_u \cup \mathcal{Y}_s\}$ with $\mathcal{X}_s \cap \mathcal{X}'_s = \emptyset$, the problem becomes *Generalized ZSL*.

It is worth mentioning that in addressing some ZSL tasks such as text classification and question answering, the original function f is sometimes transformed into a new scoring function by moving y to the input, denoted as $f' : (x, y) \to s$, where the output s is a score indicating whether y is the label of x or not. The label of a testing sample x in $\mathcal{X}_u$ (resp. $\mathcal{X}_u \cup \mathcal{X}'_s$) is predicted by finding out the class in $\mathcal{Y}_u$ (resp. $\mathcal{Y}_u \cup \mathcal{Y}_s$) that maximizes the score s.

Definition 3 (Auxiliary Information): Auxiliary information is some kind of symbolic data that describe or indicate the relationship between seen and unseen classes, such as class attribute, class text description and class hierarchy. With the auxiliary information, classes are usually encoded into sub-symbolic representations (i.e., vectors) with the relationship between classes concerned in the vector space. We denote the class encoding as the function $h : y \rightarrow \boldsymbol{y}$ where the bolded $\boldsymbol{y}$ represents the vector of the class y, $y \in \mathcal{Y}_u \cup \mathcal{Y}_s$.[2]

Since unseen classes have no labeled samples, ZSL methods rely on *auxiliary information*. In early years when ZSL was proposed in around 2009 for image classification, the majority of the solutions utilize class attributes which are often a set of key-value pairs for describing object visual characteristics [2, 28]. There are also relative attributes which enable comparing the degree of each attribute across classes (e.g., "bears are furrier than giraffes") [29], and real valued attributes for quantifying the degrees [28, 27]. The advantages and disadvantages of the attribute auxiliary information are quite obvious: it is easy to use and quite accurate with little noise, but it cannot express complex semantics for some tasks and is not easily accessible, usually requiring annotation by human beings or even domain experts. From around 2013, class textual information, varying from words and phrases such as class names to long text such as sentences and documents for describing classes, started to attract wide attention in ZSL. Typical works lie in not only image classification [10, 30] but also other tasks such as KG completion [31]. Text information is easy to access for common ZSL tasks. It can be extracted from not only the data of the ZSL tasks themselves but also encyclopedias, Web pages and other online resources. However, it is often noisy with irrelevant words and the words are often ambiguous, failing to accurately express fine-grained, quantified or more complex inter-class relationships.

In recent years, structured knowledge in the scope of KG, such as class hierarchies and commonsense knowledge, have become increasingly popular in ZSL research with very promising performance achieved. Such knowledge can often express richer semantics than attributes and text, even including logical relationships, and at the same time, they become more available with the development of KG construction techniques and the availability of many public KGs such as WordNet [32], ConceptNet [33] and Wikidata [34]. In this survey, we mainly review KG-aware ZSL studies, using Section III to independently introduce the involved KGs, and Section IV to review the involved methods.

The survey paper [7] has categorized ZSL methods into the following two categories:

- *Classifier-based*. The classifier-based methods are to directly learn a classifier for each unseen class. They could be further divided into *(i) Corresponding Methods* which exploit the correspondence between the binary one-vs-rest classifier for each class and its corresponding encoding of the auxiliary information, *(ii) Relationship Methods* which calculate and utilize the relationships among classes, and *(iii) Combination Methods* which combine classifiers for basic elements that are used to constitute the classes.
- *Instance-based*. The instance-based methods are to obtain labeled samples belonging to the unseen classes and use them for learning and prediction. They are further divided into *(i) Projection Methods* which learns a function to project the input and the class encoding into the same space (i.e., the class encodings after projection are regarded as labeled samples), *(ii) Instance-borrowing Methods* which transfer samples from seen classes to unseen classes, and *(iii) Synthesizing Methods* which obtain labeled samples for the unseen classes by synthesizing some pseudo samples.

This categorization is mainly from the perspective of ML theory and method. It aims at general ZSL methods, no matter what kind of auxiliary information is utilized. In contrast, our categorization which is to be introduced in Section IV is from the perspective of auxiliary information, and focuses on more fine-grained comparison and analysis towards those KG-aware ZSL methods. Meanwhile, since the survey [7] was published in 2019 while many KG-aware ZSL methods were proposed in recent two years, the KG-aware methods covered are not complete.

B. Few-shot Learning

We first formally define FSL, following the annotations in defining ZSL, then introduce the auxiliary information and finally present the current method categorization.

Definition 4 (Few-shot Learning): Given a set of training samples $\mathcal{D}_{tr} = \{(x, y)|x \in \mathcal{X}_s, y \in \mathcal{Y}_s\}$ and a set of few-shot samples $\mathcal{D}_{few} = \{(x, y)|x \in \mathcal{X}_{few}, y \in \mathcal{Y}_u\}$, where $\mathcal{Y}_u \cap \mathcal{Y}_s = \emptyset$, each class in $\mathcal{Y}_s$ has a large number of samples in $\mathcal{D}_{tr}$ and each class in $\mathcal{Y}_u$ has only a small number of samples in $\mathcal{D}_{few}$, FSL is to train a classifier f with $\mathcal{D}_{tr}$ and $\mathcal{D}_{few}$ for predicting samples in a testing set $\mathcal{D}_{te} = \{(x, y)|x \in \mathcal{X}_u, y \in \mathcal{Y}_u\}$ with $\mathcal{X}_u \cap \mathcal{X}_{few} = \emptyset$, or $\mathcal{D}_{te} = \{(x, y)|x \in \mathcal{X}_u \cup \mathcal{X}'_s, y \in \mathcal{Y}_u \cup \mathcal{Y}_s\}$ with $\mathcal{X}_s \cap \mathcal{X}'_s = \emptyset$.

To be consistent with ZSL, we keep calling the classes with a large number of training samples, i.e., $\mathcal{Y}_s$, as *seen classes*, those classes with few-shot labeled samples in $\mathcal{D}_{few}$, i.e., $\mathcal{Y}_u$, as *unseen classes*. As in ZSL, the original target of learning f can also be transformed into learning a scoring function for ranking the candidate classes, i.e., $f' : (x, y) \rightarrow s$.

The few-shot samples samples in $\mathcal{D}_{few}$ can be just one labeled sample per class, which is known as *one-shot learning*. It can also have multiple labeled samples, but the size is relative small and they alone are far from enough to train a robust model for an unseen class. To be more specific, we introduce the concept of *expected risk* as in [16]. For an optimal hypothesis $\hat{h}$ (i.e., the target function f), its expected risk is composed of two parts: *(i)* approximation error $\mathcal{E}_{app}$ which measures how close the best hypothesis h^* in a given hypothesis set $\mathcal{H}$ can approximate $\hat{h}$, and *(ii)* estimation error $\mathcal{E}_{est}$ which measures the effect of minimizing the empirical risk of the learned hypothesis $\bar{h}$ w.r.t. the best hypothesis h^*

[2]The raw input x could also be encoded by e.g., some pre-trained models or hand-craft rules before they are fed to f (or f'). This step is optional but is often adopted. We denote this initial encoding function as $g : x \rightarrow \boldsymbol{x}$ where the bolded $\boldsymbol{x}$ represents the encoding vector of x, $x \in \mathcal{X}_s \cup \mathcal{X}_u$.

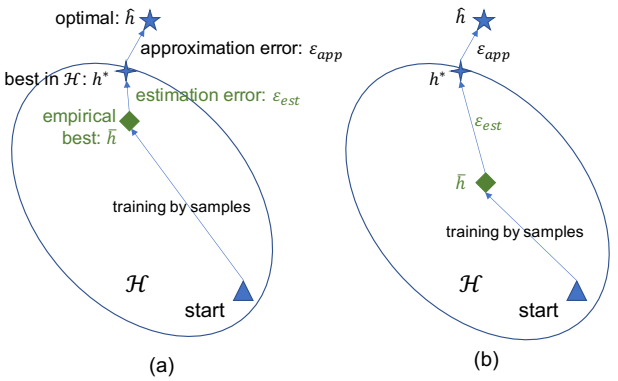


Fig. 2: Expected risk with (a) sufficient samples and (b) limited samples for training [16].

[35]. As shown in Figure 2, model training for unseen classes with $\mathcal{D}_{few}$ has a much higher estimation error than model training for seen classes with $\mathcal{D}_{tr}$.

The key difference of FSL in comparison with ZSL lies in the few-shot samples $\mathcal{D}_{few}$. Most FSL methods now focus on fully utilizing $\mathcal{D}_{few}$. They prefer some ML algorithms such as multi-task learning which allows parameter sharing between tasks, meta learning which directly predicts some parameters and hyper-parameters that are to learn or to adjust, and metric learning which compares a testing sample with the few-shot samples of each unseen class in some space after projection.

The common aspect of FSL and ZSL lie in the utilization of the auxiliary information. All the auxiliary information used in ZSL can also be used in FSL, and the utilization method can be transferred to FSL easily. Even the few-shot samples can be regarded as an additional kind of auxiliary information. KG has also been investigated in FSL as effective auxiliary information. In FSL, we also denote the encoding of the class with the auxiliary information as $h : y \rightarrow \boldsymbol{y}$.

The survey [16] divides FSL methods into the three general categories according to the aspects that are augmented:

- Data augmentation methods. They increase the size of the few-shot samples ($\mathcal{D}_{few}$) via data augmentation by e.g., transforming samples from the training set $\mathcal{D}_{tr}$ or other similar labeled data, and generating samples from weekly labeled or unlabeled data.
- Model augmentation methods. They reduce the original hypothesis set $\mathcal{H}$ to a small one for reducing the searching space. They are further divided into *(i)* multi-task learning methods which share parameters between tasks or to regularize the parameters of the target task, *(ii)* embedding methods which project samples to an embedding space where similar and dissimilar samples can be easily discriminated, *(iii)* generative modeling methods which restrict the model distribution, and so on.
- Algorithm augmentation methods. They guide and accelerate the searching of the parameters of the best hypothesis h^* by e.g., learning the optimizer and aggregating existing parameters.

Although this is a systematic categorization, it has a limited coverage on KG-aware FSL methods, and ignores the role of the auxiliary information especially KGs. In this survey, we categorize and compare KG-aware FSL methods from the perspective of how KG is exploited.

III. Knowledge Graph

In this section we will first introduce the definition and different forms of KG, and then present the existing KGs that have been adopted in ZSL and FSL studies, as well as KG construction methods for specific ZSL or FSL tasks. See Figure 3 for an overview.

A. Definition and Scope

Knowledge Graph (KG) has been widely used for representing graph structured knowledge, and has achieved great success in many applications such as search engine, recommendation system, clinic AI and personal assistant [36, 25]. In this session we first introduce a basic but widely recognized KG definition and some basic KG access operations, then introduce the ontology-equipped KGs from the semantic Web perspective, and finally introduce the scopes of KG in domains beyond the semantic Web.

Definition 5 (Knowledge Graph): A KG, denoted as $\mathcal{G} = \{E, P, L, T\}$, is composed of an entity set E, a property set P, a literal set L, and a statement set T in the form of RDF triple. Each RDF triple in T is denoted as (s, p, o), where s represents the subject which is an entity in E, p represents a predicate which is a property in P, and o represents the object which can be either an entity in E or a literal in L.

Some statements represent relational facts. In this case, o is also an entity, and p is a relation between two entities (a.k.a., object property). s and o are also known as the head entity and tail entity, respectively. A set of relational facts composes a multi-relational graph whose nodes correspond to entities and edges are labeled by relations. Some other statements represent literals as e.g., entity attributes. In this case, the predicate p uses a data property and o is a literal with some data type such as string, date, integer and decimal. The literals also include KG meta information such as entity's label, textual definition and comment, which are also represented via built-in or bespoke annotation properties.

The content of a KG can usually be efficiently accessed via two operations: *lookup* and *querying*. KG lookup (a.k.a. KG retrieval) is a service that returns the most relevant entities and/or properties in a KG that match the meaning of an input string (usually a phrase). For fast retrieval, some lexical index is usually built based on the labels and other name information of the entities and relations. KG querying is a service that returns the answers of an input query of the RDF query language SPARQL[3]. The input of such a query is actually a sub-graph pattern with variables, while the output could be not only the matched entities, properties and/or literals, but also the whole sub-graphs (i.e., triples). Some modern graph databases such as RDFox [37] can already support efficient SPARQL query.

In the semantic Web, a KG is often accompanied by an ontology as the schema, using languages from the semantic Web

[3]https://www.w3.org/TR/rdf-sparql-query/

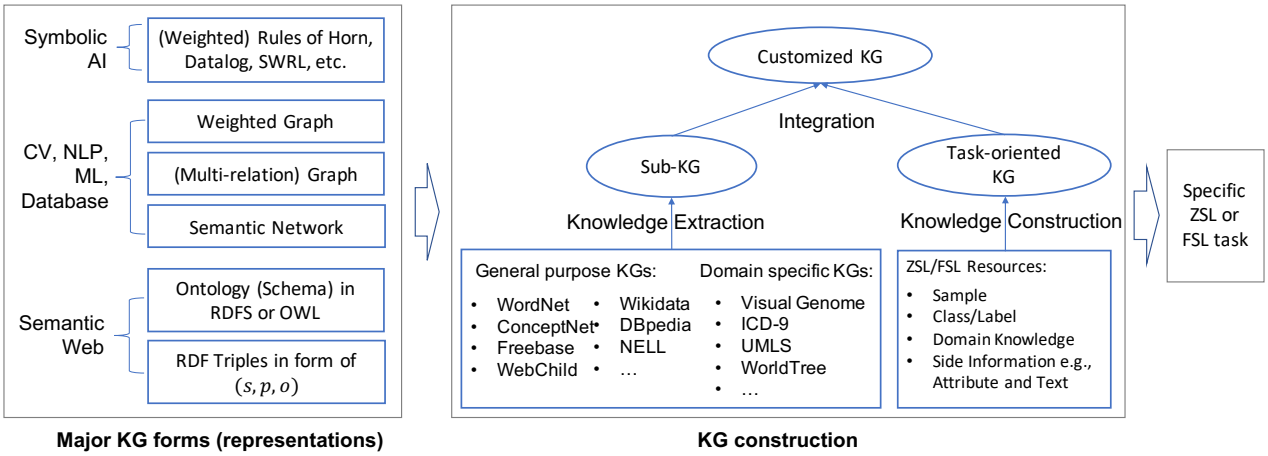


Fig. 3: An overall picture of the KGs in ZSL and FSL

community such as RDFS[4] and OWL[5] for richer semantics and higher quality [38]. They often define hierarchical classes (a.k.a. concepts)[6], properties (i.e., stating the terms used as relations), concept and relation hierarchies, constraints (e.g., relation domain and range, and class disjointness), and logical expressions such as relation composition. The languages such as RDF, RDFS and OWL have defined a number of built-in vocabularies for representing these knowledge, such as *rdf:type*, *rdfs:subClassOf*, *owl:disjointWith* and *owl:someValuesFrom*. Note RDFS also includes some built-in annotation properties such as *rdfs:label* and *rdfs:comment* for defining the above mentioned meta information. With these vocabularies, an ontology can be represented as RDF triples; for example, the subsumption between two concepts can be represented by the predicate *rdfs:subClassOf*, while the membership between an instance and a concept can be represented by the predicate *rdf:type*. The ontology alone, which is widely used to define domain knowledge, conceptualization and vocabularies such as terms and taxonomies, is also widely recognized as a KG. One typical example is SNOMED CT which systematically organizes medical terms as entities with names, definitions, existential restrictions, tree-like categorizations and so on [39]. It is worth mentioning that KGs, especially those OWL ontologies and those equipped with ontologies, can support symbolic reasoning, such as consistency checking which can find logical violations, and entailment reasoning which infers hidden knowledge according to Description Logics.

Besides the relational facts, literals and ontologies defined following the semantic Web standards, we also regard graph structured knowledge in some other forms as KGs, according to the terminologies and definitions used in other domains including ML, Database, CV and NLP. One popular KG form is Semantic Network which can be understood as a graph that connects different concepts (entities) often with labeled edges for representing different relationships. Two such KGs that are widely used in many domains are WordNet which is a lexical database with different relationships between words [32] and ConceptNet which stores commonsense knowledge and relationships between different terms [33]. We further relax the scope of KGs to single relation graphs such as simple taxonomy (i.e., a set of hierarchical classes) and graphs with weighted edges which may represent some quantitative relationships such as similarity and distance between entities.

We also regard logical rules of different forms, such as Horn clause, Datalog rules and SWRL[7] rules, as well as their soft or fuzzy extensions (i.e., weighted rules) [40], within the scope of KG. This is because many of these rules can be transformed into equivalent relational facts and ontological knowledge, and vice versa [41]. They can often be understood as logic models over KGs, through which hidden knowledge can be inferred.

B. General Purpose KGs

There have been several general purpose large-scale KGs that are open and can be directly utilized for different kinds of tasks. In this part we introduce these KGs and briefly present how their knowledge are extracted for ZSL and FSL.

- **WordNet** is a large lexical database with several different relationships between words, such as synonym, hyponym, hypernym and meronym [32]. Its 3.0 version contains $155,287$ words, organized in $117,659$ synsets for a total of $206,941$ word-sense pairs. WordNet can be directly accessed via online search and browse[8], and some python libraries such as NLTK. It is often used to build task-specific class hierarchies, especially for image classification, and has been the most widely used KG for augmenting both ZSL [13, 12, 42, 43, 44, 45, 46, 47, 14, 48, 49, 50] and FSL [51, 21, 23, 52, 53, 45].
- **ConceptNet** is a freely-available Semantic Network with commonsense knowledge[9] [33]. It stores a large number of entities which are either words or phrases. The latest version ConceptNet 5 has around 34 million facts (relationships between entities) of 34 relations including *Synonym*, *IsA*, *RelatedTo*, *HasContext*, *HasA*, etc., and well

[4]RDF Schema, https://www.w3.org/TR/rdf-schema/
[5]Web Ontology Language, https://www.w3.org/TR/owl2-overview/
[6]To distinguish *class* in ML and *class* in KG, we use *concept* for the latter.
[7]Semantic Web Rule Language, https://www.w3.org/Submission/SWRL/
[8]http://wordnetweb.princeton.edu/perl/webwn
[9]https://conceptnet.io/

supports 10 core languages. The *IsA* relation represents hyponyms and hypernyms, from which class hierarchies are often used for augmenting ZSL [54, 55, 56, 57, 58, 59, 60] and FSL [55, 61, 59] It is mostly applied in CV tasks but has also been exploited in open information extraction (e.g., Nguyen et al. [58]).

- **Freebase** is a large-scale KG with relational facts, contributed by multiple sources such as Wikipedia, MusicBrainz (a music database), Notable Names Database (an online database of biographical details of famous people) and volunteers [62]. Its official API has been shut down 2016, but it can still be accessed as a dump or via Google's Knowledge Graph API. The dump on Google[10] has around 1.9 billion triples with tens of millions of entities, while 63 million additional triples that have been deleted can also be downloaded. Freebase has been widely used for investigating KG techniques including KG augmented ZSL [63, 64, 48] and FSL [65, 63]. Different from WordNet and ConceptNet, Freebase is mainly applied in open information extraction.
- **Wikidata** is a collaboratively edited KG that is increasing at a high speed. By November 2022, it has over 100 million data items (entities). Wikidata can be directly downloaded as a dump, or accessed via its official online SPARQL query service[11] and APIs. It is increasingly used in different applications, but its usage for augmenting ZSL and FSL had not attracted any attention until when two studies [66, 65] were proposed for augmenting few-shot relation extraction and another two studies [14, 67] were proposed for augmenting ZSL.
- **DBpedia** is also a large-scale general purpose KG whose knowledge are mainly from Wikipedia, equipped with an ontological schema in OWL [68]. For the 2016-04 release, the English version has 6.0 million things (entities) and 9.5 billion RDF triples. DBpedia also has localized versions in 125 languages with much more entities. DBpedia can be directly downloaded as a dump, or accessed via its online SPARQL query service[12] and lookup service/API[13] which can efficiently return a ranked list of entities for an input phrase. It has also been used to augment ZSL, often acting as a complement of relational facts and literals such as entity descriptions [47, 14, 57]. DBpedia's schema (ontology) can also provide hierarchical concepts and other schema knowledge for e.g., augmenting zero-shot KG completion with unseen entities [48].
- **NELL** is a popular KG continuously extracted from the Web [20]. According to its official statistics accessed in November 2022, it has accumulated 2.8 million high confident beliefs (triples). NELL can be browsed online[14] or downloaded. We find two ZSL works and one FSL work that utilize NELL: Wang et al. [12] use NELL for zero-shot classification for images from NEIL — an image repository whose classes are aligned with NELL entities [69]; Geng et al. [14] use its RDFS schema (ontology) for augmenting zero-shot KG completion with unseen relations; Sui et al. [19] use its entity concepts for augmenting few-shot text classification.
- **WebChild** a large collection of commonsense knowledge extracted from the Web as NELL [70]. It contains triples that connect nouns with adjectives via fine-grained relations like hasShape, hasTaste, evokesEmotion, etc. Its 2.0 version has over 2 million disambiguated concepts and activities (entities), connected by over 18 million assertions (facts). WebChild has now been rarely used in ZSL and FSL. For augmenting (zero-shot) VQA, Chen et al. [57] and Wang et al. [71] use an auxiliary KG, whose facts are extracted from WebChild as well as ConceptNet and DBpedia.

