Construction and Applications of Billion-Scale Pre-Trained Multimodal Business Knowledge Graph

Shumin Deng*†
National University of Singapore
Singapore
shumin@nus.edu.sg

Chengming Wang*
Alibaba Group
Hangzhou, China
chengming.wcm@alibaba-inc.com

Zhoubo Li†
Zhejiang University
Hangzhou, China
zhoubo.li@zju.edu.cn

Ningyu Zhang†
Zhejiang University
China
zhangningyu@zju.edu.cn

Zelin Dai, Hethong Chen
Alibaba Group
Hangzhou, China
zelin.dzl@alibaba-inc.com
hehong.chh@alibaba-inc.com

Feiyu Xiong, Ming Yan
Alibaba Group
Hangzhou, China
feiyu.fyx@alibaba-inc.com
ym119608@alibaba-inc.com

Qiang Chen
Alibaba Group
Hangzhou, China
lapu.cq@alibaba-inc.com
chenmosha.cms@alibaba-inc.com

Mosha Chen
Alibaba Group
Hangzhou, China

Jiaoyan Chen
University of Manchester
United Kingdom
jiaoyan.chen@manchester.ac.uk

Jeff Z. Pan
University of Edinburgh
United Kingdom
j.z.pan@ed.ac.uk

Bryan Hooi
National University of Singapore
Singapore
dcsbhk@nus.edu.sg

Huajun Chen†
Zhejiang University
Hangzhou, China
huajunsir@zju.edu.cn

Abstract—Business Knowledge Graphs (KGs) are important to many enterprises today, providing factual knowledge and structured data that steer many products and make them more intelligent. Despite their promising benefits, building business KG necessitates solving prohibitive issues of deficient structure and multiple modalities. In this paper, we advance the understanding of the practical challenges related to building KG in non-trivial real-world systems. We introduce the process of building an open business knowledge graph (OpenBG) derived from a well-known enterprise, Alibaba Group. Specifically, we define a core ontology to cover various abstract products and consumption demands, with fine-grained taxonomy and multimodal facts in deployed applications. OpenBG is an open business KG of unprecedented scale: 2.6 billion triples with more than 88 million entities covering over 1 million core classes/concepts and 2,681 types of relations. We release all the open resources (OpenBG benchmarks) derived from it for the community and report experimental results of KG-centric tasks. We also run up an online competition based on OpenBG benchmarks, and has attracted thousands of teams. We further pre-train OpenBG and apply it to many KG-enhanced downstream tasks in business scenarios, demonstrating the effectiveness of billion-scale multimodal knowledge for e-commerce. All the resources with codes have been released at https://github.com/OpenBGBenchmark/OpenBG.

Index Terms—Knowledge Graph, Business, Billion-scale, Construction, Applications

1. INTRODUCTION

Knowledge Graphs (KGs) have attracted widespread attention from both academia and industry over the years [1], [2], and have been widely applied in many fields, such as education [3], [4], biomedical science [5]–[7], and finance [8], [9]. Specifically, business such as e-commerce has increasingly permeated all aspects of our daily life, business KGs [10], [11] have been accumulating a large amount of data in different fields [12]–[14], with the potential to address applications such as recommendation [15] and product search [16].

Motivation (Why to construct OpenBG). With the tremendous potential of business KG in many applications, we propose to construct a new business KG entitled OpenBG, characterized by (1) standardized organization and integration of business information about known “entities” and “relations”, as well as hyper-relational knowledge with business axioms and rules, for proprietary resources of an enterprise; (2) utilization of billion-scale multimodal business KG with pre-training, and serving for various knowledge-enhanced applications.

Challenges. OpenBG addresses the following two challenges: (1) Deficient Structure. Due to the noise and redundancy that inevitably exist in big data, large-scale business KGs built with it often suffer from structural defects. For example, as seen in Figure 1, “China” is both an instance of the class (Place) and a value of the attribute (pPlaceofOrigin), showing redundancy in definition; besides, “Cooking” and “Make Sushi” are closely related via the subClassOf relationship, but not directly linked in the current KG, indicating the lack of completeness. (2) Multiple Modalities. As big data are derived from multimedia sources, they often possess multiple modalities, e.g., texts, images, and tables, making it challenging to build such a multimodal KG from multi-source data. For example in Figure 1, to present a product “Iphone 14 Pro”, titles, technical specifications, advertising slogans, and product images all need to be formatted in a universal way.
Construction (How to construct OpenBG). To tackle the challenges and address the limitations of existing technologies, we propose to build a billion-scale open business knowledge graph (OpenBG) from multi-source and noisy big data. Specifically, we first design a core ontology of OpenBG with references to W3C standards\(^1\), mainly containing 8 types of core classes/concepts and 2,681 types of relations. Considering products are the core entities in business scenarios, and often rich in semantics with various attributes, we define the abstract types of products as classes\(^2\), entitled Category (产品类目). We also define Brand (品牌) and Place (地点) as classes w.r.t. their rich semantics and close associations with Category. Brand and Place are often described with rich knowledge and almost all products can be linked to them.

In addition, Category can also establish linkages to other objects without complex semantics, which are regarded as simple classes and defined as concepts referring to Simple Knowledge Organization System (SKOS)\(^2\). Concepts aim to bridge the semantic gap between user needs and products [17]–[19], which are also vital in business KGs. We define five types of concepts in OpenBG ontology: Time (时间), Scene (场景), Theme (主题), Crowd (人群), Market Segment (细分市场).