[10]https://developers.google.com/freebase
[11]https://query.wikidata.org/
[12]https://dbpedia.org/sparql
[13]https://lookup.dbpedia.org
[14]http://rtw.ml.cmu.edu/rtw/kbbrowser

C. KG Construction for Zero-shot and Few-shot Learning

In this part, we will introduce how task-specific KGs are often constructed for ZSL or FSL. Before that, we will first very briefly introduce how task irrelevant general purpose and domain KGs are constructed. Most high quality domain-specific ontologies such as the medical ontology SNOMED [39] are often directly constructed by domain experts via collaboration, while many aforementioned general purpose KGs are constructed via crowdsourcing — they are either extracted from existing crowdsourced resources like Wikipedia or directly contributed by volunteers. To be more comprehensive, many KGs integrate different knowledge resources and databases; for example, ConceptNet [33], which was originally developed by crowdsourcing, further fused knowledge from DBpedia, Wiktionary, OpenCyc and so on. In fact, solutions and technologies of Linked Open Data, Ontology Network and Ontology Alignment can all be used for constructing KGs via integration. With the development of data mining, ML and other data analysis techniques, knowledge extraction from unstructured and semi-structured data such as the Web pages, tables and text have recently been widely investigated and used for KG construction; for example, NELL is continuously extracted from the Web [20], while Google's KG is extended with knowledge extracted from tables in Web pages [72].

For ZSL or FSL tasks, sometimes there are exactly suitable KGs that can be directly applied. For example, Huang et al. [73] directly use the event ontology named FrameNet [74] for supporting their zero-shot event extraction method. However, for the majority of the ZSL and FSL tasks, existing KGs usually cannot be directly applied due to their large sizes and irrelevant knowledge, and an (ad-hoc) KG should be extracted or constructed. In this part, we mainly review techniques of constructing KGs for augmenting ZSL and FSL. We divide these techniques into three categories: sub-KG extraction from existing KGs, KG construction with task-specific data, and knowledge integration.

1) Sub-KG Extraction: Given a ZSL (or FSL) task with $\mathcal{Y}_u$, $\mathcal{Y}_s$, $\mathcal{D}_{tr}$ (and $\mathcal{D}_{few}$), a straightforward solution is finding out a suitable KG and re-using it by extracting relevant knowledge. In the above part, we have already introduced those popular

and general purpose KGs, and the ZSL and FSL studies where each KG is applied. A ZSL or FSL method often extract a part of the KG by first matching the ML classes with KG entities, and then extracting the matched entities as well as some other related knowledge including neighbouring entities of the matched entities within k-hops, entities associated with the matched entities according to some specific relations, entities close to the matched entities in some embedding space, (hierarchical) concepts and other schema information of the entities, literals such as entity synonyms, descriptions and data properties, etc. Besides entities, some other KG elements such as relations and concepts can also be directly matched with related knowledge extracted.

For some ZSL and FSL benchmarks, classes have already been matched with KG entities; for example, in the work [12], a WordNet sub-graph with 30K nodes are extracted as a KG for an ImageNet subset that has 1K training classes, where all ImageNet classes are originally aligned with WordNet nodes. For most other benchmarks, the matchings are built by simple name comparison or some knowledge retrieval services, sometimes with even human intervention. For example, Kampffmeyer et al. [13] and Geng et al. [14] manually match all the 50 classes in an animal image classification benchmark named AwA2 with WordNet nodes; Nguyen et al. [58], who work on zero-shot entity extraction from text, first extract nouns and pronouns with a part-of-speech algorithm from all sentences in the dataset, and then search for their corresponding entities in ConceptNet and extract the matched entities and their adjacent ones.

Besides, some other domain specific KGs have also been exploited for augmenting ZSL and FSL with a part of their knowledge. Zhang et al. [65] extract concept-level relation knowledge from **UMLS** — an ontology of medical concepts [75], for few-shot relation extraction in the medical domain. Rios et al. [76] extract class hierarchies and class descriptions from **ICD-9** diagnosis and procedure labels for zero-shot and few-shot medical text classification. Luo et al. [77] extract a sub-KG for object relationships from **Visual Genome** — a knowledge base that stores connections between image visual concepts and language concepts [78], for augmenting zero-shot object recognition. Zhou et al. [79] train their zero-shot question answering model with facts extracted from **WorldTree** (V2.0) [80] — a knowledge base that contains explanations for multiple-choice science questions in the form of graph, covering both commonsense and scientific knowledge.

Sub-KG extraction enables us to re-use existing KGs, and allows fast and easy construction of a task-oriented KG often with limited knowledge engineering. However, it is often hard to find a suitable KG for some domain specific task, and the sub-KG extracted may have incomplete knowledge.

2) Task-oriented KG Construction: Instead of utilizing existing KGs, some ZSL and FSL studies build task-specific KGs. The classes' textual information such as class label is the most frequently utilized information for mining inter-class relationships and further for constructing the edges of a KG. Palatucci et al. [1] connect a word (which corresponds to a class in that task) to another according to their co-occurrence in a text corpus. Lee et al. [43] calculate WUP similarity of class labels, and used this similarity to build KG edges for representing positive and negative inter-class relationships. Wei et al. [44], Ghosh et al. [81] and Wang et al. [49] all consider calculating and adding edges to entities that are close to each other according to their labels' word embeddings. Class attributes have also been exploited for mining inter-class relationships. Zhang et al. [55] build a KG for the CUB benchmark which includes images of birds of fine-grained classes, by computing Hadamard product over the part-level class attributes. Hu et al. [82] and Chen et al. [50] both directly utilize the co-occurrence of class attributes to build edges between KG entities. Specially, Changpinyo et al. [83] consider both class attributes and word embeddings to calculate weighted edges between entities. Different from the above methods that use some auxiliary information for building KG edges, Zhao et al. [84] and Geng et al. [47, 14] model the class attributes as additional KG entities and connect them to the entities corresponding to the classes; while Li et al. [85, 86] generate new superclasses of the seen and unseen classes by clustering of the class names, so as to constructing class hierarchies for augmenting both ZSL and FSL.

Domain knowledge, which is often in the form of heuristics and logic rules, has also be used to construct task-specific KGs. Banerjee et al. [87] use heuristics to create a synthetic KG with science facts from the QASC text corpus and commonsense facts from the Open Mind Commonsense knowledge (text) corpus, for addressing both zero-shot and few-shot question answering. Chen et al. [88] add existential restrictions (a kind of description logic that quantifies a class by associated properties) to some classes of an animal taxonomy extracted from WordNet, so as to build an OWL ontology for the animal image classification benchmark AwA2.

There are also some ZSL and FSL studies that extract structured knowledge from the task data (samples) for constructing KGs which are further fed back to learning for augmentation. When Ghosh et al. [81] construct a KG for evaluating methods for few-shot action classification where some videos (samples) are given for each unseen class, they first extracted sample features for each class, then took the mean of these features as a KG entity, and finally calculated the cosine similarity between feature means for edges between entities. Bosselut et al. [89] generate a temporary KG on demand for each prediction request of zeros-shot question answering, using its text context and a Transformer-based neural knowledge model named COMET [90]. Chen et al. [50] add a co-occurrence relation between two classes (food ingredients) by calculating their co-occurrence frequency in the training samples, besides the common class attributes and class hierarchies.

Task-oriented KG construction requires more knowledge engineering work than sub-KG extraction, such as tabular data to KG transformation, but the constructed KG is more task specific often with fine-grained domain knowledge. It will also suffer from knowledge incompleteness, especially when the task does not have much data available or it is very hard to extract semantics from its raw, unstructured data.

3) Knowledge Integration: Although some general purpose KGs contain a large quantity of knowledge and are being continuously extended, it is still common that the knowledge

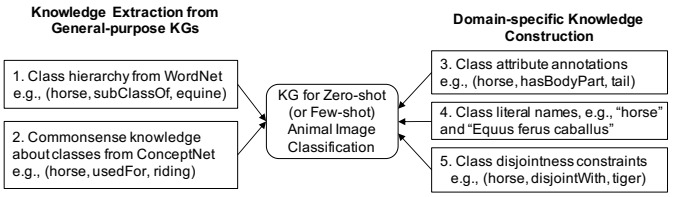


Fig. 4: An example KG construction.

extracted from such a KG is incomplete or not fine-grained enough for a specific ZSL or FSL task. Therefore, some studies proposed to integrate knowledge extracted from different KGs or/and other resources for building a high quality task-specific KG. For example, Chen et al. [57] extract RDF facts from three KGs — ConceptNet, WebChild and DBpedia to generate a unified commonsense KG for augmenting zero-shot VQA; Geng et al. [47, 14] integrate class hierarchies from WordNet, relational facts and literals from DBpedia, and knowledge transformed from class attributes for constructing KGs for zero-shot image classification; Chen et al. [50] consider and integrate class hierarchies from WordNet, and class co-occurrence relations extracted from class attributes and samples for a KG for zero-shot ingredient recognition from food images. Very recently, Geng et al. [91] conduct a benchmarking study, where six KG equipped ZSL benchmarks are created for three different tasks and used for evaluating different methods under different auxiliary information settings. The KG of each benchmark is based on the integration of multiple knowledge resources: those for zero-shot image classification integrate knowledge from WordNet, ConceptNet, class attributes, class names and so on, while those for zero-shot KG completion and relation extract integrate relation textual information, schema information from Wikidata or NELL, logic rules by human beings and so on.

As matching classes to KG entities for sub-KG extraction, the alignment of entities and relations in integrating different knowledge parts now is still mostly based on simple name matching or manual matching. There is little attention to investigating automatic knowledge integration methods for ZSL or FSL, and the impact of the knowledge quality, such as the matching accuracy and the ratio of relevant or redundant knowledge, is often ignored.

Knowledge integration alone often cannot construct a KG for a task. It acts a complement of sub-KG extraction and task-oriented KG construction for higher knowledge completeness. Although knowledge integration requires some knowledge engineering, some existing and open systems such as BERTMap [92] have already been developed.

4) A Case of KG Construction: Figure 4 shows an example of constructing a KG with different semantics for augmenting a zero-shot (or few-shot) image classification task, which is from our previous benchmarking work [91]. First, the hierarchical relationship between classes and commonsense knowledge about classes are extracted from WordNet and ConceptNet, respectively, where classes are matched with KG entities. For example, for an animal class *horse*, its ancestors such as *equine* are obtained from WordNet, and relational facts such as (*horse, usedFor, riding*) are extracted from ConceptNet. Next, some domain-specific knowledge such as attribute annotations of classes are represented as triples, where classes and attributes are represented as entities and connected via ad-hoc properties. For example, the attribute of *tail* is related to *horse* via property *hasBodyPart*. The literal names of classes are also represented using data properties. More complex semantics such as the disjointness between classes are also represented using OWL. For example, although *horse* and *tiger* have many shared attributes such as *tail* and *muscle*, they are still categorized as different species, and a disjointness constraint between them is added. Finally, all kinds of class knowledge mentioned above are integrated into one KG to serve the ZSL (or FSL) task.

IV. KG-aware Zero-shot Learning

According to the solutions for exploiting KGs, we divide the KG-aware ZSL methods into four paradigms: *Mapping-based*, *Data Augmentation*, *Propagation-based* and *Class Feature*, as shown in Figure 1. Table II summarizes the paradigms and lists the papers of each category. We will next introduce the details of each paradigm.

A. Mapping-based Paradigm

The mapping-based paradigm aims to build mapping functions towards the input (x or $\boldsymbol{x}$) and/or the classes (y or $\boldsymbol{y}$), so that their vector representations after mapping are in the same space and are comparable (i.e., an input is of a class if their vectors are close w.r.t. some metric like Cosine similarity and Euclidean distance). We denote the mapping function for the input as $\mathcal{M}$ and the mapping function for the class as $\mathcal{M}'$.

$\mathcal{M}$ and $\mathcal{M}'$ can be both linear and non-linear transformation networks, often learned from the training data $\mathcal{D}_{tr}$. Note that $\mathcal{M}$ and $\mathcal{M}'$ are different from the initial encoding functions g and h. g and h are just to represent the symbolic input and class as vectors or to learn features and semantic embeddings, while $\mathcal{M}$ and $\mathcal{M}'$ mainly aim to map the input and the class into the same space, although sometimes they may also play the role of g and h at the same time. According to the target(s) to map, we divide the ZSL methods of the mapping-based paradigm into three finer-grained categories: *Input Mapping*, *Class Mapping* and *Joint Mapping*. Figure 5 shows their insights. We will next introduce methods of each category, mainly from four dimensions — input encoding, class encoding, mapping function(s) and comparison metric.

1) Input Mapping: As shown in Figure 5 (a), methods in this category only learn $\mathcal{M}$ to map the input into the space of the class initial encoding. A simple but typical method is proposed by Palatucci et al. [1] for neural activity classification. In this task, the class is annotated by multiple attributes which are either calculated from classes' word similarity or manually created via crowdsourcing. Each class is encoded by a multi-hot vector of its attributes[15]. Given an input (neural activity signals) x, the mapping function $\mathcal{M}$, which is a multiple output

[15]A multi-hot vector is to represent a set of discrete variables. Briefly, one slot corresponds to one variable; a slot is set to 1 if its corresponding variable exists and to 0 otherwise.

Paradigms	Summary	Categories	Papers
Mapping-based	These methods project the input and/or the class into a common vector space where a sample is close to its class w.r.t. some distance metric, and prediction can be implemented by searching the nearest class.	Input Mapping	[1, 46, 63, 42, 64, 88, 85, 67]
		Class Mapping	[93, 45, 83, 94, 54]
		Joint Mapping	[63, 73, 95, 56, 57, 76, 60, 15]
Data Augmentation	These methods generate samples or sample features for the unseen classes, utilizing KG auxiliary information.	Rule-based	[96]
		Generation Model-based	[55, 31, 14]
Propagation-based	These methods propagate model parameters or a sample's class beliefs from the seen classes to the unseen classes via a KG.	Model Parameter Propagation	[12, 13, 44, 47, 50, 81, 49]
		Class Belief Propagation	[43, 77, 89]
Class Feature	These methods encode the input and the class into features often with their KG contexts considered, fuse these features and feed them directly into a prediction model.	Text Feature Fusion	[97, 98, 99, 100, 87, 79, 101, 102, 48, 103, 104, 105, 106]
		Multi-modal Feature Fusion	[48, 58, 59, 107]

TABLE II: A summary of KG-aware ZSL paradigms.

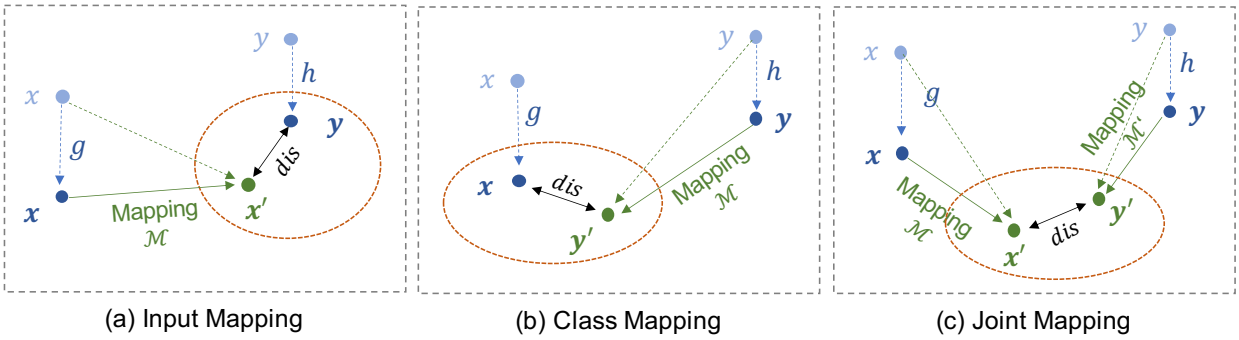


Fig. 5: Method categories and insights of the mapping-based paradigm. The dotted red circle denotes the vector space that the input and/or the class are mapped to.

linear regression model, predicts a multi-hot attribute encoding (vector) $\boldsymbol{y}'$. $\mathcal{M}$ is further attached by a 1-nearest neighbour classifier $\mathcal{L}$ which outputs the class as the one whose encoding is closest to $\boldsymbol{y}'$. The whole model ($\mathcal{M}$ and $\mathcal{L}$) is jointly learned by minimizing some vector error-based loss on $\mathcal{D}_{tr}$.

Input mapping is widely used in zero-shot image classification, often with more complicated class encoding and mapping function than the method in [1]. Li et al. [46] map image features to the space of the class encoding, where is based on the word embeddings of the class itself and its super classes, and use the Cosine similarity for comparison. Chen et al. [88] adopt a typical ZSL method named Semantic Autoencoder (SAE) [108] which uses a linear encoder as $\mathcal{M}$, and use pre-trained ontology embedding for the initial class encoding $\boldsymbol{y}$. $\mathcal{M}$ is learned on $\mathcal{D}_{tr}$ by minimizing a distance loss between $\boldsymbol{x}'$ and $\boldsymbol{y}$ and a reconstruction loss when $\boldsymbol{x}'$ is mapped back to $\boldsymbol{x}$. Li et al. [85] propose to use a Long-Short-Term-Memory (LSTM) network to model the class hierarchy for class encoding, and map the image features learned by a CNN. Liu et al. [42] use reinforcement learning and an ontology to associate rules with each class as visual characteristics, and train one Support Vector Machine (SVM) for each rule as the mapping function $\mathcal{M}$. $\mathcal{M}$ predicts associated rules of each input, and these predicted rules are compared with each candidate class.

Input mapping has also been explored in ZSL for knowledge extraction. Ma et al. [63] work on entity mention typing. They pre-train the initial class (entity type) encoding using different KG embedding methods such as prototype-driven label embedding and hierarchical label embedding, and then proposed two mapping settings. One setting is to directly map the input (entity mention features) to the class encoding by a linear transformation which is implemented by multiplying the input by a matrix of weights. Imrattanatrai et al. [64] learn initial text representation of the input (relation mention) by word embedding and a Bidirectional LSTM network, and then use a linear transformation function to map the text representation into the space of relation encoding which is based on KG TransE embedding and ad-hoc relation feature extraction. Li et al. [67] map the input text representation into the class embedding by a simple linear transformation for zero-shot relation classification, where the class encoding combines word embedding, normal KG embedding and rule-guided KG embedding.

2) Class Mapping: The methods learn the mapping function $\mathcal{M}'$ to directly map the class into the space of the initial input encoding, as shown in Figure 5 (b). It is not as widely investigated as input mapping. In total, we gather three

methods for zero-shot image classification [93, 45, 83] and one method for zero-shot KG completion with unseen entities [54]. Akata et al. [93, 45] learn an embedding model as the mapping function which maps the class initial encoding into the space of the image features. They use class hierarchies as the auxiliary information for initial class encoding, where each class is represented as a multi-hot vector of its ancestors. Changpinyo et al. [83] first generate a weighted graph where the relatedness between classes are represented, then introduce phantom classes through which seen and unseen classes can be synthesized by convex combination, and finally map the vectors of phantom classes into the input. Nayak et al. [54] propose a novel transformer Graph Convolutional Network (GCN) as $\mathcal{M}'$ which non-linearly aggregates a class's neighbours in the KG, and use a compatibility score as the metric for comparing the image CNN feature (input) and the class embedding. Shah et al. [94] predict KG triples with unseen entities using their text descriptions. The method first encodes the entity from the graph perspective by TransE or DistMult KG embedding (as initial class encoding h), and encodes the entity from the text perspective by word embedding and LSTM (as initial input encoding g), and then maps the class encoding to the space of the input encoding, where both linear and non-linear transformation functions such as Multi-Layer Perceptron (MLP) are explored.

3) Joint Mapping: As shown in Figure 5 (c), joint mapping learns both input mapping $\mathcal{M}$ and class mapping $\mathcal{M}'$ such that the input and the class are compared in one intermediate space. It is often adopted for zero-shot entity/relation extraction where features of both the input (entity mention text) and the class (entity/relation in a KG) are jointly mapped. Ma et al. [63] multiply the initial input encoding and the initial class encoding by matrices (as $\mathcal{M}$ and $\mathcal{M}'$) which are learned by minimizing a weighted approximate-rank pairwise loss, for zero-shot entity extraction. Huang et al. [73] map the features of event mentions and their structural contexts parsed from the text, and the event type encoding which is based on event ontology embedding, jointly into one vector space using one shared CNN.

Zero-shot text classification is very similar to zero-shot entity/relation extraction — $\mathcal{M}$ and $\mathcal{M}'$ are applied to text input encoding and KG-based class encoding, respectively. Rios et al. [76] map the text features learned by a CNN, and the class encoding which is by initial word embedding and GCN-based class hierarchy embedding. Chen et al. [60] linearly map the text encoding by BERT and the class encoding which is based on a word embedding model tailored by ConceptNet.

In zero-shot KG completion, the KG embedding (input) and the initial encoding of the unseen entity or relation (class) are jointly mapped. Hao et al. [95] investigate zero-shot KG completion with unseen entities. The input mapping $\mathcal{M}$ is a linear encoder over one-hot encoding of the KG entities, while the class mapping $\mathcal{M}'$ is a MLP over the encoding of the entity's attributes.

There are also some joint mapping methods in CV. Roy et al. [56] work on zero-shot image classification. They map both the initial class encoding learned by a GCN on commonsense knowledge and the image features learned by a CNN named

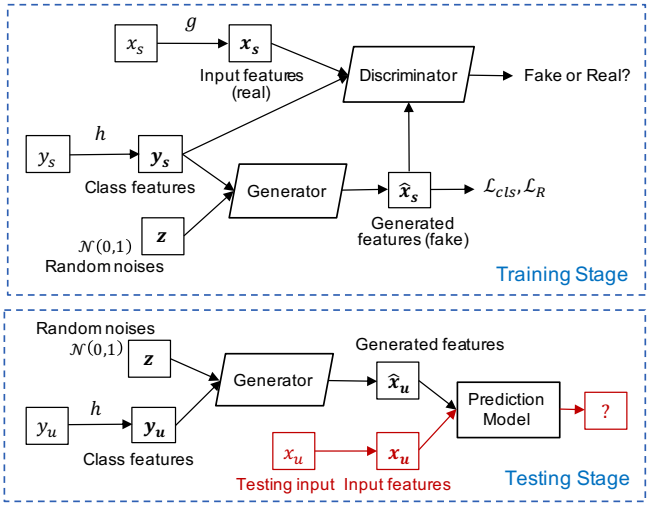


Fig. 6: Overview of the GAN-based generation paradigm.

ResNet101, using a non-linear transformation named Relation Network. This network first attaches a fully connected layer to the class encoding, then concatenates its output with the image features, and finally attaches two different fully connected layers. It is learned by minimizing a MSE loss. Chen et al. [15] propose an end-to-end transformer-based ZSL method named DUET, utilizing the pre-trained language models to enhance the discrimination capability on fine-grained visual characteristics where the structured KGs are serialized as the input. Chen et al. [57] work on zero-shot VQA. They map the input (i.e., initial encoding of a pair of image and question) and the encoding of a KG entity (class) to a common space, where the matched KG entity is regarded as the answer.

B. Data Augmentation Paradigm

A straightforward solution for addressing sample shortage is generating new data with the guidance of some auxiliary knowledge. In ZSL, some methods generate samples or sample features for unseen classes and transform the problem into a standard supervised learning problem. We regard these methods as Data Augmentation Paradigm. According to the generation method, we divide these methods into two categories: *Rule-based* and *Generation Model-based*.

1) Rule-based: Background knowledge of a task could be explicitly represented by different kinds of rules (or some equivalent logic forms such as schema constraints and templates). They enable deductive reasoning for hidden knowledge as new samples. However, this solution has been rarely investigated for ZSL. In image classification and some other tasks where the sample input and their features are uninterpretable real value vectors, generating data by rules becomes unfeasible. The only work we know is for zero-shot KG completion. Rocktaschel et al. [96] predict an unseen relation for two entity mentions extracted from text. They propose three methods to inject first-order rules, which act as commonsense knowledge, into a matrix factorization-based KG completion model. One of the methods is logically inferring additional relational facts in advance before training the matrix factorization model.

2) Generation Model-based: With the wide investigation of conditional generation, generation models such as Generative Adversarial Networks (GANs) [109] and Variational Auto-encoder (VAE) [110] have become popular tools for generating data for addressing ZSL especially for image classification [14, 111, 112, 113, 8, 7, 55]. We regard these methods as Generation Model-based. In Figure 6, we present a typical GAN-based scheme, including a training stage and a testing stage. During training, given a seen class y_s, its encoding $\boldsymbol{y_s}$, which can be based on multi-hot attribute encoding, word embedding and KG embedding, is fed into the generator of the GAN, together with a random noise vector $\boldsymbol{z}$ sampled from Normal distribution $\mathcal{N}(0, 1)$, to generate a set of sample features $\hat{\boldsymbol{x}}_s$ for y_s. Note the number of generated samples is a hyperparameter that can be tuned. Correspondingly, a set of real features (encoding) $\boldsymbol{x}_s$ are extracted from the input samples x_s of y_s to supervise the training of the generator via an adversarial discriminator which distinguish $\boldsymbol{x}_s$ and $\hat{\boldsymbol{x}}_s$. Both the generator and the discriminator are conditioned on the class embeddings. Neural networks composed of several fully connected layers are often used as their model. Some additional loss terms such as classification ($\mathcal{L}_{cls}$) and regularization ($\mathcal{L}_R$) are usually also applied to encourage the model to generate more plausible samples. During testing, the trained GAN can synthesize samples for an arbitrary unseen class y_u with random noises and its encoding $\boldsymbol{y_u}$. With the synthesized data $\hat{\boldsymbol{x}}_u$, we can learn classifiers for the unseen classes and use them to predict testing samples, as normal supervised learning. We can also directly compare the features of each input testing sample with the synthesized sample features of each unseen class to determine the output class label.

Conditional generation models were not widely applied in ZSL until around 2018. We find three KG-aware methods. Geng et al. [14] is the first work that propose to generate the image features using a KG which models the semantic relationships between seen classes and unseen classes for zero-shot image classification. The class encoding based on this KG's text-aware embeddings leads to higher-quality image features than previous class encodings based on attributes and word embeddings. GAN is used as the generation model. The follow-up work by Geng et al. [114] further considers a new disentangled KG embedding method for encoding class semantics from multiple aspects, leading to better zero-shot image classification performance. Qin et al. [31] work on a zero-shot KG completion problem, where the testing triples involve new relations that have never appeared in training. They propose to first use GAN to generate sample features (i.e., KG embeddings) of the unseen relations conditioned on their textual description embeddings (i.e., class encodings), and then calculate scores of the testing triples by comparing the generated relation embeddings with the existing embeddings of the entities. Note the two works raised by Geng et al. [14, 114] also deal with the unseen relations in zero-shot KG completion besides the zero-shot image classification. They propose to synthesize the relation embeddings conditioned on the embeddings of an ontological schema which represents the semantics of the KG relations, such as relation hierarchy, and relations' domain and range constraints. Zhang et al. [55] work on zero-shot image classification by generating few-shot samples. They use a generation module to generates an instance-level graph, where dummy features (instances) are synthesized for those unseen classes by GAN. They finally address the problem over the instance-level graph by a propagation module and a meta learning strategy. Across the literature, we find that there is no current work that combines the VAE-based generation models with KG, leaving a great research space to explore.

C. Propagation-based Paradigm

Graph-based information propagation is a straightforward solution to utilize KG auxiliary information for ZSL. We regard these KG-aware ZSL methods as Propagation-based Paradigm. These methods first align seen and unseen classes with KG entities and build a graph with node features from the auxiliary KG. Then they use a graph propagation model to either approximate model parameters or predict class beliefs (scores) of the unseen classes, where this propagation model is usually trained from seen classes whose outputs are given based on models built from $\mathcal{D}_{tr}$. Accordingly, we divide the methods into *Model Parameter Propagation* and *Class Belief Propagation*. We will next introduce the general idea of each category, and its contributes mainly from two dimensions — the propagation graph and the propagation model.

1) Model Parameter Propagation: For each seen class y_s in $\mathcal{Y}_s$, a one-vs-rest binary classifier is trained with $\mathcal{D}_{tr}$. Such a classifier is usually composed of a pre-trained input encoding g (e.g., a CNN for image feature learning) and linear or non-linear classification layer(s). Parameters of the classification layer(s) are denoted as $p(y_s)$. Considering a simple but general case in Figure 7 (a), the seen classes y_s^1, y_s^2 and y_s^3 are aligned with three graph entities e_1, e_2 and e_3, respectively. The parameters $p(y_s^1)$, $p(y_s^2)$ and $p(y_s^3)$ are assigned to e_1, e_2 and e_3 as their outputs, and the output parameters of e_0 which is aligned with an unseen class y_u^1 ($y_u^1 \in \mathcal{Y}_u$), i.e., $p(y_u^1)$, are approximated according to a graph propagation model. With $p(y_u^1)$ and the pre-trained input encoding g, samples of y_u^1 can be predicted. The graph propagation model is usually trained by minimizing errors when approximating parameters of classifiers of $\mathcal{Y}_s$.

In image classification, usually one classifier, which is composed of a linear layer and pre-trained CNN image features, is trained for each seen class via $\mathcal{D}_{tr}$, and the linear layer parameters of the seen classes are propagated. Wang et al. [12] adopt a CNN named ResNet-50 for image feature learning. The method aligns image classes with WordNet [32] entities, and uses a GCN to propagate feature combination weights on a sub-graph of WordNet. Wei et al. [44] aim at the same problem as [12], but use a Residual Graph Convolutional Network (ResGCN), which builds residual connections between hidden layers, for propagation so as to alleviate over-smoothing and over-fitting when stacking multiple GCN layers. Ghosh et al. [81] uses a 6-layer GCN for propagation on a KG for zero-shot action recognition (video classification). Wang et al. [49] construct two single-relation KGs — one by the class hierarchy from WordNet and the other by the class

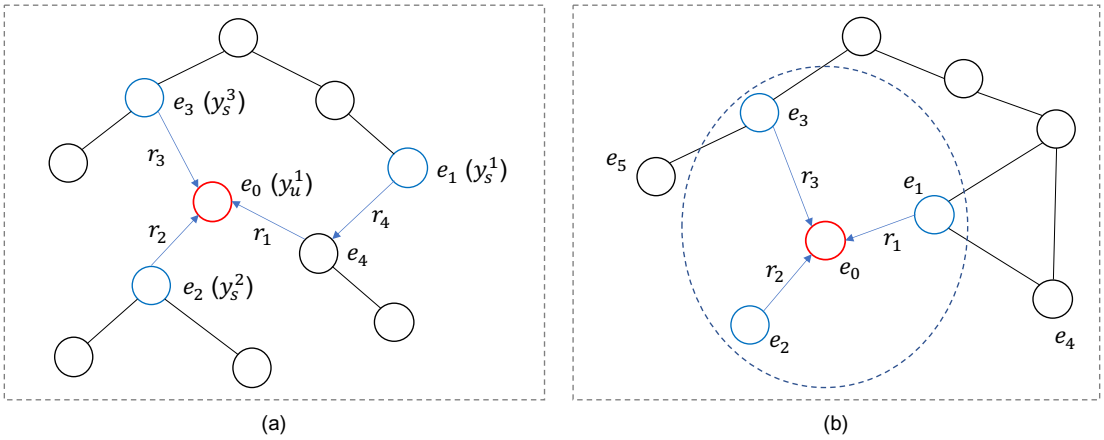


Fig. 7: (a) Graph information propagation from entities of seen classes (e.g., e_1, e_2 and e_3) to entities of unseen classes (e.g., e_0) for approximating model parameters (or predicting class beliefs). (b) Aggregating embeddings of of 1-hop neighbouring seen entities (in blue) to get the embedding of an unseen entity (in red) for few-shot KG completion.

correlation mined from word embeddings for zero-shot image classification. They use two weight-shared GCNs to propagate on the two KGs to predict classifier parameters for the unseen classes. When propagating on a sub-graph of WordNet for zero-shot image classificaton, Geng et al. [47] attach an attention layer after GCN layers to calculate the importance weights of different seen classes to an unseen class, which also provides a way for explanation on feature transferability. Chen et al. [50] develop a propagation-based method for estimating the parameters of multi-label classifiers for zero-shot ingredient recognition from food images. Since the KG, which is composed of knowledge of ingredient hierarchy, ingredient attributes and ingredient co-occurrence, has multiple different relations, an attentive multi-relational GCN is adopted, where different relations have different contributions in parameter propagation.

When propagating knowledge to distant nodes, all the above methods prefer to stack mutiple GCN layers. In contrast, Kampffmeyer et al. [13] propose to only use two GCN layers and extend to directly connect an entity to its ancestors and descendants, where an attention mechanism is used to weight the contributions of different neighbouring entities according to their distances to the target entity. Under the same task and evaluation setting as in [12], the two-layer network and these weighted additional connections often achieve better performance.

2) Class Belief Propagation: These methods are often for multi-label classification where one sample is annotated by multiple related classes. Without losing the generality, we assume a sample should be annotated by three seen classes (y_s^1, y_s^2 and y_s^3) and one unseen class (y_u^1), and these classes are matched to graph nodes, as shown in Figure 7 (a). The beliefs (scores) of y_s^1, y_s^2 and y_s^3 are predicted by their corresponding binary classifiers trained with $\mathcal{D}_{tr}$, while the score of y_u^1 is predicted by a graph propagation model trained with outputs of nodes of the seen classes.

One typical work is the zero-shot multi-label image classification method by Lee et al. [43], where multiple objects are to be recognized from an input image. It uses a gated recurrent update mechanism for iterative belief propagation on the auxiliary KG, where the propagation is directional from seen classes to unseen classes, and a standard fully-connected neural network to output a final belief for the entity of each unseen class. Note that the initial status of an entity is determined by the features of the corresponding class's samples and word embedding. Luo et al. [77] work on recognizing multiple interactive objects in an image where some objects are unseen in training. They use Conditional Random Field to infer the unseen objects using the recognized seen objects in the image and a KG with prior knowledge about the relationships between objects. Bosselut et al. [89] focus on zero-shot question answering. They propose to construct a context-relevant commonsense KG from deep pre-trained language models, where the question acts as a root entity and the answers act as leaf entities, and then they infer over the graph by aggregating paths to find the right answer. Although this method predicts only one answer (class) for each question (sample), but associates one question with multiple candidate answers for graph information propagation.

D. Class Feature Paradigm

As shown in Figure 8, class feature paradigm uses a transformed scoring function $f' : (g(x), h(y)) \rightarrow s$ to calculate a matching score s between an input x and a class y. The KG auxiliary information is usually utilized by the class encoding h. f' is usually composed of a fusion model, which combines $g(x)$ and $h(y)$, and a prediction model. f' can be trained with $\mathcal{D}_{tr}$. h and g can be separately pre-trained or trained jointly with f'.

This paradigm actually transforms the ZSL problem into a classic *domain adaption* problem: considering the new input of f', the training data ($y \in \mathcal{Y}_s$) have a different distribution as the testing data ($y \in \mathcal{Y}_u$ or $y \in \mathcal{Y}_u \cup \mathcal{Y}_s$). According to the types of $g(x)$ and $h(y)$, we further classify these KG-aware ZSL methods into two categories: *Text Feature Fusion* and *Multi-modal Feature Fusion*. Next we will introduce the

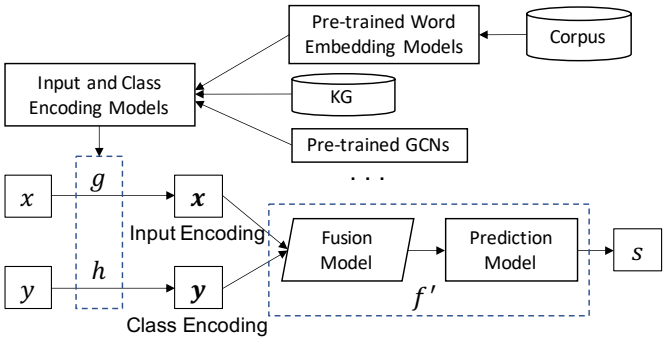


Fig. 8: Overview of the class feature paradigm.

works of each category mainly from three dimensions — the class encoding, the input encoding, and the fusion model.

1) Text Feature Fusion: These methods usually aim at ZSL tasks within a KG context where the auxiliary information is some kind of text. One typical setting is KG completion with unseen entities where entities are described by name phrases and/or textual descriptions. In this case, the class, i.e., an entity (or a relation) in a triple to predict, and the input, i.e., the remaining two elements of the triple, are both represented as in a textual form and encoded by text embedding.

Zhao et al. [97] adopt the TF-IDF algorithm to combine the embeddings of words to encode each entity with its text description. For a triple, the scoring function f' uses the encodings of its two entities to calculate a triple score, where the interactions between any two elements of the triple are modeled. Shi et al. [98] propose a zero-shot KG completion method named ConMask for dealing with unseen entities using their names and text descriptions. They feed the text encodings of the entities and the relation of a triple into a CNN-based fusion model. Niu et al. [101] follow the general direction of [97] and [98], but work out a new multiple attention-based method with a Bidirectional LSTM network and an attention layer for modeling and utilizing the interaction between the head entity description, head entity name, the relation name, and the tail entity description. Amador et al. [48] work on triple classification with unseen entities. The ontological information such as the entity's hierarchical classes are utilized with their word embeddings. For each entity, its hierarchical classes' word embeddings are combined with its own word embedding by concatenation, averaging or weighted averaging. Wang et al. [103] propose a KG completion method InductiveE which can deal with unseen entities using entity textual descriptions. It encodes an entity by concatenating its text embeddings by the fastText word embedding model [115] and the pre-trained BERT [116]. For each triple, it feeds the entities' encodings into a model composed of an encoder — a gated-relational GCN and a decoder — a simplified version of ConvE [117] to predict a score.

Recently, due to the wide investigation of pre-trained language models, some methods that fine-tune these models for utilizing textual information for addressing zero-shot KG completion have been proposed. Different from [103] where BERT is used for initial but fixed entity encoding, the entity and relation encoding in the following methods are trained jointly with f' as the pre-trained language model BERT is fine-tuned. Yao et al. [100] propose a KG triple prediction method called KG-BERT. It transforms each triple into a text sentence with the name information of its head entity, predicate and tail entity, and then makes triple prediction as a downstream text classification task, where BERT is fine-tuned with triples for training. Zha et al. [104] also propose to predict triples as a downstream text classification task of BERT. They fine-tune BERT using not only single triples but also possible paths that connect two entities (reasoning is conducted explicitly to discover usch paths). Wang et al. [105] attempt to address two cons of KG-BERT: the combinatorial explosion in triple inference and the failure to utilize structured knowledge. They propose a structure-aware encoder to represent a triple's text with different combinations and interactions between its entities and relations. Wang et al. [102] propose a joint text and entity embedding method named KEPLER which is also able to predict KG triples with unseen entities and relations. It utilizes the text of the entities and the relations to fine-tune the BERT model via a masked language modeling loss.