We then populate the OpenBG ontology by linking large-scale multimodal product triples to the pre-defined classes/concepts. We also polish the original ontology along with triple accumulation. Finally, we release the ongoing OpenBG containing 1,131,579 core classes/concepts, 88,881,723 entities, and 2,603,046,837 triples. Since the scale of OpenBG is enormous, we have deployed almost 4,000 persons/day of manpower to annotate the large-scale raw data from open sources, e.g., Alibaba e-commerce platform, and pre-processed the data with 100×V100 GPUs.

Applications (How to utilize OpenBG). To demonstrate the business value of OpenBG, we apply it in different dimensions. (1) KG-centric Applications. To simplify direct usage of the billion-scale KG, we construct and release OpenBG benchmarks, sampled from OpenBG, for both research analysis and industrial applications. We construct three OpenBG benchmarks\(^3\), including OpenBG-IMG, OpenBG500, and OpenBG500-L. Therein, OpenBG-IMG is a small multimodal KG with 14,718 multimodal entities (27,910 entities in total); OpenBG500 is a single-modal KG with 249,743 entities, and OpenBG500-L is a large-scale version of OpenBG500, containing 2,782,223 entities. We evaluate OpenBG benchmarks with some popular KG embedding models, and the experimental results demonstrate their promising capacity. Besides, we also run an online competition\(^4\) based on OpenBG benchmarks, which has attracted more than 4,700 researchers and thousands of participants (over 3,000 teams) at the time of writing.

(2) KG-enhanced Applications. To further unleash the tremendous potential of the billion-scale OpenBG, we pre-train OpenBG and apply it on many downstream tasks in business scenarios, making it serve as a backbone for e-commerce platforms. We select several tasks as examples, including link prediction of item categories; NER for item titles; item title summarization; IE for item reviews; salience evaluation for commonsense knowledge. We pre-train OpenBG with a vision-language foundation model mPLUG [20] and other baseline models. The experiments reflect that pre-training models enhanced with OpenBG can achieve satisfactory performance on all tasks, especially in low-resource scenarios. We also apply the pre-trained OpenBG to provide services on online e-commerce platforms, and present some cases on online systems in this paper. Note that during the pre-training stage, we leverage 14×A100 GPUs with 600,000 steps to train the large model for our tasks, and the entire pre-training process costs 32×V100 GPUs for 120 days approximately.

Contributions of this work can be summarized:

- We introduce a detailed process of constructing a billion-scale multimodal pre-trained business KG (OpenBG), with standard KG schema definitions, which have been applied to real-world systems, hopefully enlightening KG construction in other fields.
- We publish OpenBG, an ongoing business KG, containing more than 88 million entities covering over 1.1 million classes/concepts and over 2.6 billion triples. To the best of our knowledge, OpenBG is the most comprehensive business KG publicly available on the Web.
- We release OpenBG benchmarks and run an online competition for KG-centric applications through OpenBG pre-training, which have attracted thousands of participants, and we also integrate OpenBG to large pre-trained models for KG-enhanced applications.

---

1. https://www.w3.org/
2. https://www.w3.org/TR/skos-primer/
3. https://github.com/OpenBGBenchmark/OpenBG
4. https://tianchi.aliyun.com/dataset/dataDetail?dataId=122271&lang=en-us.
The remainder of this paper is organized as follows. Sec. II shows the process of building billion-scale OpenBG. Sec. III introduces construction and evaluation of the released OpenBG benchmarks. Sec. IV presents some KG-centric and KG-enhanced applications with pre-trained OpenBG. Sec. V reviews related work on business KGs. Sec. VI concludes and discusses the future work.

II. OPENBG CONSTRUCTION

A. Preliminary and Ontology Overview

To address the challenge of deficient structure in business KG construction, we propose to define a canonical ontology (or schema) for it with references to W3C standards, which imposes constraints on the links coupled with business logic. Furthermore, as the scale of business data is often enormous and unmanageable, defining the standard ontology is fundamental and necessary for business KG construction. We present the core ontology of OpenBG in Figure 2, mainly containing classes/concepts and relations. Note that we focus on ontology definition and KG construction in this section, and omit technical details about raw knowledge acquisition (mostly general NLP methods, such as named entity recognition, relation extraction, and coreference resolution which are already resolved in data pre-processing) due to page limits. As OpenBG is billion-scale and the raw data are also enormous, the data processing is time-consuming and labor-intensive. The annotation of raw data has cost 4,000 person/day approximately. We deploy the model on 100 × V100 GPUs for daily new data ingestion and processing. We also carry out routine maintenance of OpenBG quality, which costs almost 30 person/day.

Automation of OpenBG. Given the defined ontology and pre-processed structured data, we utilize a free and open source Java framework called Apache Jena to automatically (1) formalize OpenBG ontology with Jena ontology API, and then (2) populate OpenBG ontology by linking instances to it with RDF (Resource Description Framework) API.

Symbol Overview: The OpenBG ontology $O = \{C, P, R\}$ mainly comprises classes $C$, concepts $P$, and relations $R$. Therein, $C \equiv C^C \cup C^B \cup C^P$ contains Category $C^C$, Brand $C^B$, and Place $C^P$. A category $c \in C^C$ is regarded as the abstract types of a product $u$, denoted by $\langle u, \text{rdf: type}, c \rangle$, where products are core entities in OpenBG. Category $C^C$ with a rich hierarchy is defined to indicate the taxonomy of product, and Brand $C^B$, Place $C^P$ as well as concepts $P$ are all linkable to it via relations $R$.

Core Classes/Concepts: As seen in Figure 2, the central node in OpenBG ontology is Category, which indicates the taxonomy of products with various attributes, such as size, weight, and color. Considering products are informative with rich semantics, we define the abstract types of products (named Category) as classes in the ontology. For example, mobile phone, as an abstract type of iPhone 14 Pro, can be defined as a class of Category. In addition, products can be linkable to other objects, such as brands and places. We also define their abstract types as subclasses of Brand and Place class w.r.t. their rich knowledge, e.g., a brand also has various attributes such as founder and logo. Furthermore, products can also establish relations to brands and places.

Besides, products also can be linked to some objects without complex semantics, but these objects can create a bridge between products and users’ needs. For example, iPhone 14 Pro is related to a scenario of giving gifts (not requiring rich-semantic description), which represents the business knowledge that iPhone 14 Pro is commonly given as a gift. We define these simple-semantic objects as concepts, inspired by AliCoCo [17] (an e-commerce cognitive concept net). Concepts are deemed as simple classes, because they are abstract entities without complex semantics. We also refer to Simple Knowledge Organization System (SKOS) to standardize concept definitions.

Overall, we define 3 types of core classes to express knowledge related to products (i.e., "Category", "Place", "Brand") and 5 types of core concepts to express knowledge related to business concepts (i.e., "Time", "Scene", "Market Segment", "Crowd" and "Theme").

Core Relations: $R = \{R_{obj}, R_{meta}, R_{data}\}$ includes three types of relations, including object properties $R_{obj}$, meta-properties $R_{meta}$, and data properties $R_{data}$.

(1) $R_{obj}$: Object properties reflect relationships among classes/concepts, which constrain the type of head entity (domain) and tail entity (range). For example, placeOfOrigin conveys association between Category $C^C$ and Place $C^P$, thus the domain of placeOfOrigin must belongs to $C^C$, and the range must belongs to $C^P$. As seen in Figure 2, we define some object properties to model associative relationships between Category and other core classes/concepts, such as brands, placeOfOrigin, appliedTime, relatedScene, aboutTheme, forCrowd, inMarket (a set of relations, in short for brevity).

(2) $R_{meta}$: Meta-properties (also called built-in proper-
ties) indicate axioms of the ontology defined by W3C standards\(^7\). We import four types of meta-properties to express taxonomy (rdfs:subClassOf, skos:broader), synonymy (owl:equivalentClass) and instantiation (rdf:type) of classes/concepts. For example, (Category, rdfs:subClassOf, owl:Thing), (Scene, skos:broader, skos:Concept), and \(\langle c/p, \quad \text{owl:equivalentClass, } x \rangle\), where \(c \in C, \quad p \in P, \quad \text{and } x\) denotes an accessible exogenous object linked to OpenBG. Moreover, we also import two other types of meta-properties to express properties of properties, i.e., rdfs:subPropertyOf and owl:equivalentPropertyOf.

(3) \(R_{\text{data}}\): Data properties are used to describe the attributes of classes. For example, the iPhone 14 Pro product has the data properties of size, weight, color, and so on. Considering that data properties in business scenarios are often derived from the general domain, we also link some data properties in OpenBG to cnSchema\(^6\) (an open Chinese KG schema, application-oriented) with two meta-properties: rdfs:subPropertyOf and owl:equivalentPropertyOf.

The detailed statistics of the ongoing OpenBG in current version are shown in Table I. We also show a snapshot of OpenBG containing some examples in Figure 3.

### TABLE I

**Statistics of OpenBG at the time of writing.**

| Overall          |       |       |       |       |       |       |       |       |       |
|------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| # core classes   | 460,805 |       |       |       |       |       |       |       |       |
| # core concepts  | 670,774 |       |       |       |       |       |       |       |       |
| # relation types | 2,681  |       |       |       |       |       |       |       |       |
| # products (instances of categories) | 3,062,313 |       |       |       |       |       |       |       |       |
| # triples        | 2,603,046,837 |       |       |       |       |       |       |       |       |

| Core Class/Concept | # level1 | # level2 | # level3 | # level4 | # level5 | # All | # leaf |
|---------------------|----------|----------|----------|----------|----------|-------|--------|
| Category            | 93       | 899      | 3,080    | 3,049    | 7,108    | 3,399 |        |
| Brand               | 45       | 411,234  | 411,279  | 411,284  | 411,299  |       |        |
| Place               | 208      | 266      | 333      | 2,847    | 38,773   | 42,426 | 38,969 |
| Scene               | 19       | 4,027    | 617      | 729      | 5,392    | 5,170 |        |
| Crowd               | 8        | 37       | 45,105   | 57       | 45,207   | 45,180 |        |
| Theme               | 14       | 5,219    | 143      | 143      | 5,519    | 5,360 |        |
| Time                | 1        | 55       | /        | /        | 58 / 55  |        |        |
| Market_S            | 614,598  | /        | /        | /        | 614,598 / 614,598 |        |        |