Besides KG completion, we also find two KG-aware zero-shot question answering methods and on KG-aware zero-shot knowledge extraction method that belong to text feature fusion. Banerjee et al. [87] perform question answering via triple learning where the context, question and answer are modeled as a triple, and one of them is predicted given the other two. In implementation, a transformer-based model that generates the answer given the text features of the context and question is learned by span masked language modeling, using triples extracted from text. Zhou et al. [79] also model question answering as triple prediction with all the text features fused, and learn the prediction model by alternatively masking the subjects and the objects of the training triples which are from a corpus named WorldTree. Gong et al. [106] fine-tune a BERT model for zero-shot relation extraction, where prompts are constructed as the input using the relation's corresponding knowledge in ConceptNet.

2) Multi-modal Feature Fusion: In these methods, the input encoding and the class encoding are of different types. Due to the heterogeneity of the inputs, the mechanism of f', especially the fusion model, will differ from that of the text feature fusion category. Nguyen et al. [58] work on cross-domain entity recognition from the text, where the testing entities are from a different domain and unseen, The input is a sequence encoded as token features by a pre-trained BERT, while the class is an entity encoded as graph features learned by a Recurrent GNN over an ontology. These two different kinds of features are fed into an integration network for fusion. Ristoski et al. [107] work on zero-shot entity extraction. They fuse the features of the entity mention and entity description (input), with the entity's graph vector (output) which encodes the entity's KG semantics such as types. Zhang et al. [59] work on zero-shot text classification. They fuse the input text encoding and the ConceptNet-based class encoding, and feed them into a CNN classifier. The class encoding encodes the associated entity of the class, its ancestors and its description entities, using multi-hot encoding.

Paradigm	Summary	Categories	Papers
Mapping-based	These methods project the input and/or the class into a common vector space where a sample is close to its class w.r.t. some distance metric, and prediction can be implemented by searching the nearest class. ZSL methods can often be extended for FSL.	Input Mapping	[52, 63, 53]
		Class Mapping	[85]
		Joint Mapping	[93, 63, 45, 118, 86, 84, 119, 19, 65, 76]
Data Augmentation	These methods generate additional samples or sample features for the unseen classes, utilizing KG auxiliary information.	Rule-based	[21]
		Generation Model-based	[120, 24]
Propagation-based	These methods propagate model parameters, or class embeddings (or a sample's class beliefs) from the seen classes to the unseen classes via a KG.	Model Parameter Propagation	[23, 51]
		Embedding Propagation	[121, 122, 123, 124, 125, 126, 127]
Class Feature	These methods encode the input and the class into features often with their KG contexts considered, fuse these features and feed them directly into a prediction model.	Text Feature Fusion	[87]
		Multi-modal Feature Fusion	[128, 61, 129]
Optimization-based	These methods adopt meta learning algorithms to optimize the training that relies on the few-shot samples.	KG-specific Optimization	[130, 131, 132]
		KG-agnostic Optimization	[133, 134, 66, 119, 55, 65]
Transfer-based	These methods directly apply models of seen classes to unseen classes, often with the few-shot samples utilized in prediction.	Neural Network Transfer	[135, 136, 137]
		Rule Transfer	[138, 139, 140, 141, 142]

TABLE III: A summary of KG-aware FSL paradigms.

V. KG-aware Few-shot Learning

Many KG-aware FSL methods also follow the four paradigms of KG-aware ZSL: *Mapping-based*, *Data Augmentation*, *Propagation-based* and *Class Feature*. However, some other KG-aware FSL methods, belong to none of the above. Instead, we regard those that focus on utilizing the few-shot samples by accelerating the adaption in training with meta learning algorithms, as a new paradigm named *Optimization-based*, and regard those that directly transfer models (such as rules) built according to data of seen classes as another new paradigm named *Transfer-based*. Figure 1 presents these paradigms and their method categories while Table III summarizes the paradigms and lists the papers of each category. We will next introduce the details of each paradigm.

A. Mapping-based Paradigm

The general idea of the mapping-based paradigm of FSL is very close to that of ZSL, which is to train an input mapping model $\mathcal{M}$, a class mapping model $\mathcal{M}'$ or both $\mathcal{M}$ and $\mathcal{M}'$ as shown in Figure 5. In contrast to ZSL, FSL has a small number of labeled samples associated with each unseen class ($\mathcal{D}_{few}$). They usually can play an important role and are fully utilized. For example, they are sometimes used as another kind of auxiliary information of the classes. As ZSL, we further categorize the mapping-based KG-aware FSL methods into *Input Mapping*, *Class Mapping* and *Joint Mapping*. Next we will introduce the methods of each category, from the dimensions of input encoding, class encoding, mapping function(s), comparison metric and few-shot sample utilization.

1) Input Mapping: ZSL methods of input mapping can often be directly extended to FSL by augmenting the learning of the mapping model $\mathcal{M}$ with the few-shot samples. Ma et al. [63] support zero-shot and few-shot entity mention typing at the same time. They get the initial class (entity type) encoding via KG embedding and then directly map the input encoding (entity mention features) to the space of the class encoding. In the few-shot setting, they train the mapping mode $\mathcal{M}$, i.e., a linear transformation function with both training samples $\mathcal{D}_{tr}$ and few-shot samples $\mathcal{D}_{few}$. Jayathilaka et al. [52] also learn a mapping model $\mathcal{M}$ using both $\mathcal{D}_{tr}$ and $\mathcal{D}_{few}$. They represent logical relationships such as class disjointness and class subsumption by an OWL ontology and embed it by a logic embedding algorithm named EL Embedding [143] for initial class encoding. Monka et al. [53] investigate KG-augmented few-shot image classification. They use a KG curated by experts for modeling the relationship between classes, embed the KG by a variant of GCN for class encoding, and adopt a contrastive loss to train a MLP as $\mathcal{M}$ for mapping the image features.

2) Class Mapping: The learning of the class mapping model $\mathcal{M}'$ can also be directly extended using both $\mathcal{D}_{tr}$ and $\mathcal{D}_{few}$. However, such extension has been rarely investigated. The only KG-aware FSL method that belongs to class mapping is by Li et al. [85]. It maps the embeddings of hierarchical classes (class encoding) into the space of image CNN features (input encoding) for both zero-shot and few-shot image classification. The mapping function learning does not use $\mathcal{D}_{few}$, but in prediction, the average of the CNN features of the few-shot samples as well as the mapped class vector are both used and compared with the input of a testing sample.

3) Joint Mapping: Some KG-aware FSL methods of joint mapping are also simple extensions of their ZSL counterparts. Akata et al. [93, 45] jointly map the WordNet-based class encoding and the image encoding into one common space for few-shot image classification, where the mapping models $\mathcal{M}$ and $\mathcal{M}'$ are trained with an additional loss on few-shot samples

$\mathcal{D}_{few}$. Similarly, Ma et al. [63] utilize $\mathcal{D}_{few}$ to augment the training of the mapping models for few-shot entity typing. Note they consider not only input mapping, but also joint mapping. Rios et al. [76] also augment the joint training of $\mathcal{M}$ and $\mathcal{M}'$ with $\mathcal{D}_{few}$, for KG-aware few-shot text classification, where the input text encoding is based on CNN while the class encoding uses GCN.

Some other KG-aware FSL methods of joint mapping are specifically developed for utilizing the few-shot samples $\mathcal{D}_{few}$. They regard $\mathcal{D}_{few}$ as a kind of auxiliary information, and map them and the testing samples into one common vector space. Li et al. [86] jointly learn a mapping of the image CNN features (input encoding) and a mapping of the class encoding, using the training samples $\mathcal{D}_{tr}$. In prediction, they calculate the center of the mapped vectors of few-shot images of each class, and compare a testing image to this center. Note learning the mapping of the class encoding impacts the learning of the mapping of the image features. Xiong et al. [118] work on one-shot KG completion with unseen relations. They develop a matching network to compare a testing entity pair with the one-shot entity pair of each unseen relation, where the features of an entity pair (i.e., input encoding) are learned by a neighbourhood encoder, and a matching score is predicted by an LSTM network. Zhang et al. [119] work on few-shot KG completion with unseen relations, with a similar idea as the above work [118]. Zhao et al. [84] jointly map the image features (input encoding) and the class encoding (which is based the fusion of KG embeddings and text embeddings) into one common space by MLPs. In prediction, a testing sample is compared with the few-shot samples of each unseen class via calculating the Consine similarity. As the method in [86], the mapping learning of the class encoding impacts the mapping learning of the input encoding. Sui et al. [19] propose a KG-aware few-shot text classification method. It maps and compares a testing sample with the few-shot samples of each unseen class using a task-agnostic relation network and a task-relevant relation network armed with external knowledge from NELL. Zhang et al. [65] work on few-shot relation extraction from text, utilizing concept-level knowledge from Wikidata [34] or UMLS [75]. They match the mapped testing sample (i.e., an entity mention pair) to the mapped few-shot samples and to the mapped relation (class) encoding, and combine the two matching scores. Note sample input mapping model is a network which considers the sentence features, the entity description features and the KG concept features, while the relation encoding is based on the relation representations extracted from the KG embeddings.

B. Data Augmentation Paradigm

There have been some FSL studies that attempt to generate additional samples or sample features for the unseen classes by using KGs. We divide these methods into two categories: *Rule-based* and *Generation-based*. Although rules (heuristics) can be directly applied to FSL by e.g., annotating labels to samples as in distant supervision, we only find one KG-aware FSL work of this category. Instead, we find some KG-aware FSL studies of generation-based, which usually utilize statistical generation models such as GANs [109] and VAEs [110]. Next we will introduce the works of each category.

1) Rule-based: Tsai and Salakhutdinov [21] use a simple heuristic rule for sample generation for one-shot image classification. They take an attention mechanism over the class encodings, which are based on the embeddings of a KG extracted from WordNet, to select the most relevant unseen classes for a seen class, such that samples of the seen class are transformed into a set of quasi-samples of these unseen classes as additional training samples.

2) Generation-based: Some works leverage GANs and VAEs to generate extra labeled data for the unseen classes conditioned on their auxiliary information, as the generation-based category in KG-aware ZSL. For example, Wang et al. [120] work on few-shot KG completion involving both unseen entities and unseen relations, and propose a triple generator with Conditional VAE [144] to supplement the real triple set. Following the basic idea of VAE, the encoder is implemented with a recognition network and a prior network to learn the variational posterior distribution $q_\theta(z|O)$ and the conditional prior distribution $p_\phi(z|o_r)$ by taking as input the embedded textual descriptions of the triples O and that of the relations o_r, respectively. Next, the decoder by a generative network is proceeded to reconstruct the triple embeddings (g_h, g_r, g_t) by sampling them from the latent semantics z conditioned on o_r, i.e., $p_\varphi(g_h, g_r, g_t|z, o_r)$. During testing, more triple embeddings of unseen entities and unseen relations can be generated conditioned on their latent semantics z.

Besides obtaining additional training data using generative models, there are also some other works that are motivated by the application of GANs in domain adaption. They attempt to generate features that are more transferable from the "lots-of-samples" domains to the different but related "few-samples" domains. For example, Zhang et al. [24] work out a general feature generation framework for addressing few-shot unseen relations in two tasks — few-shot KG completion with unseen relations and few-shot relation extraction from text. The framework is adversarially trained to generate the features that are invariant to the seen and unseen relations, and transfer such features to unseen relations with weighted combination. The feature generation module is implemented by a CNN to iteratively extract features from the entity pair (or from the text sentence for relation extraction) until the discriminator cannot distinguish features of the seen relations and the unseen relations.

C. Propagation-based Paradigm

We find two KG-aware FSL studies that adopt the idea of model parameter propagation as in KG-aware ZSL but no studies adopting class belief propagation. This may be because the current methods usually focus on utilizing the few-shot samples. For few-shot KG completion tasks, graph propagation is widely utilized for addressing unseen entities or unseen relations that have few-shot associated triples. They often aggregate the embeddings of the neighbouring entities and relations, which are usually seen, to get the embedding of an unseen relation or entity. Figure 7 (b) shows this idea

with an example. The entity e_0, which is unseen without a trained embedding, is connected to seen entities e_1, e_2 and e_3 through some few-shot triples with relations of r_1, r_2 and r_3. With a propagation model, the embedding of e_0 can be predicted via the trained embeddings of e_1, e_2 and e_3. We classify these methods into a new category named *Embedding Propagation*. We will next introduce the works of model parameter propagation and embedding propagation, mainly from two dimensions — the propagation graph and the propagation model.

1) Model Parameter Propagation: Peng et al. [23] work on WordNet augmented few-shot image classification. They first do model parameter propagation as in KG-aware ZSL, which uses a GCN and a graph to predict classifier parameters of the unseen classes, and then ensemble these predicted classifiers with the classifiers learned from the few-shot samples. Chen et al. [51] use a graph whose nodes are seen and unseen classes, and whose edges are assigned by correlation weights between classes for few-shot image classification. They initialize the classifier parameters of each graph node (class), and then use a Gated Graph Neural Network (GGNN) to update the classifier parameters of each graph node with multiple iterations. The GGNN is trained with a cross-entropy loss on $\mathcal{D}_{tr}$ and $\mathcal{D}_{few}$ a regularisation term on the classifier parameters.

2) Embedding Propagation: The embedding propagation methods mainly aim at few-shot KG completion tasks which predict triples involving unseen entities or unseen relations, without re-training the embedding of the original KG. Thus the graph for propagation is the KG to complete itself. These unseen entities (resp. relations) are also named as *out-of-KG entities (resp. relations)* in some papers since they are usually not observed in the KG whose embeddings have been trained.

As far as we know, Hamaguchi et al. [121] propose the earliest embedding propagation method for KG completion with unseen entities. They use a GNN but revise its propagation mechanism for the KG, and adopt a translation-based objective function for scoring a triple and for a loss for training. Wang et al. [122] propose a Logic Attention Network (LAN) to get the embeddings of unseen entities from their neighbouring entities and relations. In LAN, logic rules are exploited to measure neighbouring relations' usefulness, and neighbours connected by different relations have different weights w.r.t. an unseen entity. Bhowmik and Melo [124] use a variant of Graph Transformer encoder to embed an unseen entity by aggregating its neighbours based on their relevance to a given relation. It predicts the object of a triple, and can explain the prediction by finding out paths from the subject to the object. Ali et al. [127] predict relations between seen entities and unseen entities, and between unseen entities. For predicting relations between unseen entities, they initialize the entity embeddings by the entities' textual information using Sentence BERT, and then propagate to update the entities' embeddings by a graph encoder named StarE [145].

Some simpler propagation models have also been explored for KG completion with unseen entities. Besides GNN, Ali et al. [127] also explore a linear projection of entity features to relation features without considering the graph structure. This could also be classified as a mapping-based method by considering the entity features as input encoding and the relation features as class encoding. Dai et al. [126] use two modules: an estimator which calculates a candidate set of embeddings for an unseen entity according to its all associated triples using the translation operation of TransE (or RotatE), and a reducer which calculates the unseen entity's embedding according to all its candidate embeddings. Albooyeh et al. [123] use some simple aggregation operations such as averaging to get the embedding of an unseen entity from its neighbours. This method can be applied to any KG embedding models, but it requires that the original KG embedding training is adjusted for the aggregation operation.

All the above mentioned few-shot KG completion methods deal with unseen entities. We also find one embedding propagation method [125] that can deal with both unseen entities and unseen relations. It mainly uses specific transition functions, aggregation functions and graph attention mechanisms to transform information from the associated triples to an unseen entity or relation, where a triple is scored by a translation-based function and the model is trained with a margin loss. Note that this method does not deal with the situation with both unseen entities and unseen relations.

D. Class Feature Paradigm

The class feature paradigm of FSL is close to that of ZSL, as shown in Figure 8. As the mapping-based paradigm, many KG-aware ZSL methods of the class feature paradigm can be directly extended to support FSL by training f' with both the training samples $\mathcal{D}_{tr}$ and the few-shot samples $\mathcal{D}_{few}$. In this part, we focus on those class feature paradigm works that are originally proposed for FSL. Some such works are found but not many since class feature fusion under the FSL setting does not significantly differ from class feature fusion under normal supervised learning settings. We classify them into two categories: *Text Feature Fusion* where the input encoding $g(x)$ and the class encoding $h(y)$ are both of text features, and *Multi-modal Feature Fusion* where $g(x)$ and $h(y)$ are of different kinds of features. Next we will introduce the works of each category from the dimensions of input encoding, class encoding and the fusion model.

1) Text Feature Fusion: We find one KG-aware FSL work that belongs to text feature fusion. Banerjee et al. [87] propose a KG-aware method for few-shot question answering. The input (the text context and the question) and the class (answer) are fed into a transformer-based model as f'. This model is learned by span masked language modeling from KG triples, each of which simulate a combination of context, question and answer. Note that the method can also support ZSL as introduced in Section IV-D.

2) Multi-modal Feature Fusion: Fusing input encoding and class encoding of different kinds is harder and often requires a more complicated fusion model. We find two KG-aware FSL studies of this category. Zhang et al. [128] investigate text relation extraction for long-tailed relations which have few-shot samples (sentences). The proposed method uses a GCN to learn the embedding of each relation as class encoding, over a KG derived from Freebase where the hierarchical

relationship between relations are modeled, and then feeds the class encoding and the sentence features (input encoding) into an attention-based model. Yang et al. [61] investigate few-shot visual question answering. A baseline that they adopt, named KRISP [129], uses KGs such as ConceptNet for augmentation and is applied to this task. In KRISP, the features of the image and the question are first fused by a Transformer-based model for input encoding, and then the input encoding is further fused with the class encoding (features of knowledge retrieved from the KG) to predict the answer.

E. Optimization-based Paradigm

To optimize the training with the few-shot samples $\mathcal{D}_{few}$, some meta-learning algorithms have been applied for fast adaption and for avoiding over-fitting by obtaining better initial parameter settings, more optimized searching steps or more suitable optimizers. We regard these FSL methods as *Optimization-based Paradigm* and present their general workflow in Figure 9. To mimic the learning with the few-shot samples that are assumed to be available in testing, they all adopt an episode-based training strategy and generate a set of learning tasks. Each task consists of a support set and a query set to simulate the few-shot sample set $\mathcal{D}_{few}$ and the testing samples $\mathcal{D}_{te}$, respectively, and aims at learning a Meta Learner parameterized by θ, which is expected to be able to learn efficiently from only a small number of samples. After multiple training iterations on these training tasks, the method can obtain an optimal Meta Learner. More formally, in a training task t, we have the support set $\mathcal{D}_t^{support}$ and the query set $\mathcal{D}_t^{query}$, the Meta Learner F_θ can either automatically produce a model f from $\mathcal{D}_t^{support}$ to predict the samples in $\mathcal{D}_t^{query}$, or learn optimized gradients ∇_f from $\mathcal{D}_t^{support}$ to better update the model parameters on $\mathcal{D}_t^{query}$. The optimization at each step t is generated by computing the loss on the query set $\mathcal{D}_t^{query}$. When at test time, a set of tasks disjoint with the training tasks is designed for evaluation, where the support set is from $\mathcal{D}_{few}$, and the query set is $\mathcal{D}_{te}$ we need to predict.

We collect quite a few KG-aware FSL methods of this paradigm: some of them are for KG completion tasks with unseen entities or relations that are associated with a small number of triples [131, 130, 132, 119, 134, 133], while the others are for KG augmented few-shot image classification and few-shot text relation extraction [66, 55, 65]. We find some studies develop new meta learning algorithms or revise the existing ones w.r.t. the KG, while some other studies just apply meta learning independently without specifically considering the KG context. We thus classify these FSL methods into *KG-specific Optimization* and *KG-agnostic Optimization*.