**Core Relation**

| object property | # brands | # placeOfOrigin | # relatedScene | # forCrowd | # forMarket |
|-----------------|----------|-----------------|----------------|------------|------------|
| # aboutTheme    | 132,135  | 773,371         | 1,654,057,075 | 3,906,635  | 5,466,655  |
| # hasImage      | 3,062,313 | 3,072,337       | 670,774        | 670,774    | 490,096    |
| # hasImage      | 3,062,300 | 102,504         | 75,466,635     |            |            |
| # rdf:label     | 3,062,300 | 102,504         | 75,466,635     |            |            |
| # skos:preLabel | 3,062,300 | 102,504         | 75,466,635     |            |            |
| # skos:inLabel  | 3,062,300 | 102,504         | 75,466,635     |            |            |
| # rdfs:label    | 3,062,300 | 102,504         | 75,466,635     |            |            |
| # rdfs:subClassOf | 460,760 | 670,774         | 670,774        | 490,096    | 2,448     |

**Construction of “Category”**. We follow a top-down approach to constructing Category.

(1) Define “Category” and Taxonomy. We first create a single top-level class: Category, before specializing it with subclasses. We further break down each class layer by layer, for example, electronic components are broken down into some subclasses, such as LED, power supply, etc. To evaluate the quality of Category definition, we schedule qualified human resources (experts in e-commerce, 30 person/day) to go through a daily review process. The main concern of Category quality includes: (i) if the class label definition is clear and reasonable enough; (ii) if a category contains its child nodes completely; (iii) if a category excludes child nodes of other categories; (iv) if products associated with the category category is popular enough; (v) if the category is acknowledged by customers and sellers. Domain experts will rate and tune each category according to these factors every day.

(2) Create Multimodal Instances of “Category”. In fact, the instances of “Category” are products in business scenarios. We randomly sample some products for each leaf node in category taxonomy from Alibaba e-commerce platform. To address the challenge of multiple modalities in business KG construction, we formalize multimodal information of products all in triple format with different relations. To organize semi-structured association and attribute information, we associate products with other classes/concepts through object properties \(R_{\text{obj}}\), e.g., (iPhone 14 Pro, brandIs, Apple); we then define data properties \(R_{\text{data}}\) of each product to express its attribute-related information, e.g., (iPhone 14 Pro, weight, 206g). Particularly, to organize unstructured attribute information, we retrieve the textual corpus of product description and images of products from Alibaba e-commerce platform, and link them to products with two data properties (rdfs:comment, imageIs), enabling multimodal knowledge about products, i.e., textual descriptions and images.

**Construction of “Place” and “Brand”**. We generally utilize a schema mapping approach [4], [21] to construct “Place” and “Brand” from diverse sources.

(1) Define “Place” and Taxonomy. We integrate and transform administrative region data in different formats from diverse KGs, such as Wikidata\(^1\) and OpenKG\(^8\). The taxonomy of “Place” includes country or great region, province, city, county, village/town in order.

(2) Define “Brand” and Taxonomy. “Brand” is divided into 45 major categories following the guideline for declaration of goods\(^9\), e.g., Clothes, Furniture, and Vehicle. We also integrate and transform various kinds of brand-related data from authoritative websites, e.g., the logo of a brand on its official homepage.

(3) Link “Place” and “Brand” with Products. We link “Place” and “Brand” to products via matching their textual labels. Concretely, for each product containing place and brand
information, we map the textual labels of its place and brand to standard names defined in “Place” and “Brand” taxonomy, by jointly conducting trie prefix tree precise matching and fuzzy matching of synonyms.

C. Construction of Ontology Concepts

OpenBG currently contains five types of core concepts, i.e., Scene, Crowd, Theme, Time, and Market Segment. We follow a bottom-up approach for Concept taxonomy construction.

(1) Create Instances of “Concept”. We first leverage sequence labeling to extract concepts from real-world large-scale business text corpus, such as user-written reviews, product titles, and search queries. We utilize BERT-CRF model, a popular model for many sequence labeling tasks [22], [23]. BERT-CRF model consists of a BERT layer and a CRF (Conditional Random Field) layer, where BERT [24] enables to obtain a contextual representation for each word and CRF considers the correlations between the current label and neighboring labels.

(2) Define “Concept” and Taxonomy. Given the five top-level broad concepts: Scene, Crowd, Theme, Time, and Market Segment, we classify instances of “Concept” into these five predefined concepts. Then we summarize the narrower concepts to broader ones level by level, until to the top level. Moreover, to evaluate the quality of concepts, we formulate concepts in leaf nodes following multi-faceted commonsense knowledge [25], mainly including four dimensions:

(i) Plausibility: Indicating whether a concept-oriented statement is meaningful, e.g., (sports shoes, forCrowd, the elderly) where sports shoes are products and the elderly are a narrower concept of Crowd, is a meaningful statement for “the elderly”.

(ii) Typicality: Indicating whether a concept-oriented statement is valid for the majority of instances, e.g., (lightweight sports shoes, forCrowd, the elderly) are typical while (trendy sports shoes, forCrowd, the elderly) are not, because trendy shoes are not suitable for most of elderly people.

(iii) Remarkability: Indicating whether a concept is distinguishable enough from closely related ones, e.g., (non-slip shoes, forCrowd, the elderly) are remarkable while (thin and light shoes, forCrowd, the elderly) are not, because thin and light shoes are also suitable for young people.

(iv) Salience: Indicating whether a concept is representative enough, so that instances can be associated with it spontaneously. Generally, a statement both satisfying Typicality and Remarkability implies Salience.

III. OPENBG BENCHMARK: KG-CENTRIC APPLICATION

To promote further research in developing plausible KG representation solutions to real-world applications, we present the OpenBG Benchmark, a challenging benchmark to facilitate reproducible, scalable, and multimodal KG research. OpenBG Benchmark is sampled from OpenBG, covering most of the features defined in OpenBG. Specifically, OpenBG Benchmark is designed following three principles: (1) Key problems in business KG. We select tasks to address critical academic and industrial issues that remain to be solved for business KG, (2) Business application-centric task design. All the tasks in the benchmark are selected based on business scenarios in Alibaba Group to address real-world applications. (3) Quality control in long-term maintenance. We utilize several denoising strategies and artificially review data to ensure data quality. Thus, OpenBG Benchmark is mature enough to be considered
for industry-wide acceptance. We also conduct experiments and release leaderboards publicly.

A. Benchmark Construction

OpenBG benchmark contains several subsets, including OpenBG-IMG, OpenBG500, and OpenBG500-L. Therein, OpenBG-IMG is a small multimodal dataset, OpenBG500 is a single-modal dataset, while OpenBG500-L is a large-scale version of OpenBG500, aiming to benchmark efficient machine learning methods over the large-scale KG. We show the statistical details of the OpenBG benchmark with comparison to some existing large-scale multimodal KG Wikidata5M [26] and OGB [27], [28] in Table II.

Table II

| Dataset          | # Ent | # Rel | # Train | # Dev | # Test |
|------------------|-------|-------|---------|-------|--------|
| OpenBG-IMG       | 27,910 | 136   | 230,087 | 5,000 | 14,675 |
| OpenBG500        | 249,743 | 500   | 1,242,550 | 5,000 | 5,000  |
| OpenBG500-L      | 2,782,223 | 500   | 47,410,032 | 10,000 | 10,000 |
| OpenBG (Full)    | 88,881,723 | 2,681 | 260,304,568 | -    | -      |
| Wikidata5M       | 4,594,485 | 822   | 20,614,279 | 5,163 | 5,133  |
| OGB-LSC          | 91,230,610 | 1,387 | 608,062,811 | 15,000 | 10,000 |

Given OpenBG (full) with a full set of entities, relations, triples, denoted by \{E, R, T\}, we utilize a three-stage method to build high-quality OpenBG benchmarks, including step 1: selecting relations from R (relation refinement); step 2: filtering head entities from E (head entity filtering); step 3: sampling tail entities in triples T (tail entity sampling). We illustrate the process of building OpenBG benchmarks in Figure 4.

(1) **Relation Refinement.** We follow two principles to manually filter representative relations in OpenBG: (i) To select high-frequency relations, as they are supposed to imply a relatively high-value applications. (ii) To select closely business-related relations, as they directly describe the attributes of products. In this way, we obtain relation subsets \(R_{500}, R_{500-L}\) and \(R_{136}\), respectively containing 500 (for OpenBG500 & OpenBG500-L) and 136 (for OpenBG-IMG) relations, where \(R_{136} \subseteq R_{500}, R_{500-L} \subseteq R\). Note that some triples contain no image, thus OpenBG-IMG has fewer relations than OpenBG500.

We show the distribution of 136 relations in OpenBG-IMG, as seen in Figure 5, which follows a long-tail distribution.

OpenBG500 as well as OpenBG500-L also follow similar relation distributions, and we omit figures due to page limits.

(2) **Head Entity Filtering.** Regarding the long-tail relation distribution of \(R_{136}, R_{500}\), and \(R_{500-L}\), we split \(R_N (N = 136, 500, 500-L)\) into head-relations \(R_{N_{head}}^N\) and tail-relations \(R_{N_{tail}}^N\). Then we divide the entities into \(R_{head}^N\) and \(R_{tail}^N\) (\(E_{N_{head}}^N\) and \(E_{N_{tail}}^N\), respectively) and also distribute a sampling rate \(\alpha_{N_{head}}\) for the former, as well as \(\alpha_{N_{tail}}\) for the latter, where \(\alpha_{N_{head}} > \alpha_{N_{tail}}\). The entity sampling can be denoted by:

\[
E_N = \text{Sample}(E_{N_{head}}, \alpha_{N_{head}}) + \text{Sample}(E_{N_{tail}}, \alpha_{N_{tail}})
\]

where \(\text{Sample}(S, \alpha)\) denotes randomly sampling the set \(S\) at a certain rate \(\alpha\), and \(E_N\) (where \(N = 136, 500, 500-L\)) is the sampled head entities for triples.