1) KG-specific Optimization: Chen et al. [131] propose a new method named MetaR to predict triples involving unseen relations which have only a small number of associated triples. A learning task is defined for a specific relation, and each sample is its one associated entity pair and the number of support samples is set to be no more than 5. Firstly, a relation-meta learner is designed to extract higher-order relation representations from the embeddings of the support

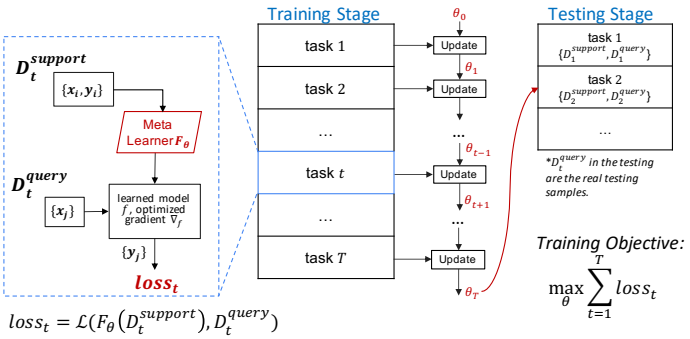


Fig. 9: Overview of the optimization-based paradigm of FSL.

entity pair as relation meta, and then the gradient meta, which will guide how the relation meta should be efficiently updated, is generated by feeding the relation meta and each entity pair into an embedding learner to compute the triple score. Finally, the updated relation meta is transferred to the triples in the query set to compute their scores via the same embedding learner. The loss of query set is used to update the whole model so as to quickly learn better relation meta for testing when only a small number of support samples are given.

Meanwhile the typical meta learning algorithm Model-Agnostic Meta-Learning (MAML) which is to learn a good parameter initialization for a new meta-learning task [146] is often adopted and augmented with KG. Wang et al. [130] work on a few-shot KG reasoning task which is to predict the tail entity given a head entity and an unseen relation and infer paths from the head entity to the tail entity. They augment MAML with additional task (relation) specific information encoded by a neighbour encoder based on embedding concatenation and linear transformation operations, and a path encoder based on LSTM. Baek et al. [132] work on a realistic few-shot KG completion task, where relations between seen entities and unseen entities, and between unseen entities are both predicted using GNNs. They propose a Graph Extrapolation Network for quickly learning the embeddings of unseen entities with only a few associated triples, where a set of tasks are formulated with simulated unseen entities so as to generalize to the real unseen entities raised at test time.

2) KG-agnostic Optimization: In some other optimization-based FSL studies, the application of meta learning algorithms is independent of the KG. Note that some of these studies are still reviewed as they aim at KG related prediction tasks. Lv et al. [133] work on the same task as [130], i.e., few-show KG completion with unseen relations. They adopt reinforcement learning to search tail entities and paths which could infer these tail entities, and directly apply MAML with one relation modeled as one task. Zhang et al. [134] propose another method for few-shot KG completion with unseen relations, where MAML is directly applied for well initializing an on-policy reinforcement learning model for fast adaption. Qu et al. [66] work on few-shot relation extraction by modeling the posterior distribution of prototype vectors for different relations. To this end, they first initialize the relation prototype vectors by a BERT model over the samples (i.e., sentences) and a

GNN over a global relation graph extracted from different ways, and then effectively learn their posterior distribution by a Bayesian meta-learning method which is related to MAML but can handle the uncertainty of the prototype vectors.

It is worth mentioning that meta learning-based optimization can simply act as a complement for model training in methods of other paradigms. Zhang et al. [119] predict KG triples with unseen relations. Their few-shot relational learning method FSRL, which is of the mapping-based paradigm as it compares a testing entity pair with few-shot samples of each unseen relation after mapping, uses MAML for fast adaption in training the mapping models. Zhang et al. [55] attempt to address both zero-shot and few-shot image classification, with an approach named Transfer Graph Generation (TGG) which has a graph generation module for generating instance-level graph, and a propagation module for utilizing this graph for prediction. They train the whole model with an episodic training strategy of meta learning. Zhang et al. [65] use a joint mapping method to predict relations for entity mentions in a sentence. In this method, a knowledge-enhanced prototypical network and a relation meta learning model, which implement the matching between instances and the matching between instance and relation meta, respectively, are trained with gradient meta.

F. Transfer-based Paradigm

Some KG-aware FSL methods directly apply models that are built from samples of seen classes ($\mathcal{D}_{tr}$) to predicting samples of unseen classes ($\mathcal{D}_{te}$) with the help of the few-shot samples ($\mathcal{D}_{few}$). These methods are regarded as the transfer-based paradigm. It is worth noting that some methods of the other paradigms such as the model parameter propagation methods also have an idea of implicitly transferring data or parameters from seen classes to unseen classes. The difference is that methods of the transfer-based paradigm directly apply the whole prediction models learned from $\mathcal{D}_{tr}$ to $\mathcal{D}_{te}$. They are often for few-show KG completion, where one KG is given for model learning, while another KG composed of triples of unseen entities is for inference (prediction), as shown in Figure 10. For convenience, we name the first KG as the seen KG and the second KG as the unseen KG. Such a task in common in real-world: given an existing KG (seen), the unseen KG could be either an emerging sub-KG that is to be added, or a KG of another domain with shared relations but different entities. We further classify these FSL methods into two categories: *Neural Network Transfer* and *Rule Transfer*. We will next introduce the works of each category mainly from the dimension of the specific model to transfer.

1) Neural Network Transfer: As GNNs can represent statistical regularities and structure patterns in a graph, some methods transfer a GNN learned from the seen KG to the unseen KG for inference. Teru et al. [135] propose a method named GraIL. It learns a GNN by extracting subgraphs from the seen KG and labeling their entities with their structural roles (e.g., the shortest distance between two entities), and apply this GNN to predict the relation between two unseen entities in the unseen KG. Chen et al. [137] extend GraIL by using R-GCN [147] for supporting multiple relations in the

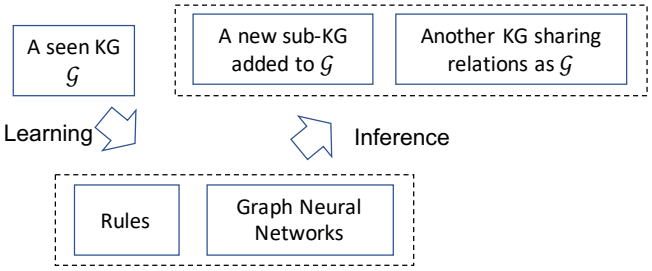


Fig. 10: Overview of the transfer-based paradigm of FSL when applied to KG completion.

KG. Besides, they propose to transfer another model named Relational Correlation Network learned from the seen KG to the unseen KG, and combine its triple score with that by the extended GraIL. Note that the relational correlation network is learned from a relation correlation graph whose nodes represent the relations and whose edges indicate the topological correlation patterns between any two relations in the original KG. Liu et al. [136] propose to reformulate the original KG as a graph as follows: two connected KG entities or an entity and its own, are represented as one graph node, and each node is initialized with features indicating the triples in which the two entities are involved. They learn a GCN from the graph of the seen KG, which is shown to be able to capture graph patterns represented in Datalog rules, and apply this GCN to predict graph node features (i.e., triples) of the unseen KG.

2) Rule Transfer: Different rules such as Horn rules, first-order rules and their weighted versions can be learned from a KG for represent graph patterns and regularities [148, 149]. They may not be as expressive as GNNs for representing very complicated statistical regularities, but are more interpretable. Sadeghian et al. [138] propose a method named DRUM for few-shot KG completion, where first-order logical rules (such as $brother(X, Z) \wedge fatherOf(Z, Y) \rightarrow uncleOf(X, Y)$) associated with weights are learned from the seen KG by a differentiable way using the rule mining method named Neural LP, and these rules are applied in the unseen KG for deductive reasoning for new triples. This method uses the KG relations as the rule predicates and assumes that the relations of the seen and the unseen KGs are the same, such that the rules can be directly transferred. For the situation where predicates (relations) of the unseen KG are different from those of the seen KG, we find the following two solutions for rule transfer: *(i)* matching predicates between rules, proposed by Mihalkova et al. [139, 140] for transferring rules mined by Markov Logic Networks (MLNs), and *(ii)* extracting and transferring more general higher order rules that are summarized from the original rules [141, 142].

G. ZSL and FSL Comparison

Some KG-aware FSL methods are specifically developed to utilize the few-shot samples. Typical kinds of such methods are the optimization-based paradigm, the transfer-based paradigm, and the embedding propagation category of the propagation-based paradigm. The meta learning algorithms used in the

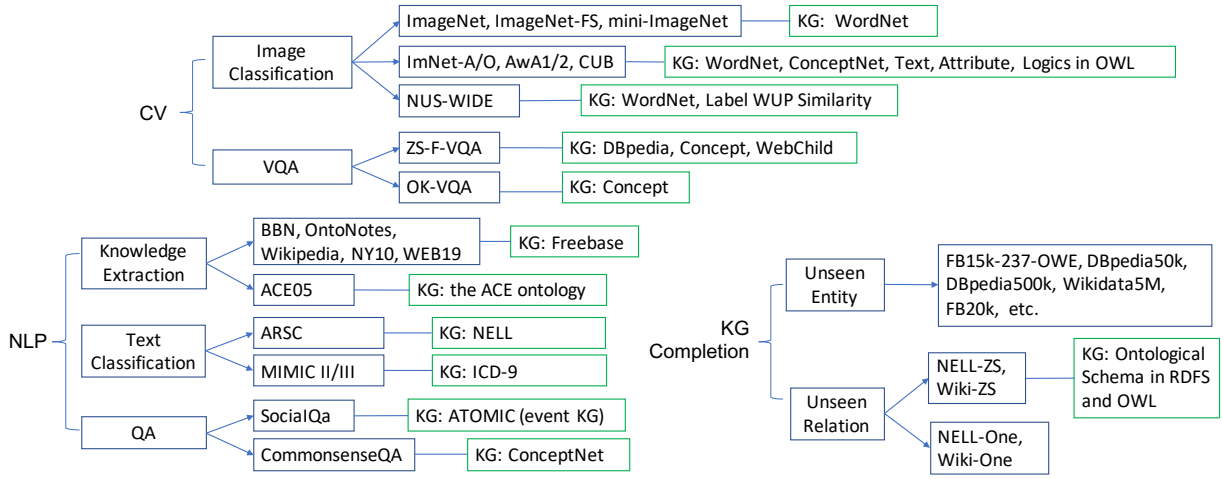


Fig. 11: Overview of the benchmarks of different tasks and their KGs. The green box denotes the KG auxiliary information. Some KG completion benchmark without additional KG (ontology) auxiliary information are also collected, as the zero-shot or few-shot task itself is within a KG context.

optimization-based paradigm and the models directly transferred both rely on the few-shot samples, and thus cannot be applied to ZSL. For example, when a set of rules or a GNN are transferred to predict triples involving unseen entities, these unseen entities must be associated with some triples for evidences (graph patterns) for inference. Many of these FSL methods aim at KG completion tasks. They often ignore or do not well utilize the auxiliary information.

Meanwhile, some other KG-aware FSL methods are simple extensions of corresponding KG-aware ZSL methods. They train the original ZSL models with the additional few-shot samples, or ensemble the ZSL models with the models trained from the few-shot samples. This is common in methods of the mapping-based paradigm and the class feature paradigm. The mapping models and the fusion models can be trained with both the training samples and the few-shot samples (e.g., [63, 45, 76]). Actually, the majority of the ZSL methods, which usually well utilize the auxiliary KG, can be extended to support FSL with the above extension ideas, although for the class belief propagation category, there are currently only ZSL works but no FSL works. However, well combing the KG (or some other auxiliary information) with the few-shot samples is still an open problem.

VI. Applications and Resources

In this section, we first very briefly revisit KG-aware ZSL and FSL in different domains and tasks, and then introduce some public resources. See Table IV for a summary of methods of each task, and see Figure 11 for an overview of benchmarks of each task.

A. Computer Vision

1) Image Classification: Regarding zero-shot image classification[16], the early works mainly utilize class attributes [2, 28, 3, 29] and class text information [153, 154, 10, 30, 11, 155], with the mapping-based paradigm and the data augmentation paradigm often adapted. However, the state-of-the-art performance on many benchmarks now are achieved by those methods utilizing KGs constructed by various sources including existing KGs, task-specific data and domain knowledge [55, 56, 54, 12, 13, 14]. To utilize the KGs, the propagation-based paradigm starts to be widely adopted in some recent studies such as [12, 13, 47].

To support method development and evaluation, some open benchmarks on KG-aware zero-shot image classification have been proposed:

- **ImageNet** is a large-scale image database containing a total of 14 million images from 21K classes [156]. Each image is labeled with one class, each class is matched to a WordNet [32] entity, and the class hierarchies from WordNet can be used as the auxiliary information. In the works [12] and [13], 1K classes with balanced images are used as seen classes for training, while classes that are 2-hops or 3-hops away, or all the other classes are used as unseen classes for testing. The weakness of ImageNet mainly lies in that the KG has only class hierarchies and class name) without any other knowledge such as class attributes and commonsense knowledge.
- **ImNet-A** and **ImNet-O** are extracted from ImageNet by Geng et al. [14, 91]. ImNet-A includes 80 classes from 11 animal species, while ImNet-O including 35 classes of general objects. In the experiment in [14], ImNet-A is partitioned into 28 seen classes (37,800 images) and

[16]Note that object recognition is often transformed into two steps: discovering object bounding boxes and classifying these bounding boxes. Zero-shot object recognition in many works (e.g., [150]) usually aim at the second step and are equivalent to zero-shot image classification. There are also some works using KGs for supporting object recognition [151, 152].

Tasks	Paradigms	ZSL Works	FSL Works
Image Classification	Mapping-based	[1, 93, 45, 46, 83, 42, 54, 85, 56, 88, 15]	[93, 45, 86, 84, 85, 52, 53]
	Data Augmentation	[55, 14]	[21]
	Propagation-based	[43, 12, 13, 44, 50, 77, 47, 49]	[23, 51]
	Optimization-based	—	[55]
VQA	Mapping-based	[88, 57]	—
	Class Feature	—	[129]
Knowledge Extraction	Mapping-based	[63, 73, 64, 67]	[63]
	Data Augmentation	—	[24]
	Class Feature	[58, 107, 106]	[128]
	Optimization-based	—	[66, 65]
Text Classification	Mapping-based	[76, 57]	[76, 19]
	Class Feature	[59]	—
QA	Propagation-based	[89]	—
	Class Feature	[87, 79]	[87]
KG Completion (Unseen Entity)	Mapping-based	[94, 95]	—
	Data Augmentation	—	[120]
	Propagation-based	—	[121, 122, 124, 123, 125, 126]
	Class Feature	[97, 98, 100, 103, 104, 105, 101, 102, 48, 127]	—
	Optimization-based	—	[132]
	Transfer-based	—	[138, 135, 136, 137]
KG Completion (Unseen Relation)	Mapping-based	—	[118, 119]
	Data Augmentation	[96, 31, 14]	[120, 24]
	Propagation-based	—	[125]
	Class Feature	[100, 104, 105]	—
	Optimization-based	—	[130, 131, 133, 134]

TABLE IV: A summary of KG-aware ZSL and FSL works of different tasks.

52 unseen classes (39,523 images), while ImNet-O is partitioned into 10 seen classes (13,407 images) and 25 unseen classes (25,954 images). In their latest version released in [91], each benchmark is equipped with a KG which is semi-automatically constructed with several kinds of auxiliary knowledge, including class attribute, class textual information, commonsense knowledge from ConceptNet, class hierarchy (taxonomy) from WordNet and logical relationships such as disjointness.

- **AwA2**, originally proposed in [27], has 50 animal classes, $37,322$ images collected from public Web sources such as Flickr and Wikipedia, and 85 real-valued attributes annotated by experts for describing animal visual characteristics. It can also be used to evaluate KG-aware ZSL methods, since the classes are aligned with WordNet entities and the animal taxonomy from WordNet can be used as a simple KG. In the extended version by Geng et al. [91], a KG is constructed for AwA2 with the same types of knowledge as ImNet-A and ImNet-O. Note AwA in [91] actually refers to AwA2, while the original AwA1 released in [28] does not have public copyright license for all of its images.
- **NUS-WIDE** [157] is a multi-label image classification dataset including nearly 270K images. Each image contains multiple objects, and thus NUS-WIDE is widely used for evaluating multi-label zero-shot image classification [43, 158, 159]. To be more specific, the images have two versions of label sets: NUS-1000 and NUS-81. The former comprises 1000 noisy labels collected from Flickr user tags and the latter is a dedicated one with 81 human-annotated labels. To perform multi-label ZSL, the labels in NUS-81 is taken as the unseen label set, while the seen label set is derived from NUS-1000 with 75 duplicated ones removed and thus results in 925 seen label classes. In works on KG-aware ZSL such as [43], NUS-WIDE is accompanied by a KG with 3 types of label relations, including a super-subordinate correlation from WordNet, positive and negative correlations computed by label similarities such as WUP similarity [160].

For few-shot image classification, the majority of the existing methods aim at utilizing the few-shot samples by e.g., meta learning, while the KG-aware studies often try to combine benefits from the KG external knowledge and the few-shot samples. Some of them simply extend their mapping-based models which are originally developed for zero-shot image classification by training with additional samples of the unseen classes (e.g., [52, 45, 86]), while some others further generate more data for unseen classes conditioned on KGs (e.g., [21]) or utilize KGs to transfer images features from seen classes to unseen classes (e.g., [51, 23]).

There are also some open benchmarks that can be used for KG-aware few-shot image classification. The following are some widely used ones:

- **ImageNet-FS** [161] and **mini-ImageNet** [162] are two derivatives of ImageNet. ImageNet-FS covers $1,000$ ImageNet classes with balanced images and these classes

are divided into 389 seen classes and 611 unseen classes. During evaluation, images of 193 seen classes and 300 unseen classes are used for cross validation, while images of the remaining 196 seen classes and 311 unseen classes are used for testing. in contrast, mini-ImageNet is relatively small. It has 100 classes, each of which has 600 images. These classes are partitioned into 80 seen classes and 20 unseen classes. Since all the classes are aligned with WordNet entities, WordNet can be used as the external knowledge.

- **AwA1** [28], **AwA2** [27] and **CUB** [163] are three typical zero-shot image classification benchmarks that can be easily extended for a few-shot setting. AwA1 and AwA2 both have 50 coarse-grained animal classes, with 40 of them being seen classes and the remaining being unseen classes. CUB has 200 fine-grained bird classes, with 150 of them being seen classes and the remaining be unseen classes. A small number of labeled images (usually 10) are added for each unseen class so as to support a few-shot setting. Meanwhile, several KGs have been added to these benchmarks for evaluating KG-aware methods: Tsai and Salakhutdinov [21] and Akata et al. [93, 45] add WordNet classes hierarchies to AwA1 and CUB; Zhao et al. [84] construct a domain-specific KG for CUB based on the attribute annotations of samples; Zhang et al. [55] exploit ConceptNet to construct a KG for AwA2, and utilize part-level attributes to construct a KG for CUB.

2) Visual Question Answering (VQA): VQA is to answer a natural language question according to a given image. Teney et al. [164] first propose zero-shot VQA as the setting where there are unseen concepts in the text. Namely, a testing sample is regarded as unseen if there is at least one novel word in its question or answer. Ramakrishnan et al. [165] consider novel objects in the image. Namely, an image object that has never appeared in the training images is regarded as unseen. KGs have been exploited for addressing zero-shot VQA, but not widely. The work [57] proposes a mapping-based method, where answers that have never appeared in training are predicted via comparing the KG-based embeddings of the question and answer embeddings. The work [88] also adopts the mapping-based paradigm, but builds and embeds an OWL ontology for establishing connections between seen answers and unseen answers.

A few VQA datasets have been published, but only a small number of them have been used for KG-aware zero-shot VQA:

- **ZS-F-VQA** [57], constructed by re-splitting a fact-based VQA benchmark named F-VQA [71], has no overlap between answers of the training samples and answers of the testing samples. In average, the training set has $2,384$ questions, $1,297$ images, and 250 answers, while the testing set has $2,380$ questions, $1,312$ images, and another 250 answers. Chen et al. [57] extract facts from three public KGs (DBpedia [68], ConceptNet [33] and WebChild [70]), and construct a auxiliary KG for evaluating KG-aware methods.
- **OK-VQA** [166] is a recent benchmark where the visual content of an image is not sufficient to answer the question. It has $14,031$ images and $14,055$ questions, and the correct answers are annotated by volunteers. Chen et al. [88] use it for evaluating KG-aware zero-shot VQA, by extracting 768 seen answers and 339 unseen answers, using auxiliary information from ConceptNet.

Regarding few-shot VQA, the existing methods (e.g., [61]) often rely on pre-trained language models such as GPT-3 which have already learned a large quantity of knowledge from text corpora. To incorporate images, visual language models can be pre-trained with images and text, or images can also be transformed into text by e.g., image captions so as to be utilized in language models [167]. Meta learning is also applied for fully utilizing the few-shot samples and fast model training [168].

KGs have complementary knowledge besides the pre-trained (visual) language models and the few-shot samples. We find some KG-aware few-shot VQA studies but no open benchmarks. Yang et al. [61] propose a supervised learning method which use knowledge retrieved from KGs for augmenting the question-answer samples, and this method is used as a baseline in comparison with the GTP-3-based method. Marino et al. [129] first fuse features of the question and the image by a Transformer-based model, and then fuse these features with knowledge from ConceptNet. On the other hand, the aforementioned mentioned zero-shot VQA benchmarks ZS-F-VQA and OK-VQA can be easily adjusted by adding few-shot samples for supporting the few-shot VQA setting.

B. Natural Language Processing

1) Knowledge Extraction: By knowledge extraction, we refer to those NLP tasks that are to extract structured knowledge including entities, relations, events and so on from natural language text. Relational facts, which are sometimes simply called triples in this domain, can also be extracted, after entities and relations are recognized. Since the entities, relations and events can often be aligned with elements in a KG (such as a general purpose KG and an event ontology), their relationships represented in the KG can be exploited to address both zero-shot and few-shot knowledge extraction. For these tasks, most KG-aware zero-shot methods follow the mapping-based paradigm utilizing the entities', relations' or events' embeddings in the KG [73, 63, 64, 67], while KG-aware few-shot methods often follow the optimization-based paradigm which utilizes meta learning algorithms for fast training [66, 65]. Some methods also consider the class feature paradigm by fusing features from a KG with the input features for both ZSL and FSL [58, 128].

There are quite a few knowledge extraction benchmarks that can be used for evaluating both KG-aware ZSL and KG-aware FSL. Here are several representative ones:

- **BBN**, **OntoNotes** and **Wikipedia** are three benchmarks for fine-grained named entity typing, where the entity types are (partially) matched with types in Freebase. They are all adopted by Ma et al. [63] for evaluating zero-shot entity typing, where the training set has only coarse-grained types, while the testing set has the second-level (fine-grained) types. They use a set of manually annotated

documents (sentences) for validation and testing with a partitioning ratio of 1:9. Specifically, BBN has $2,311$ manually annotated Wall Street Journal articles with around 48K sentences and 93 two-level hierarchical types [169]. 47 out of 93 types are mapped to Freebase types. 459 documents (6.4K sentences) are used for validation and testing. OntoNotes is an incrementally updated corpus that covers three languages (English, Chinese, and Arabic) and four genres (NewsWire, Broadcast News, Broadcast Conversation, and Web text) [170]. It has $13,109$ news documents that are manually annotated using 89 three-level hierarchical types. 76 manually annotated documents ($1,300$ sentences) are used for validation and testing. Wikipedia has around 780.5K Wikipedia articles (1.15M sentences), 112 fine-grained Freebase type annotations, and 434 validation and testing sentences.

- **NYT10** and **WEB19** are two benchmarks used in [64] for zero-shot relation (property) extraction. NYT10 is constructed by Freebase triples and New York Times (NYT) corpus [171]. WEB19 is formed by first selecting predicate paths in the FB15k benchmark [172] as properties, then generating samples (a text corpus) associated with these properties using Microsoft Bing search engine API with the aid of human evaluation [64]. Under the ZSL setting in [64], 217 and 54 properties of WEB19 are set to seen (for training) and unseen (for validation and testing), respectively, while a ll of the 54 properties of NYT10 are used as unseen (for testing).
- **ACE05** is a corpus for event extraction, annotated by 33 fine-grained types which are sub-types of 8 coarse-grained main types such as Life and Justice from the ACE (Automatic Content Extraction) ontology. Huang et al. [73] make two zero-shot event extraction settings: *(i)* predicting 23 unseen fine-grained sub-types by training on 1, 3, 5, or 10 seen sub-types; *(ii)* predicting unseen sub-types that belong to other main types by training on seen sub-types of Justice.

2) Text Classification: Few-shot text classification is similar to zero-shot text classification: the majority of the solutions mainly utilize different kinds of word embeddings, and the research on KG-aware method is rare. Rios et al. [76] propose an ontology augmented CNN classifier for both few-shot and zero-shot text classification, while Sui et al. [19] utilize knowledge retrieved from the NELL KG for augmenting a network which calculates the matching of the input and the class. Here are the benchmarks used in these two works:

- **MIMIC II** [173] and **MIMIC III** [174] are multi-label text classification benchmarks used in [76]. Their labels are concepts in the ICD-9 [17] ontology which is an international standard diagnostic classification for all general epidemiological, many health management purposes and clinical use. MIMIC II has $18,822$ labels for training and $1,711$ labels for testing; while MIMIC III has $37,016$ labels for training and $1,356$ labels for testing.
- **ARSC** is a popular benchmark for binary text classification of sentiment [175], generated from Amazon reviews for 23 products (classes). In [19], 12 products including books, DVDs, electronics and kitchen appliances are selected as the unseen classes, for each of which 5 labeled reviews are given, and NELL is used as the auxiliary KG.

3) Question Answering: Zero-shot and few-shot question answering (QA)[18] started to attract wide attention in recent years, mainly due to the fast development of pre-trained language models such as BERT and GPT-3 which are inherently capable of addressing ZSL and FSL problems in NLP since a large quantity of knowledge are learned from large-scale corpora as parameters [61, 176]. Similar to text classification, the output answer (class) is often regarded as an additional input and fed to a prediction model together with the question input. It is worth mentioning that the definition of zero-shot QA varies from paper to paper. Some are consistent with our general ZSL definition which mainly requests that the classes (answer labels) for prediction have no associated training data, while the others are not. For example, Ma et al. [176] simply regard testing the model on a QA dataset that is different from the training QA datasets as zero-shot QA; Wei et al. [177] fine-tune language models on a collection of datasets of some specific tasks (e.g., sentiment classification and summarization), test the models on datasets of different tasks (e.g., commonsen QA), and regard this as its zero-shot QA setting .

Although pre-trained language models have contained much knowledge via large scale parameters, symbolic knowledge (including commonsense and domain knowledge with logics) from KGs are often complementary and beneficial for addressing zero-shot QA. Therefore, there have been some KG-aware zero-shot QA studies [89, 87]. For example, Banerjee et al. [87] model the QA problem via knowledge triple learning where the context, question and answer are modeled as a triple, and the answer is predicted given the context and question. Their knowledge triple learning model is learned from KG triples. Similar to [87], Zhou et al. [79] also frame the multiple-choice QA task as a knowledge completion (triple prediction) problem, where the model is trained by alternatively masking the subjects and the objects in triples. Bosselut et al. [89] use COMET — a Transformer-based model trained on commonsense KGs such as ConceptNet to generate a context-relevant commonsense triples for each QA sample, and then infer the answer from these triples.

Similarly, KGs can also benefit few-shot QA. For example, Banerjee et al. [87] directly extend their knowledge triple learning model from zero-shot QA to few-shot QA where 8% of the training data are given as the few-shot samples. Bosselut et al. [89] also extend their zero-shot QA method, which infers the answer of a question according to their context-relevant commonsense triples, to few-shot QA by using 4, 10 or 20 validation samples in evaluation. However, due to challenges such as retrieving exactly relevant knowledge from a large KG and injecting KG knowledge into pre-trained language models,

[17]https://www.cdc.gov/nchs/icd/icd9.htm

[18]The scope of QA is actually quite wide. It often includes or has a high overlap with quite a few problems such as VQA, Knowledge Base QA, Table QA, Machine Reading Comprehension (MRC). In this part, we just refer to the problem of giving an answer or answers to a natural language question w.r.t. a context described by text.

the investigation of KG-aware zero-shot and few-shot QA is still quite preliminary.

There have been a few widely used QA benchmarks such as PhysicalIQA for commonsense physical reasoning [178]. They can be used for benchmarking zero-shot and few-shot QA after suitable dataset partitioning. We suggest benchmarks that are constructed with KGs or have been partially aligned with KG entities. The tasks of these benchmarks often rely on external knowledge, and their corresponding external KGs can be directly used for evaluating KG-aware methods. Here are some such benchmarks:

- **SocialIQa** [179] is a large-scale QA resource to evaluates a model's capability to understand the social dynamics underlying situations described in short text snippets. It has 38K question-answer pairs. Each sample consists of a context, a question about that context, and three multiple choice answers from crowdsourcing. Commonsense knowledge (i.e., seeds for creating the contexts and answers) are extracted from an event KG named ATOMIC [180]. This dataset is used by Bosselut et al. [89] and Banerjee et al. [87] for evaluating their KG-aware zero-shot and few-shot QA methods.
- **CommonsenseQA** [181] is a challenging dataset for evaluating commonsense QA methods. In total, it has $12,247$ questions, each of which has 5 answer candidates. The ground truth answers are annotated by crowdsourcing based on question relevant subgraphs of ConceptNet [33]. CommonsenseQA is also adopted by Banerjee et al. [87] for evaluation.
- **STORYCS** [182] consists of 5-sentence stories with annotated motivations and emotional responses. It is originally for emotion classification, where the labels are drawn from classical theories of psychology. Bosselut et al. [89] transform the classification task into a QA task by posing an individual question for each emotion label, and use it for evaluating their KG-aware method for both zero-shot and few-shot settings.
- **aNLI** [183], **QASC** [184], **OpenBookQA** [185] and **ARC** [186] are adopted by Banerjee et al. [87] for evaluating their KG-augmented triple learning model for zero-shot and few-shot QA, besides SocialIQa and CommonsenseQA. Specifically, aNLI which has 171K question-answer pairs is a dataset with commonsese knowledge, while QASC, OpenBookQA and ARC, whose sample sizes range from 6K to 10K, are three QA datasets with scientific knowledge. **OpenBookQA** and **ARC** are also adopted by Zhou et al. [79] for zero-shot QA.

C. Knowledge Graph Completion

KG completion is to infer missing knowledge in a KG. Most existing studies aim at predicting relational facts (triples), which is sometimes called link prediction. In this paper we mainly refer to these link prediction studies. In a zero-shot or few-shot setting, we are required to handle entities and/or relations that emerge after the KG embeddings have been learned. Since the solutions to addressing unseen entities and unseen relations are quite different, we introduce the studies for unseen entities and for unseen relations separately.

1) KG Completion with Unseen Entities: There have been quite a few methods on KG completion with unseen entities. They often utilize entities' auxiliary information such as name information, textual descriptions and attributes, following the mapping-based paradigm [94, 95] and the class feature paradigm [97, 101, 102, 48, 98, 103, 104, 105, 100]. Various benchmarks have been proposed. They are usually constructed based on some existing normal KG completion benchmarks such as FB15k [172], FB15k-237 [187], WordNet11 [188], WN18RR [117] and NELL-995 [189], and some sub-KGs extracted from original KGs such as DBpedia [68] and Wikidata [34]. Their entity auxiliary information is often collected from the benchmarks' original KGs or some associated public resources. For example, the textual descriptions of entities in DBpedia50k, FB20k and Wikidata5M can be collected from DBpedia, Freebase and Wikipedia, respectively; while the textual descriptions of entities of FB15k-237 in [190] are extracted from the introduction section of their corresponding Wikipedia pages.

These benchmarks are often constructed following a common way. Given an original KG completion benchmark, a set of entities are first selected as unseen entities. Then their associated triples in the training set are removed. Next, the relations that appear in both the training set and the testing set are adopted, and the triples of the not adopted relations are removed in both the training set and the testing set. For a testing triple to predict, there could be two cases: *(i)* its head (or tail) is an unseen entity while the other is a seen entity, and *(ii)* both its head and tail are unseen entities. Accordingly, we regard the benchmark with the first case testing triples *semi-ZS*, the benchmark with the second case testing triples *fully-ZS*, and the benchmark with both testing triples as *mixture-ZS*. Here are some typical benchmarks:

- **FB15k-237-OWE** [94] is a typical *semi-ZS* benchmark built on FB15k-237. First, testing triples whose tail entities are to be predicted are collected. Specifically, a set of tail entities are selected, and some associated head entities are randomly picked from the FB15k-237 triples (by uniform sampling over all the associated head entities). Each picked head entity is removed from the training graph by moving all the triples whose heads are this entity to the testing set, and removing all the training triples whose tails are this entity. Testing triples whose head entities are to be predicted are processed in the same way. Then a testing set is generated by merging the above two kinds of testing triples and removing the testing triples whose relations are not in the training set. This testing set is further splitted into a validation set and a final testing set. The dataset contains $2,081$ unseen entities, $12,324$ seen entities and 235 relations. The numbers of triples for training, validation and testing are $242,489$, $10,963$ and $36,250$, respectively.
- **DBpedia50k** and **DBpedia500k** [98] are also typical *semi-ZS* benchmarks, constructed in a similar way as FB15k-237-OWE. DBpedia50k has $49,900$ entities and 654 relations, with $32,388$, 399 and $10,969$ training, validation and testing triples, respectively. DBpedia500k

has $517,475$ entities and 654 relations, with $3,102,677$, $10,000$ and $1,155,937$ training, validation and testing triples.

- **Wikidata5M** [102], originally developed for evaluating text-aware KG embedding methods, is an important *fully-ZS* benchmark. It is constructed based on the Wikidata dump and the English Wikipedia dump. Each entity in Wikidata is aligned to a Wikipedia page and this page's first section is extracted as the entity's textual description. Entities with no Wikipedia pages or with descriptions being shorter than 5 words are discarded. Next, all the relational facts (triples) are extracted from the Wikidata dump. One triple is kept if both of its entities are not discarded, and its relation has a corresponding nonempty page in Wikipedia. Otherwise, this triple is discarded. The benchmark contains $4,594,485$ entities, 822 relations and $20,624,575$ triplets. To support the zero-shot setting, Wang et al. [102] randomly extract two sub-KGs as the validation set and the testing set, and use the remaining as the training set. The three sets, respectively, have $4,579,609$, $7,374$ and $7,475$ entities, 822, 199 and 201 relations, and $20,496,514$, $6,699$ and $6,894$ triples.
- **FB20k** [191] is a benchmark whose testing triples may involve unseen entities. It has the same training set and validation set as the normal KG completion benchmark FB15k, but extends FB15k's testing set by adding triples involving unseen entities. Specifically, a candidate set of unseen entities are first selected from Freebase. They should be associated with some entities in FB15k entities within one hop. Then some new triples whose relations are ensured to be already in FB15k are extracted from Freebase and added to the testing set. These new testing triples have four kinds: those whose head and tail are both seen entities, those whose heads are unseen and whose tails are seen, those whose tails are unseen and whose heads are seen, and those whose heads and tails are both unseen. The first kind of testing triples are for normal KG completion, while the other three kinds are for zero-shot KG completion. So the task of FB20k can be understood as generalized zero-shot KG completion. The numbers of the test triples of the above four types are $57,803$, $18,753$, $11,586$, and 151, respectively, and all these triples involve $19,923$ entities. The subsets of FB15k-237 and WN18RR proposed in [190] are similar.

In few-shot KG completion, unseen entities usually have a small number of associated triples given. The current methods often aim to fully utilize these triples, mainly following the propagation-based paradigm [125, 123, 121, 124, 122, 126, 127], the transfer-based paradigm [135, 136, 137, 138] and the optimization-based paradigm [132]. Several few-shot KG completion benchmarks with unseen entities have also been constructed based on normal KG completion benchmarks.

According to the type of the entity that an unseen entity is linked to, we categorize these benchmarks into three categories. For the first category, the entity linked to is seen in training. Thus the few-shot triples can be utilized to propagate embeddings from seen entities to unseen entities by e.g., GNNs [121, 122, 124]. Typical benchmarks of this category include subsets extracted from WordNet11 by [121], subsets extracted from FB15k by [122], and subsets extracted from WN18RR, FB15k-237 and NELL-995 by [124]. For the second category, the entity linked to is also an unseen entity. These benchmarks are to evaluate the generalization ability of a model trained on one KG to another KG or an emerging sub-KG with different entities. Methods of the transfer-based paradigm are often be adopted. Typical benchmarks of this category include subsets of WN18RR, FB15k-237 and NELL-995 extracted by [135]. In the third category, the entity linked to can be either unseen or seen. Typical benchmarks include subsets of WN18RR, FB15k-237 and NELL-995 contributed in [132] where a meta learning method is often applied to learn embeddings of the unseen entities from their few-shot triples. Next, we will introduce more details of some representative benchmarks of each category:

- **Subsets of WordNet11 by Hamaguchi et al.** [121] are of the first category. They are constructed from the normal KG completion benchmark WordNet11 in the following way. First, entities in the original testing set are extracted as unseen entities, while all the other entities are regarded as seen. Among the unseen entities, those that are associated to only seen entities in the original training triples are kept and the others are discarded. Second, the original training triples that do not contain any unseen entities are selected for a new training set, those that contain exactly one unseen entity are selected as few-shot samples, and those that contain two unseen entities are discarded. Next, a new testing set is constructed from the original testing triples by removing those containing no unseen entities. Nine subsets of different scales are extracted for the few-shot setting, by setting the size of testing triples for extracting unseen entities to $1,000$, $3,000$ and $5,000$, and by setting the position for extracting unseen entities to head, tail and both.
- **Subsets of WN18RR, FB15k-237 and NELL-995 by Teru et al.** [135] are of the second category. They are constructed in the following way. Given one original benchmark, two disjoint graphs are sampled: *train-graph* for training and *ind-test-graph* for testing. It is ensured that their entity sets are disjoint, while the relations of *ind-test-graph* are all involved in *train-graph*. In particular, 10% of the triples of the *ind-test-graph* are randomly selected for testing. These benchmarks are also adopted for evaluation in [137].
- **Subsets of WN18RR, FB15k-237 and NELL-995 by Baek et al.** [132] are of the third category. These subsets are extracted from each original benchmark as follows. First, a set of entities, which have a relatively small amount of associated triples, are randomly sampled as the unseen entities, and they are further partitioned and used for constructing three meta sets of triples: a meta-training set, a meta-validation set and a meta-testing sets. The other entities in the original benchmark are regarded as seen entities. Second, triples composed of seen entities alone are extracted to construct a graph named *In-Graph*.

Finally, the meta sets are cleaned, such that each of their triples has at least one unseen entity and all the triples are out of *In-Graph*.