(3) **Tail Entity Sampling.** Given the filtered relations \(R_N (N = 136, 500, 500-L)\) and sampled head entities \(E_N\) in triples, we firstly filter triples \(T(E_N, R_N)\) from \(T\), with head entities in \(E_N\), relations in \(R_N\) and all corresponding tail entities of the given relations. Then we build three benchmarks \(T_N (N = 136, 500, 500-L)\) entitled OpenBG-IMG, OpenBG500, OpenBG500-L through sampling tail entities from \(T(E_N, R_N)\) and connecting them with head entities \(E_N\) and relations \(R_N\) to obtain the whole triples, denoted by:

\[
T_N = \text{Sample}(T(E_N, R_N), \alpha_N)
\]

where \(\alpha_N\) is determined by each benchmark \(T_N\).
TABLE III
RESULTS OF LINK PREDICTION ON OPENBG-IMG. THE BOLD NUMBERS DENOTE THE BEST RESULTS.

| Model          | Hits@1 | Hits@3 | Hits@10 | MR  | MRR  |
|----------------|--------|--------|---------|-----|------|
| Single-Modal Approaches |        |        |         |     |      |
| TransE [29]    | 0.150  | 0.387  | 0.647   | 118 | 0.315|
| TransH [30]    | 0.129  | 0.525  | 0.743   | 112 | 0.357|
| TransD [31]    | 0.137  | 0.532  | 0.746   | 110 | 0.364|
| DistMult [32]  | 0.060  | 0.157  | 0.279   | 524 | 0.139|
| ComplEx [33]   | 0.143  | 0.244  | 0.371   | 782 | 0.221|
| TuckER [34]    | 0.497  | 0.690  | 0.820   | 1473| 0.611|
| KG-BERT [35]   | 0.092  | 0.207  | 0.405   | 61  | 0.194|
| StAR [36]      | 0.176  | 0.307  | 0.493   | 79  | 0.280|
| Multi-Modal Approaches |        |        |         |     |      |
| TransAE [37]   | 0.274  | 0.489  | 0.715   | 36  | 0.421|
| RSME [38]      | 0.485  | 0.687  | 0.838   | 72  | 0.607|
| MKGformer [39] | 0.448  | 0.651  | 0.822   | 23  | 0.575|

TABLE IV
RESULTS OF LINK PREDICTION ON OPENBG500 AND OPENBG500-L. THE BOLD NUMBERS DENOTE THE BEST RESULTS.

| Model          | OpenBG500 | OpenBG500-L |
|----------------|-----------|-------------|
|                | Hits@1↑   | Hits@3↑     | Hits@10↑  | MR↓  | MRR↑ | Hits@1↑   | Hits@3↑     | Hits@10↑  | MR↓  | MRR↑ |
| TransE [29]    | 0.207     | 0.340       | 0.513     | 5381 | 0.304| 0.314     | 0.583       | 0.820     | 888  | 0.482|
| TransH [30]    | 0.143     | 0.402       | 0.569     | 6501 | 0.296| 0.247     | 0.569       | 0.813     | 1157 | 0.441|
| TransD [31]    | 0.146     | 0.411       | 0.576     | 6129 | 0.302| 0.279     | 0.575       | 0.813     | 1157 | 0.441|
| DistMult [32]  | 0.068     | 0.131       | 0.255     | 5709 | 0.129| 0.012     | 0.147       | 0.299     | 3065 | 0.108|
| ComplEx [33]   | 0.081     | 0.187       | 0.313     | 6393 | 0.156| 0.088     | 0.195       | 0.300     | 4569 | 0.165|
| TuckER [34]    | 0.428     | 0.615       | 0.735     | 2573 | 0.541|         |             |           |      |      |
| KG-BERT [35]   | 0.071     | 0.145       | 0.262     | 401  | 0.138|         |             |           |      |      |
| GenKGC [40]    | 0.203     | 0.280       | 0.351     | -    | -    |         |             |           |      |      |

B. Baselines and Settings
To evaluate the quality of OpenBG benchmarks, we implement link prediction experiments on several baselines regarding their varying scales and settings.

(1) Single-modal KG Embedding Approaches\(^{10}\) for OpenBG500 & OpenBG500-L & OpenBG-IMG: TransE [29], TransH [30], TransD [31], DistMult [32], ComplEx [33], TuckER [34], and StAR [36] leverage the KG structure for link prediction. KG-BERT [35] utilizes the description text in KG embedding with the pre-trained language model. GenKGC [40] converts KG embedding to sequence-to-sequence generation task with the pre-trained language model.

(2) Multimodal KG Embedding Approaches for OpenBG-IMG only: TransAE [37] uses a multimodal auto-encoder based on TransE for a unified representation of textual and visual input. RSME [38] designs a filter gate and a forget gate to enhance visual information learning. MKGformer [39] utilizes a hybrid transformer architecture with multi-level fusion for better multimodal entity representation.

Implementation Details of Baselines. We utilize top-K hit rate (Hits@K, K = 1, 3, 10), Mean Rank (MR), and Mean Reciprocal Ranking (MRR) as evaluation metrics. Note that OpenBG-IMG is in the multimodal setting with both texts and images, while OpenBG500 and OpenBG500-L follow vanilla settings. We conduct experiments on 1×V100 GPU (32G NVIDIA TESLA) in PyTorch 1.8.1 environment. With regard to settings of the training process on different baselines, we respectively use AdaGrad and SGD optimizer, with epochs of 5; 100; 200; 500; 1,000. The dimension of embedding is 200; 100; 200; 500; 1000; 1500, and learning rate is 1.0; 0.5; 5e-4; 3e-5; 1e-5.

C. Overall Performance Comparison
We have conducted both single-modal and multimodal link prediction on OpenBG benchmarks. Link prediction in this paper aims to identify missing connections in a KG based on the features of existing triples. For example, given an incomplete triple \((h, r, ?)\) with head entity \(h\) and relation \(r\), we aim to predict a tail entity \(t\) to complete the triple. \(h\) and \(t\) can both be of the text modality (single-modal) or respectively be of the text and image modality (multimodal).

(1) Single-modal Link Prediction. The experimental results in Table III and Table IV illustrate that translational distance models (TransE, TransH, etc.) greatly outperform normal bilinear models (DistMult, ComplEx, etc.). We notice that TuckER can archive higher Hits@K and MRR scores due to

---

\(^{10}\) Note that due to the large scale of OpenBG500-L, for some baselines requiring excessive computational resources, e.g., TuckER [34], we omit part of experiments as only one V100 GPU is available for benchmark experiments.
the powerful representation ability of Tucker decomposition. However, TuckER obtains the worst score on the MR metric in Table III; we argue that the baseline performance may vary among different instances. We further observe that baselines based on textual embeddings such as KG-BERT and StAR are not competitive here. For OpenBG500-L, we notice that vanilla TransE can yield better performance than most of the sophisticated baselines, which indicates that more works should be investigated for large-scale KG representation learning.

2) Multimodal Link Prediction. The experimental results in Table III show that RSME achieves the best performance except on MR metric, indicating its advanced multimodal information fusion ability. MKGformer can yield better results on MR metric and achieve comparable performance on the other metrics compared with RSME.

D. Distribution and Maintenance

OpenBG benchmarks have already been released publicly together with an online competition. Up to now, more than 4,700 researchers have applied OpenBG benchmarks, and over 3,000 teams have signed up for the competition. We will continue to maintain and expand the benchmarks, and plan to enrich them with more new datasets.

IV. OPENBG PRE-TRAINING: KG-ENHANCED APPLICATION

A. Overview

To further unleash the potential of OpenBG, we pre-train the large model enhanced with OpenBG and apply it to e-commerce downstream tasks requiring domain knowledge about items.

Data for Pre-Training. The supervised data comprises OpenBG and annotated item titles and reviews, which can be denoted as label-sample pairs, such as product-category; item-product; item-title; item-image; item-triple; short-title; long-title; item-review; triple-review. For texts and KG, we normalize these data in different modalities into unified textual expressions with artificially constructed discrete prompts, and obtain almost 100 million supervised text pairs, denoted by $X_{sup} = (X_{src}, X_{tgt})$. In addition, we also collect about 140GB of unsupervised texts in e-commerce scenarios, such as user reviews and product descriptions, denoted by $X_{uns}$. We then averagely combine supervised text pairs and unsupervised e-commerce texts together as the input text tokens, denoted by $X = [X_{sup}, X_{uns}]$. Text tokens $X$ along with visual tokens $V$ constitute the model input, denoted by $\{X, V\}$.

Model for Pre-training. We illustrate the framework of the large model pre-training enhanced with OpenBG in Figure 6. As seen, our pre-training model adopts a vision-language foundation model mPLUG [20] as the backbone, which is pre-trained end-to-end on large-scale image-text pairs with both discriminative and generative objectives. We revise the text encoder of vanilla mPLUG into a KG-enhanced text encoder, where the texts and KG triples are converted into unified text tokens as input. We utilize cross entropy as the loss function for supervised data, similar to the multi-task pre-training of ExT5 [41]; and we adopt the span-denosing objective [42] for unsupervised data.

Distributed Pre-training for Large Scale. Training the big mPLUG model on large-scale datasets along with OpenBG is extremely challenging. Similar to primary mPLUG [20], to implement time-saving computation, we use BF16 precision training, which is a new data type supported by NVIDIA’s new Ampere architecture GPU like A100. We have trained mPLUG model for 600,000 steps on 14×A100 GPUs in a data-parallel manner, with the batch size of 512 and weight decay of 0.02. We use AdamW optimizer, and adopt a linear schedule to the learning rate with warmup of 0.1.

Model Fine-tuning. Each downstream task is formatted as a generation task and fine-tuned separately based on the pre-trained model until the loss no longer decreases on the validation set. To prevent label leakage, the data in the validation set of downstream tasks is restricted not to appear in the pre-training period.

In this section, we will take five representative downstream tasks as examples, to introduce how to utilize OpenBG to enhance pre-trained large models and serve for real-world applications, including: category prediction, NER for titles, title summarization, IE for reviews, and salience evaluation.

B. Category Prediction

1) Task Definition: Category prediction means link prediction in a KG specifically for item categories. Given an incomplete triple $e, \text{rdfs:subClassOf}, ?$ in which $e$ denotes an item, we require to find the missing tail entity. For example, given an item entitled “Northeast Wuchang Rice Daohuaxiang No.2 (东北五常大米稻花香2号)”, the model should link the leaf category “Rice (大米)” to it with a relation of rdfs:subClassOf.
(2) Baseline Models: We select RoBERTa\textsubscript{large} [43] as a baseline model to compare with other methods based on mPLUG. RoBERTa\textsubscript{large} [43] is an improved replication of BERT [24] and achieves promising performance in many downstream tasks. Note that the RoBERTa\textsubscript{large} adopted in this task is pre-trained with corpus in general domains and excluding data in the business domain.

(3) Dataset Details: The dataset for category prediction includes a training set of 10 million triples which is generated from crowd-sourced annotations, and a validation set of 210 thousand triples which is created by domain experts. The total number of categories is 13,000 (referring to the entire e-commerce platforms, thus containing more categories than OpenBG). For low-resource scenarios, we randomly sample categories with 1 shot and 5 shots respectively.