2) KG Completion with Unseen Relations: Zero-shot KG completion with unseen relations usually utilize the relations' auxiliary information such as their names and descriptions, mainly following the data augmentation paradigm [14, 31, 91] and the class feature paradigm [104, 105, 100]; while few-shot KG completion with unseen relations usually relies on the few-shot triples using methods of the optimization-based paradigm [130, 131, 134, 133], the mapping-based paradigm [119, 118] and the propagation-based paradigm [125]. In comparison with KG completion with unseen entities, there are fewer benchmarks for KG completion with unseen relations. We find NELL-ZS and Wiki-ZS for the zero-shot setting, and NELL-One and Wiki-One for the few-shot setting. NELL-ZS and NELL-One are sub-KGs extracted from NELL, while Wiki-ZS and Wiki-One are sub-KGs extracted from Wikidata. Their details are introduced as follows:

- **NELL-ZS** and **Wiki-ZS** [31] both have a training set with triples of seen relations, a validation set and a testing set with triples of unseen relations. The entities in the testing triples and the validation triples have all been involved in some training triples. NELL-ZS has 139, 10 and 32 training, validation and testing relations, respectively, and $65,567$ entities; while Wiki-ZS has 469, 20, 48 training, validation and testing relations, respectively, and $605,812$ entities. For both NELL-ZS and Wiki-ZS, Qin et al. [31] use relation textual descriptions as the auxiliary information, while Geng et al. [14, 91] construct ontological schemas, which contain not only textual information but also relation hierarchies, relation domains and ranges, relation characteristics and so on.
- **NELL-One** and **Wiki-One** are originally developed by Xiong et al. [118] for evaluating one-shot KG completion with unseen relations. In construction, relations that are associated with less than 500 triples but more than 50 are extracted from the original KGs as task relations (i.e., one relation corresponds to one task). In NELL-One, 67 such relations are extracted and they are partitioned into 51, 5 and 11 for constructing triples of the training, validation and testing set, respectively; while in Wiki-One, 183 such relations are extracted and partitioned into 133, 16 and 34 for constructing triples of the training, validation and testing set, respectively. $68,545$ entities are extracted for NELL-One, and $4,838,244$ entities are extracted for Wiki-One. In addition, another 291 and 639 relations are extracted, respectively, as background relations constructing more triples for the entities. Note that these two benchmarks can also be simply revised for more general few-shot KG completion by adding more than one triples.

VII. Open Problems

A. Knowledge Graph Quality

For a ZSL or FSL task, one critical challenge is constructing a customized KG with exactly necessary and high-quality knowledge. Although we now can re-use existing KGs, extract knowledge from some task data, and curate knowledge with domain experts, some open problems still remain.

First, knowledge and data integration is rarely investigated, and the impact of low-quality knowledge, which could be biased or even erroneous, has not been studied in the current KG-aware ZSL and FSL works. Second, the coverage of necessary knowledge and the ratio of irrelevant knowledge are often ignored in investigating a KG's usefulness towards a ZSL or FSL task, and there is a shortage of methods that are able to (semi-)automatically retrieve relevant knowledge from a large scale KG for a given task. Third, several popular knowledge sources, such as natural language text, Web tables and databases, and their corresponding knowledge extraction methods like open information extraction, have been rarely explored for constructing the auxiliary KG for ZSL or FSL. This could be a promising direction to improve the coverage and quality of the auxiliary KG. Fourth crowdsourcing and human-on-the-loop techniques for data curation and KG construction (e.g., [192]) are also worth investigating as another potential way for further augmenting ZSL and FSL.

B. Learning Paradigms

1) KG-aware ZSL: The mapping-based paradigm has been widely investigated, while the data augmentation paradigm has only four methods, one of which uses rules while the remaining three of which use generation models. Mapping-based methods are often biased to unseen classes in prediction, while the generation-based paradigm can flexible choose the model and avoid the bias after data are generated. Thus we think generation-based methods conditioned on KG embeddings is worth of more investigation in the future. Meanwhile, it is hard to use rules to generate numeric samples or features, but in KG completion, it is feasible to use ontological schemas and logical rules to infer triples for the unseen entities and/or relations. It would be a promising direction to combine symbolic reasoning with data augmentation.

Regarding the propagation-based paradigm, belief propagation has not been widely investigated, but would be a good solution for some CV tasks such as scene graph extraction and VQA, where multiple objects in an image or video, as well as their semantic relationships, need to be recognized. With the development of pre-trained language models, the class feature paradigm is becoming more and more popular, especially for tasks whose inputs are text, such as text classification, question answering and knowledge extraction. We think this trend will continue in the future, while KGs will still play an important role by providing symbolic knowledge that these parameter-based language models cannot represent.

2) KG-aware FSL: Many FSL methods focus on utilizing the few-shot samples via applying meta learning algorithms or extending the ZSL methods. It is still challenging to combine the KG auxiliary information and the few-shot samples. For the data augmentation paradigm, how to merge the generated samples and the few-shot samples? For the optimization-based paradigm, how to use KG to guide the meta learning algorithm? For the transfer-based paradigm, how to guide the

model transfer with KG? For embedding propagation for KG completion, how to augment the propagation models such as GNN with auxiliary information especially the ontological schema? We think all these are still open problems and worth further investigation in the future.

C. Zero-shot and Few-shot Learning in KG Construction

Nowadays KG construction uses not only heuristics (e.g., hand-craft rules and templates), symbolic knowledge engineering and manual curation, but also machine learning prediction for (semi-)automation [193]. Prediction tasks range from knowledge extraction from different data sources such as text, tables and Web pages, to knowledge curation such as KG completion, entity alignment, entity resolution, entity typing and schema inference. Many such tasks rely on supervised learning, but often suffer from the shortage of labeled samples. Although some tasks such as entity linking and KG completion have been widely investigated in FSL and ZSL, developing robust ZSL and FSL methods for these prediction tasks under a KG context is still an open problem and should attract wider attention. Meanwhile, some KG construction and curation tasks, such as entity typing and table to KG matching, some dynamic KG contexts with e.g., involving schema and/or data, and some complex knowledge representations such as OWL ontology and Datalog rule, can be considered for new benchmarks for KG-aware ZSL and FSL.

D. Benchmarking

Although there have been some benchmarking studies for ZSL and FSL [27, 194], systematic evaluation and comparison of KG-aware methods is still not enough. The existing KG-aware ZSL and FSL benchmarks are usually associated with fixed KGs. The current methodology studies do not consider the impact of different settings on knowledge coverage, representation and quality settings, and rarely apply one method to different tasks, which cannot show the generalization capability. Our recent benchmarking work [91] has analyzed the impact of different KG semantics such as textual information, attributes and RDFS schemas on two typical and representative ZSL methods OntoZSL [14] and DeViSE [10] for three tasks (see results in the original paper), but fare and comprehensive comparison of more methods across different datasets, different knowledge settings and different tasks are urgently needed in the future. Meanwhile, more benchmarks should be developed to cover more domains where KGs are widely used such as health science domains.

VIII. Conclusion

KGs have become popular auxiliary information for augmenting ZSL and FSL, and at the same time KG construction also involve many prediction tasks with zero-shot and few-shot settings. Thus KG-aware ZSL and FSL have gained widespread attention and popularity in many domains such as CV, NLP, ML and the semantic Web. In this survey, we systematically review over 90 KG-aware studies for addressing sample shortage in ML from perspectives of the KG, the methodology and the application. The content covers *(i)* the introduction of KGs that have been applied and the methods for constructing such task-specific KGs, *(ii)* the review of the KG-aware ZSL and FSL methods of each paradigm, and *(iii)* the presentation of the development of ZSL and FSL research for different tasks in CV, NLP and KG completion, as well as the resources that can be used for evaluating KG-aware ZSL and FSL methods. Besides, we have also analyzed and discussed the challenges of KG-aware zero-shot and few-shot learning, and some potential future directions.

Acknowledgments

This work is supported by Samsung Research UK, the EPSRC projects ConCur (EP/V050869/1), UK FIRES (EP/S019111/1) and OASIS (EP/S032347/1), NSFCU19B2027/91846204, Zhejiang Provincial Natural Science Foundation of China (No. Q23F020051), the joint project DH-2022ZY0012 from Donghai Lab and the Chang Jiang Scholars Program (J2019032).

References

[1] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell, "Zero-shot learning with semantic output codes," *Advances in Neural Information Processing Systems*, vol. 22, 2009.

[2] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 951–958.

[3] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 1778–1785.

[4] M. Fink, "Object classification from a single example utilizing class relevance metrics," *Advances in Neural Information Processing Systems*, vol. 17, pp. 449–456, 2005.

[5] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 594–611, 2006.

[6] Y. Fu, T. Xiang, Y.-G. Jiang, X. Xue, L. Sigal, and S. Gong, "Recent advances in zero-shot recognition: Toward data-efficient understanding of visual content," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 112–125, 2018.

[7] W. Wang, V. W. Zheng, H. Yu, and C. Miao, "A survey of zero-shot learning: Settings, methods, and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–37, 2019.

[8] J. Chen, Y. Geng, Z. Chen, I. Horrocks, J. Z. Pan, and H. Chen, "Knowledge-aware zero-shot learning: Survey and perspective," in *IJCAI Survey Track*, 2021.

[9] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[10] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov, "Devise: A deep visual-semantic embedding model," *Advances in Neural Information Processing Systems*, vol. 26, pp. 2121–2129, 2013.

[11] R. Qiao, L. Liu, C. Shen, and A. Van Den Hengel, "Less is more: zero-shot learning from online textual documents with noise suppression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2249–2257.

[12] X. Wang, Y. Ye, and A. Gupta, "Zero-shot recognition via semantic embeddings and knowledge graphs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6857–6866.

[13] M. Kampffmeyer, Y. Chen, X. Liang, H. Wang, Y. Zhang, and E. P. Xing, "Rethinking knowledge graph propagation for zero-shot learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 487–11 496.

[14] Y. Geng, J. Chen, Z. Chen, J. Z. Pan, Z. Ye, Z. Yuan, Y. Jia, and H. Chen, "OntoZSL: Ontology-enhanced zero-shot learning," in *Proceedings of the Web Conference*, 2021, pp. 3325–3336.

[15] Z. Chen, Y. Huang, J. Chen, Y. Geng, W. Zhang, Y. Fang, J. Z. Pan, W. Song, and H. Chen, "Duet: Cross-modal semantic grounding for contrastive zero-shot learning," in *AAAI*, 2023.

[16] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Computing Surveys (CSUR)*, vol. 53, no. 3, pp. 1–34, 2020.

[17] C. Lemke, M. Budka, and B. Gabrys, "Metalearning: A survey of trends and technologies," *Artificial Intelligence Review*, vol. 44, no. 1, pp. 117–130, 2015.

[18] W. Yin, "Meta-learning for few-shot natural language processing: A survey," *arXiv preprint arXiv:2007.09604*, 2020.

[19] D. Sui, Y. Chen, B. Mao, D. Qiu, K. Liu, and J. Zhao, "Knowledge guided metric learning for few-shot text classification," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021, pp. 3266–3271.

[20] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel *et al.*, "Never-ending learning," *Communications of the ACM*, vol. 61, no. 5, pp. 103–115, 2018.

[21] Y.-H. H. Tsai and R. Salakhutdinov, "Improving one-shot learning through fusing side information," *arXiv preprint arXiv:1710.08347*, 2017.

[22] J. Chen, F. Lécué, J. Z. Pan, I. Horrocks, and H. Chen, "Knowledge-based transfer learning explanation," in *Sixteenth International Conference on Principles of Knowledge Representation and Reasoning*, 2018.

[23] Z. Peng, Z. Li, J. Zhang, Y. Li, G.-J. Qi, and J. Tang, "Few-shot image recognition with knowledge transfer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 441–449.

[24] N. Zhang, S. Deng, Z. Sun, J. Chen, W. Zhang, and H. Chen, "Relation adversarial network for low resource knowledge graph completion," in *Proceedings of The Web Conference 2020*, 2020, pp. 1–12.

[25] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. d. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier *et al.*, "Knowledge graphs," *arXiv preprint arXiv:2003.02320*, 2020.

[26] Y. Hu, A. Chapman, G. Wen, and D. W. Hall, "What can knowledge bring to machine learning?–a survey of low-shot learning for structured data," *arXiv preprint arXiv:2106.06410*, 2021.

[27] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning — a comprehensive evaluation of the good, the bad and the ugly," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 9, pp. 2251–2265, 2018.

[28] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 453–465, 2013.

[29] D. Parikh and K. Grauman, "Relative attributes," in *2011 International Conference on Computer Vision*. IEEE, 2011, pp. 503–510.

[30] M. Elhoseiny, B. Saleh, and A. Elgammal, "Write a classifier: Zero-shot learning using purely textual descriptions," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2584–2591.

[31] P. Qin, X. Wang, W. Chen, C. Zhang, W. Xu, and W. Y. Wang, "Generative adversarial zero-shot relational learning for knowledge graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 8673–8680.

[32] G. A. Miller, "WordNet: A lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[33] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[34] D. Vrandečić and M. Krötzsch, "Wikidata: A free collaborative knowledgebase," *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014.

[35] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *Siam Review*, vol. 60, no. 2, pp. 223–311, 2018.

[36] J. Z. Pan, G. Vetere, J. Gomez-Perez, and H. Wu, Eds., *Exploiting Linked Data and Knowledge Graphs for Large Organisations*. Springer, 2017.

[37] Y. Nenov, R. Piro, B. Motik, I. Horrocks, Z. Wu, and J. Banerjee, "Rdfox: A highly-scalable rdf store," in *International Semantic Web Conference*. Springer, 2015, pp. 3–20.

[38] I. Horrocks, "Ontologies and the semantic web," *Communications of the ACM*, vol. 51, no. 12, pp. 58–67, 2008.

[39] S. Schulz and R. Cornet, "SNOMED CT's ontological

commitment," *Nature Precedings*, pp. 1–1, 2009.

[40] J. Z. Pan, G. Stamou, V. Tzouvaras, and I. Horrocks, "f-SWRL: A fuzzy extension of SWRL," in *Proc. of the International Conference on Artificial Neural Networks (ICANN 2005), Special section on "Intelligent multimedia and semantics"*, 2005.

[41] I. Horrocks and P. F. Patel-Schneider, "A proposal for an OWL rules language," in *Proceedings of the 13th international conference on World Wide Web*, 2004, pp. 723–731.

[42] B. Liu, L. Yao, Z. Ding, J. Xu, and J. Wu, "Combining ontology and reinforcement learning for zero-shot classification," *Knowledge-Based Systems*, vol. 144, pp. 42–50, 2018.

[43] C.-W. Lee, W. Fang, C.-K. Yeh, and Y.-C. F. Wang, "Multi-label zero-shot learning with structured knowledge graphs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1576–1585.

[44] J. Wei, Y. Yang, J. Li, L. Zhu, L. Zuo, and H. T. Shen, "Residual graph convolutional networks for zero-shot learning," in *Proceedings of the ACM Multimedia Asia*, 2019, pp. 1–6.

[45] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 7, pp. 1425–1438, 2015.

[46] X. Li, S. Liao, W. Lan, X. Du, and G. Yang, "Zero-shot image tagging by hierarchical semantic embedding," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2015, pp. 879–882.

[47] Y. Geng, J. Chen, Z. Ye, Z. Yuan, W. Zhang, and H. Chen, "Explainable zero-shot learning via attentive graph convolutional network and knowledge graphs," *Semantic Web*, no. Preprint, pp. 1–28, 2020.

[48] E. Amador-Domínguez, E. Serrano, D. Manrique, P. Hohenecker, and T. Lukasiewicz, "An ontology-based deep learning approach for triple classification with out-of-knowledge-base entities," *Information Sciences*, vol. 564, pp. 85–102, 2021.

[49] J. Wang and B. Jiang, "Zero-shot learning via contrastive learning on dual knowledge graphs," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 885–892.

[50] J. Chen, L. Pan, Z. Wei, X. Wang, C.-W. Ngo, and T.-S. Chua, "Zero-shot ingredient recognition by multi-relational graph convolutional network," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 10 542–10 550.

[51] R. Chen, T. Chen, X. Hui, H. Wu, G. Li, and L. Lin, "Knowledge graph transfer network for few-shot recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 10 575–10 582.

[52] M. Jayathilaka, T. Mu, and U. Sattler, "Ontology-based n-ball concept embeddings informing few-shot image classification," *arXiv preprint arXiv:2109.09063*, 2021.

[53] S. Monka, L. Halilaj, S. Schmid, and A. Rettinger, "Learning visual models using a knowledge graph as a trainer," in *International Semantic Web Conference*. Springer, 2021, pp. 357–373.

[54] N. V. Nayak and S. H. Bach, "Zero-shot learning with common sense knowledge graphs," *arXiv preprint arXiv:2006.10713*, 2020.

[55] C. Zhang, X. Lyu, and Z. Tang, "Tgg: Transferable graph generation for zero-shot and few-shot learning," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1641–1649.

[56] A. Roy, D. Ghosal, E. Cambria, N. Majumder, R. Mihalcea, and S. Poria, "Improving zero shot learning baselines with commonsense knowledge," *arXiv preprint arXiv:2012.06236*, 2020.

[57] Z. Chen, J. Chen, Y. Geng, J. Z. Pan, Z. Yuan, and H. Chen, "Zero-shot visual question answering using knowledge graph," in *International Semantic Web Conference*, 2021.

[58] H.-V. Nguyen, F. Gelli, and S. Poria, "DOZEN: Cross-domain zero shot named entity recognition with knowledge graph," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 1642–1646.

[59] J. Zhang, P. Lertvittayakumjorn, and Y. Guo, "Integrating semantic knowledge to tackle zero-shot text classification," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 1031–1040.

[60] Q. Chen, W. Wang, K. Huang, and F. Coenen, "Zero-shot text classification via knowledge graph embedding for social media data," *IEEE Internet of Things Journal*, 2021.

[61] Z. Yang, Z. Gan, J. Wang, X. Hu, Y. Lu, Z. Liu, and L. Wang, "An empirical study of gpt-3 for few-shot knowledge-based vqa," *arXiv preprint arXiv:2109.05014*, 2021.

[62] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, 2008, pp. 1247–1250.

[63] Y. Ma, E. Cambria, and S. Gao, "Label embedding for zero-shot fine-grained named entity typing," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 171–180.

[64] W. Imrattanatrai, M. P. Kato, and M. Yoshikawa, "Identifying entity properties from text with zero-shot learning," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 195–204.

[65] J. Zhang, J. Zhu, Y. Yang, W. Shi, C. Zhang, and H. Wang, "Knowledge-enhanced domain adaptation in few-shot relation classification," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discov-*

ery & Data Mining, 2021, pp. 2183–2191.

[66] M. Qu, T. Gao, L.-P. Xhonneux, and J. Tang, "Few-shot relation extraction via bayesian meta-learning on relation graphs," in *International Conference on Machine Learning*. PMLR, 2020, pp. 7867–7876.

[67] J. Li, R. Wang, N. Zhang, W. Zhang, F. Yang, and H. Chen, "Logic-guided semantic representation learning for zero-shot relation classification," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 2967–2978.

[68] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A nucleus for a web of open data," in *The semantic web*. Springer, 2007, pp. 722–735.

[69] X. Chen, A. Shrivastava, and A. Gupta, "NEIL: Extracting visual knowledge from web data," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1409–1416.

[70] N. Tandon, G. De Melo, F. Suchanek, and G. Weikum, "Webchild: Harvesting and organizing commonsense knowledge from the web," in *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, 2014, pp. 523–532.

[71] P. Wang, Q. Wu, C. Shen, A. Dick, and A. Van Den Hengel, "FVQA: Fact-based visual question answering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 10, pp. 2413–2427, 2017.

[72] M. Cafarella, A. Halevy, H. Lee, J. Madhavan, C. Yu, D. Z. Wang, and E. Wu, "Ten years of webtables," *Proceedings of the VLDB Endowment*, vol. 11, no. 12, pp. 2140–2149, 2018.

[73] L. Huang, H. Ji, K. Cho, I. Dagan, S. Riedel, and C. Voss, "Zero-shot transfer learning for event extraction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 2160–2170.

[74] C. F. Baker and H. Sato, "The framenet data and software," in *The Companion Volume to the Proceedings of 41st Annual Meeting of the Association for Computational Linguistics*, 2003, pp. 161–164.

[75] A. T. McCray, "An upper-level ontology for the biomedical domain," *Comparative and Functional genomics*, vol. 4, no. 1, pp. 80–84, 2003.

[76] A. Rios and R. Kavuluru, "Few-shot and zero-shot multi-label learning for structured label spaces," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 3132–3142.

[77] R. Luo, N. Zhang, B. Han, and L. Yang, "Context-aware zero-shot recognition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 11 709–11 716.

[78] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L. Li, D. Shamma *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International Journal of Computer Vision*, pp. 1–42, 2017.

[79] Z. Zhou, M. Valentino, D. Landers, and A. Freitas, "Encoding explanatory knowledge for zero-shot science question answering," *arXiv preprint arXiv:2105.05737*, 2021.

[80] Z. Xie, S. Thiem, J. Martin, E. Wainwright, S. Marmorstein, and P. Jansen, "Worldtree v2: A corpus of science-domain structured explanations and inference patterns supporting multi-hop inference," in *Proceedings of the 12th Language Resources and Evaluation Conference*, 2020, pp. 5456–5473.

[81] P. Ghosh, N. Saini, L. S. Davis, and A. Shrivastava, "All about knowledge graphs for actions," *arXiv preprint arXiv:2008.12432*, 2020.

[82] Y. Hu, G. Wen, A. Chapman, P. Yang, M. Luo, Y. Xu, D. Dai, and W. Hall, "Graph-based visual-semantic entanglement network for zero-shot image recognition," *IEEE Transactions on Multimedia*, 2021.

[83] S. Changpinyo, W.-L. Chao, B. Gong, and F. Sha, "Synthesized classifiers for zero-shot learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*, 2016, pp. 5327–5336.

[84] J. Zhao, X. Lin, J. Zhou, J. Yang, L. He, and Z. Yang, "Knowledge-based fine-grained classification for few-shot learning," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2020, pp. 1–6.

[85] A. Li, Z. Lu, J. Guan, T. Xiang, L. Wang, and J.-R. Wen, "Transferrable feature and projection learning with class hierarchy for zero-shot learning," *International Journal of Computer Vision*, vol. 128, pp. 2810–2827, 2020.

[86] A. Li, T. Luo, Z. Lu, T. Xiang, and L. Wang, "Large-scale few-shot learning: Knowledge transfer with class hierarchy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7212–7220.

[87] P. Banerjee and C. Baral, "Self-supervised knowledge triplet learning for zero-shot question answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 151–162.

[88] J. Chen, F. Lécué, Y. Geng, J. Z. Pan, and H. Chen, "Ontology-guided semantic composition for zero-shot learning," in *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning*, vol. 17, no. 1, 2020, pp. 850–854.

[89] A. Bosselut, R. L. Bras, and Y. Choi, "Dynamic knowledge graph construction for zero-shot commonsense question answering," in *The Thirty-Fifth AAAI Conference on Artificial Intelligence*, 2021.

[90] A. Bosselut, H. Rashkin, M. Sap, C. Malaviya, A. Celikyilmaz, and Y. Choi, "Comet: Commonsense transformers for automatic knowledge graph construction," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 4762–4779.

[91] Y. Geng, J. Chen, X. Zhuang, Z. Chen, J. Z. Pan, J. Li, Z. Yuan, and H. Chen, "Benchmarking knowledge-driven zero-shot learning," *Journal of Web Semantics*, vol. 75, p. 100757, 2023.

[92] Y. He, J. Chen, D. Antonyrajah, and I. Horrocks, "BERTMap: A BERT-based ontology alignment system," *arXiv preprint arXiv:2112.02682*, 2021.

[93] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for attribute-based classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 819–826.

[94] H. Shah, J. Villmow, A. Ulges, U. Schwanecke, and F. Shafait, "An open-world extension to knowledge graph completion models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3044–3051.

[95] Y. Hao, X. Cao, Y. Fang, X. Xie, and S. Wang, "Inductive link prediction for nodes having only attribute information," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 2020.

[96] T. Rocktäschel, S. Singh, and S. Riedel, "Injecting logical background knowledge into embeddings for relation extraction," in *Proceedings of the 2015 conference of the north American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 1119–1129.

[97] Y. Zhao, S. Gao, P. Gallinari, and J. Guo, "Zero-shot embedding for unseen entities in knowledge graph," *IEICE Transactions on Information and Systems*, vol. 100, no. 7, pp. 1440–1447, 2017.

[98] B. Shi and T. Weninger, "Open-world knowledge graph completion," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[99] L. Logeswaran, M.-W. Chang, K. Lee, K. Toutanova, J. Devlin, and H. Lee, "Zero-shot entity linking by reading entity descriptions," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 3449–3460.

[100] L. Yao, C. Mao, and Y. Luo, "Kg-bert: Bert for knowledge graph completion," *arXiv preprint arXiv:1909.03193*, 2019.

[101] L. Niu, C. Fu, Q. Yang, Z. Li, Z. Chen, Q. Liu, and K. Zheng, "Open-world knowledge graph completion with multiple interaction attention," *World Wide Web*, vol. 24, no. 1, pp. 419–439, 2021.

[102] X. Wang, T. Gao, Z. Zhu, Z. Zhang, Z. Liu, J. Li, and J. Tang, "KEPLER: A unified model for knowledge embedding and pre-trained language representation," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 176–194, 2021.

[103] B. Wang, G. Wang, J. Huang, J. You, J. Leskovec, and C.-C. J. Kuo, "Inductive learning on commonsense knowledge graph completion," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[104] H. Zha, Z. Chen, and X. Yan, "Inductive relation prediction by bert," *arXiv preprint arXiv:2103.07102*, 2021.

[105] B. Wang, T. Shen, G. Long, T. Zhou, Y. Wang, and Y. Chang, "Structure-augmented text representation learning for efficient knowledge graph completion," in *Proceedings of the Web Conference 2021*, 2021, pp. 1737–1748.

[106] J. Gong and H. Eldardiry, "Prompt-based zero-shot relation classification with semantic knowledge augmentation," *arXiv preprint arXiv:2112.04539*, 2021.

[107] P. Ristoski, Z. Lin, and Q. Zhou, "Kg-zeshel: Knowledge graph-enhanced zero-shot entity linking," in *Proceedings of the 11th on Knowledge Capture Conference*, 2021, pp. 49–56.

[108] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3174–3183.

[109] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[110] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[111] H. Huang, C. Wang, P. S. Yu, and C.-D. Wang, "Generative dual adversarial network for generalized zero-shot learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 801–810.

[112] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5542–5551.

[113] Y. Zhu, M. Elhoseiny, B. Liu, X. Peng, and A. Elgammal, "A generative adversarial approach for zero-shot learning from noisy texts," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1004–1013.

[114] Y. Geng, J. Chen, W. Zhang, Y. Xu, Z. Chen, J. Z. Pan, Y. Huang, F. Xiong, and H. Chen, "Disentangled ontology embedding for zero-shot learning," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 443–453.

[115] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov, "Fasttext. zip: Compressing text classification models," *arXiv preprint arXiv:1612.03651*, 2016.

[116] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[117] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Thirty-second AAAI Conference on Artificial Intelligence*, 2018.

[118] W. Xiong, M. Yu, S. Chang, X. Guo, and W. Y. Wang, "One-shot relational learning for knowledge graphs," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 1980–1990.

[119] C. Zhang, H. Yao, C. Huang, M. Jiang, Z. Li, and N. V. Chawla, "Few-shot knowledge graph completion," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 03, 2020, pp. 3041–3048.

[120] Z. Wang, K. Lai, P. Li, L. Bing, and W. Lam, "Tackling long-tailed relations and uncommon entities in knowledge graph completion," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 250–260.

[121] T. Hamaguchi, H. Oiwa, M. Shimbo, and Y. Matsumoto, "Knowledge transfer for out-of-knowledge-base entities: a graph neural network approach," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017, pp. 1802–1808.

[122] P. Wang, J. Han, C. Li, and R. Pan, "Logic attention based neighborhood aggregation for inductive knowledge graph embedding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 7152–7159.

[123] M. Albooyeh, R. Goel, and S. M. Kazemi, "Out-of-sample representation learning for knowledge graphs," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, 2020, pp. 2657–2666.

[124] R. Bhowmik and G. de Melo, "Explainable link prediction for emerging entities in knowledge graphs," in *International Semantic Web Conference*. Springer, 2020, pp. 39–55.

[125] M. Zhao, W. Jia, and Y. Huang, "Attention-based aggregation graph networks for knowledge graph information transfer," *Advances in Knowledge Discovery and Data Mining*, vol. 12085, p. 542.

[126] D. Dai, H. Zheng, F. Luo, P. Yang, B. Chang, and Z. Sui, "Inductively representing out-of-knowledge-graph entities by optimal estimation under translational assumptions," *arXiv preprint arXiv:2009.12765*, 2020.

[127] M. Ali, M. Berrendorf, M. Galkin, V. Thost, T. Ma, V. Tresp, and J. Lehmann, "Improving inductive link prediction using hyper-relational facts," in *International Semantic Web Conference*. Springer, 2021, pp. 74–92.

[128] N. Zhang, S. Deng, Z. Sun, G. Wang, X. Chen, W. Zhang, and H. Chen, "Long-tail relation extraction via knowledge graph embeddings and graph convolution networks," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 3016–3025.

[129] K. Marino, X. Chen, D. Parikh, A. Gupta, and M. Rohrbach, "Krisp: Integrating implicit and symbolic knowledge for open-domain knowledge-based vqa," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 111–14 121.

[130] H. Wang, W. Xiong, M. Yu, X. Guo, S. Chang, and W. Y. Wang, "Meta reasoning over knowledge graphs," *arXiv preprint arXiv:1908.04877*, 2019.

[131] M. Chen, W. Zhang, W. Zhang, Q. Chen, and H. Chen, "Meta relational learning for few-shot link prediction in knowledge graphs," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 4217–4226.

[132] J. Baek, D. B. Lee, and S. J. Hwang, "Learning to extrapolate knowledge: Transductive few-shot out-of-graph link prediction," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[133] X. Lv, Y. Gu, X. Han, L. Hou, J. Li, and Z. Liu, "Adapting meta knowledge graph information for multi-hop reasoning over few-shot relations," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3376–3381.

[134] C. Zhang, L. Yu, M. Saebi, M. Jiang, and N. Chawla, "Few-shot multi-hop relation reasoning over knowledge bases," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, 2020, pp. 580–585.

[135] K. Teru, E. Denis, and W. Hamilton, "Inductive relation prediction by subgraph reasoning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 9448–9457.

[136] S. Liu, B. Grau, I. Horrocks, and E. Kostylev, "Indigo: Gnn-based inductive knowledge graph completion using pair-wise encoding," in *Advances in Neural Information Processing Systems*, 2021.

[137] J. Chen, H. He, F. Wu, and J. Wang, "Topology-aware correlations between relations for inductive link prediction in knowledge graphs," in *AAAI*. AAAI Press, 2021, pp. 6271–6278.

[138] A. Sadeghian, M. Armandpour, P. Ding, and D. Z. Wang, "Drum: End-to-end differentiable rule mining on knowledge graphs," *Advances in Neural Information Processing Systems*, vol. 32, pp. 15 347–15 357, 2019.

[139] L. Mihalkova, T. Huynh, and R. J. Mooney, "Mapping and revising markov logic networks for transfer learning," in *Proceedings of the 22nd national conference on Artificial Intelligence*, 2007, pp. 608–614.

[140] L. Mihalkova and R. J. Mooney, "Transfer learning from minimal target data by mapping across relational domains," in *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[141] J. Davis and P. Domingos, "Deep transfer via second-order markov logic," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 217–224.

[142] J. Van Haaren, A. Kolobov, and J. Davis, "Todtler: Two-order-deep transfer learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.

[143] M. Kulmanov, W. Liu-Wei, Y. Yan, and R. Hoehndorf, "El embeddings: Geometric construction of models

for the description logic el++," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2019, pp. 6103–6109.

[144] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Advances in Neural Information Processing Systems*, vol. 28, pp. 3483–3491, 2015.

[145] M. Galkin, P. Trivedi, G. Maheshwari, R. Usbeck, and J. Lehmann, "Message passing for hyper-relational knowledge graphs," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 7346–7359.

[146] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1126–1135.

[147] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *European Semantic Web Conference*. Springer, 2018, pp. 593–607.

[148] L. A. Galárraga, C. Teflioudi, K. Hose, and F. Suchanek, "Amie: Association rule mining under incomplete evidence in ontological knowledge bases," in *Proceedings of the 22nd international conference on World Wide Web*, 2013, pp. 413–422.

[149] A. Kimmig, S. Bach, M. Broecheler, B. Huang, and L. Getoor, "A short introduction to probabilistic soft logic," in *Proceedings of the NIPS Workshop on Probabilistic Programming: Foundations and Applications*, 2012, pp. 1–4.

[150] Z. Li, L. Yao, X. Zhang, X. Wang, S. Kanhere, and H. Zhang, "Zero-shot object detection with textual descriptions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 8690–8697.

[151] Y. Fang, K. Kuan, J. Lin, C. Tan, and V. Chandrasekhar, "Object detection meets knowledge graphs." International Joint Conferences on Artificial Intelligence, 2017.

[152] C. Lang, A. Braun, and A. Valada, "Contrastive object detection using knowledge graph embeddings," *arXiv preprint arXiv:2112.11366*, 2021.

[153] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean, "Zero-shot learning by convex combination of semantic embeddings," in *The International Conference on Learning Representations*, 2014.

[154] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng, "Zero-shot learning through cross-modal transfer," in *Advances in Neural Information Processing Systems*, 2013, pp. 935–943.

[155] S. Reed, Z. Akata, H. Lee, and B. Schiele, "Learning deep representations of fine-grained visual descriptions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 49–58.

[156] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, 2009, pp. 248–255.

[157] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "Nus-wide: a real-world web image database from national university of singapore," in *Proceedings of the ACM international conference on image and video retrieval*, 2009, pp. 1–9.

[158] H. Huang, W. Tang, P. S. Yu, Y. Chen, W. Zheng, and Q. Chen, "Multi-label zero-shot classification by learning to transfer from external knowledge," in *BMVC*. BMVA Press, 2020.

[159] S. Narayan, A. Gupta, S. Khan, F. S. Khan, L. Shao, and M. Shah, "Discriminative region-based multi-label zero-shot learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 8731–8740.

[160] Z. Wu and M. Palmer, "Verb semantics and lexical selection," p. 133–138, 1994.

[161] B. Hariharan and R. Girshick, "Low-shot visual recognition by shrinking and hallucinating features," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3018–3027.

[162] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra *et al.*, "Matching networks for one shot learning," *Advances in Neural Information Processing Systems*, vol. 29, pp. 3630–3638, 2016.

[163] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," 2011.

[164] D. Teney and A. v. d. Hengel, "Zero-shot visual question answering," *arXiv preprint arXiv:1611.05546*, 2016.

[165] S. K. Ramakrishnan, A. Pal, G. Sharma, and A. Mittal, "An empirical evaluation of visual question answering for novel objects," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4392–4401.

[166] K. Marino, M. Rastegari, A. Farhadi, and R. Mottaghi, "OK-VQA: A visual question answering benchmark requiring external knowledge," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3195–3204.

[167] P. Banerjee, T. Gokhale, Y. Yang, and C. Baral, "Weaqa: Weak supervision via captions for visual question answering," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 3420–3435.

[168] D. Teney and A. van den Hengel, "Visual question answering as a meta learning task," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 219–235.

[169] R. Weischedel and A. Brunstein, "BBN pronoun coreference and entity type corpus," *Linguistic Data Consortium, Philadelphia*, vol. 112, 2005.

[170] R. M. Weischedel, E. H. Hovy, M. P. Marcus, and M. Palmer, "OntoNotes: A large training corpus for

enhanced processing," 2017.

[171] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2010, pp. 148–163.

[172] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[173] A. Perotte, R. Pivovarov, K. Natarajan, N. Weiskopf, F. Wood, and N. Elhadad, "Diagnosis code assignment: models and evaluation metrics," *Journal of the American Medical Informatics Association*, vol. 21, no. 2, pp. 231–237, 2014.

[174] A. E. Johnson, T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark, "Mimic-iii, a freely accessible critical care database," *Scientific data*, vol. 3, no. 1, pp. 1–9, 2016.

[175] J. Blitzer, M. Dredze, and F. C. Pereira, "Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification," in *ACL*, 2007.

[176] K. Ma, F. Ilievski, J. Francis, Y. Bisk, E. Nyberg, and A. Oltramari, "Knowledge-driven data construction for zero-shot evaluation in commonsense question answering," in *35th AAAI Conference on Artificial Intelligence*, 2021.

[177] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le, "Finetuned language models are zero-shot learners," *arXiv preprint arXiv:2109.01652*, 2021.

[178] Y. Bisk, R. Zellers, J. Gao, Y. Choi *et al.*, "PIQA: Reasoning about physical commonsense in natural language," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 7432–7439.

[179] M. Sap, H. Rashkin, D. Chen, R. Le Bras, and Y. Choi, "Social IQa: Commonsense reasoning about social interactions," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 4463–4473.

[180] M. Sap, R. Le Bras, E. Allaway, C. Bhagavatula, N. Lourie, H. Rashkin, B. Roof, N. A. Smith, and Y. Choi, "Atomic: An atlas of machine commonsense for if-then reasoning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3027–3035.

[181] A. Talmor, J. Herzig, N. Lourie, and J. Berant, "Commonsenseqa: A question answering challenge targeting commonsense knowledge," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4149–4158.

[182] L. Lucy and J. Gauthier, "Are distributional representations ready for the real world? evaluating word vectors for grounded perceptual meaning," in *Proceedings of the First Workshop on Language Grounding for Robotics*, 2017, pp. 76–85.

[183] C. Bhagavatula, R. Le Bras, C. Malaviya, K. Sakaguchi, A. Holtzman, H. Rashkin, D. Downey, W.-t. Yih, and Y. Choi, "Abductive commonsense reasoning," in *The International Conference on Learning Representations*, 2020.

[184] T. Khot, P. Clark, M. Guerquin, P. Jansen, and A. Sabharwal, "QASC: A dataset for question answering via sentence composition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 8082–8090.

[185] T. Mihaylov, P. Clark, T. Khot, and A. Sabharwal, "Can a suit of armor conduct electricity? a new dataset for open book question answering," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 2381–2391.

[186] S. Bhakthavatsalam, D. Khashabi, T. Khot, B. D. Mishra, K. Richardson, A. Sabharwal, C. Schoenick, O. Tafjord, and P. Clark, "Think you have solved direct-answer question answering? try arc-da, the direct-answer ai2 reasoning challenge," *arXiv preprint arXiv:2102.03315*, 2021.

[187] K. Toutanova, D. Chen, P. Pantel, H. Poon, P. Choudhury, and M. Gamon, "Representing text for joint embedding of text and knowledge bases," in *Proceedings of the 2015 conference on empirical methods in natural language processing*, 2015, pp. 1499–1509.

[188] R. Socher, D. Chen, C. D. Manning, and A. Ng, "Reasoning with neural tensor networks for knowledge base completion," in *Advances in neural information processing systems*, 2013, pp. 926–934.

[189] W. Xiong, T. Hoang, and W. Y. Wang, "DeepPath: A reinforcement learning method for knowledge graph reasoning," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 564–573.

[190] D. Daza, M. Cochez, and P. Groth, "Inductive entity representations from text via link prediction," in *Proceedings of the Web Conference*, 2021, pp. 798–808.

[191] R. Xie, Z. Liu, J. Jia, H. Luan, and M. Sun, "Representation learning of knowledge graphs with entity descriptions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[192] L. Jiang, L. Chen, and Z. Chen, "Knowledge base enhancement via data facts and crowdsourcing," in *2018 IEEE 34th International Conference on Data Engineering (ICDE)*. IEEE, 2018, pp. 1109–1119.

[193] G. Weikum, L. Dong, S. Razniewski, and F. Suchanek, "Machine knowledge: Creation and curation of comprehensive knowledge bases," *arXiv preprint arXiv:2009.11564*, 2020.

[194] W. Yin, J. Hay, and D. Roth, "Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*

and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 3914–3923.