(4) Experimental Analysis: We evaluate category prediction task with the metric of accuracy, and present experimental results in Table V. As seen, on the category prediction task, mPLUG\textsubscript{base} with both discriminative and generative objects is able to achieve better results than the discriminative model, i.e., RoBERTa\textsubscript{large}, and the enhancement with KG also brings about 1.38% improvement. However, increasing the model capacity (comparing mPLUG\textsubscript{large+KG} to mPLUG\textsubscript{base+KG}) only achieves 0.12% outperformance, and we estimate that the phenomenon is probably due to inevitable noises in the training set.

### TABLE V
Evaluation results of downstream tasks.

| Models                  | Category Prediction (Accuracy) | NER for Titles (P | R | F) | Title Summarization (ROUGE-L) | IE for Reviews (P | R | F) | Salience Evaluation (Accuracy) |
|-------------------------|-------------------------------|-----------------|-----|-----|-----------------------------|-----------------|-----|-------------------------------|
| RoBERTa\textsubscript{large} [43] | 68.80                         | 70.10 | 69.70 | 69.10 | /                           | /               | /   | /                             |
| UIE [44]                | /                             | 66.35 | 63.70 | 65.00 | /                           | /               | /   | /                             |
| mT5 [45]                | /                             | /     | /     | /     | 70.12                       | 83.42           | 83.21 | 83.32                         |
| BERT [24]               | /                             | /     | /     | /     | 71.82                       | 82.60           | 83.06 | 82.83                         |
| mPLUG\textsubscript{base} [20] | 73.10                         | 69.68 | 65.98 | 67.78 | 71.82                       | 82.60           | 83.06 | 82.83                         |
| mPLUG\textsubscript{base+KG} [20] | 74.48                         | 73.97 | 72.05 | 73.00 | 72.30                       | 83.99           | 83.53 | 83.76                         |
| mPLUG\textsubscript{large+KG} [20] | 76.60                         | 75.09 | 72.52 | 73.79 | 78.29                       | 84.87           | 84.95 | 84.91                         |

We also evaluate the performance of category prediction in low-resource scenarios, and the results are shown in Table VI. As seen, mPLUG\textsubscript{base+KG} with KG enhanced achieves the outperformance of 11.06% than mPLUG\textsubscript{base} in the 1-shot setting, 3.01% in the 5-shot setting. RoBERTa\textsubscript{base+KG} enhanced with OpenBG even exceeds RoBERTa\textsubscript{large} by 11.58% in the 1-shot setting, and 0.26% in the 5-shot setting. Thus, we can conclude that enhance the model with OpenBG can significantly improves performance in few-shot category prediction tasks, and the more deficient data is, the more advantageous to integrate models with KG. Besides, increasing the capacity of the model can also helpful, since mPLUG\textsubscript{large+KG} achieves 8.74% and 1.39% outperformance than mPLUG\textsubscript{base+KG} in the 1-shot and 5-shot setting, respectively.

### C. NER for Titles

(1) Task Definition: NER (Named Entity Recognition) for titles aims to recognize the properties and corresponding values of items, with the input of item titles. For example, given an item entitled “Zero-fat Konjac Noodles 100g*3 (零脂魔芋面条100克*3袋)”, the model should find out the property-value pairs, including [Nutrients (营养成分): Zero-fat (零脂)], [Ingredients (原料成分): Konjac (魔芋)], [Category (品类): Noodles (面条)], [Packing Specification (包装规格): 100g*3 (100克*3袋)].

(2) Baseline Models: We select RoBERTa\textsubscript{large} [43] and UIE [44] as baselines, thereinto, UIE [44] is a unified text-to-structure generation framework, which can uniformly handle different information extraction tasks such as NER, and the model performance is also promising. We also pre-train UIE with corpus in general domains.

(3) Dataset Details: The dataset of NER for titles is generated from crowd-sourced annotations, including 1,500\textsuperscript{12} samples covering 167 entity types in total. The training set and validation set is randomly splitted with the ratio of 8:2. For low-resource scenarios, we randomly sample entity types with 1 shot and 5 shots respectively.

(4) Experimental Analysis: We evaluate NER for titles with the metric of precision (P), recall (R), F1 score (F), and present experimental results in Table V. As seen, mPLUG\textsubscript{base+KG} outperforms RoBERTa\textsubscript{large} on P, R, F, while mPLUG\textsubscript{base} performs even worse than RoBERTa\textsubscript{large}, indicating the significant advantage of introducing OpenBG to large models. The reason may relate to the great correlation between KG and NER tasks, and titles often contain rich attribute information which are also expressed in KG triples. For example, the item title “Zero-fat Konjac

\textsuperscript{12}To annotate NER datasets in business scenarios is actually a hard work, and we will continue to enrich the ongoing annotations.
Noodles 100g*3 (零脂魔芋面条100克*3袋)” implies the triple of ⟨e, Nutrients (营养成分), Zero-fat (零脂)⟩ already existing in KG, where e denotes the corresponding product of the item.

We also evaluate the performance (F1 score) of NER for titles in low-resource scenarios, and the results are shown in Table VII. As seen, in 1-shot setting, mPLUGbase performs worse than UIE, and mPLUGbase+KG enhanced with KG exceeds UIE, demonstrating that KG is significant in low-resource scenarios. Furthermore, both mPLUGbase and mPLUGbase+KG performs worse than RoBERTa base+KG and mT5 base+KG, as shown in Table V. As seen, vanilla mPLUG (mPLUGbase) underperforms than mT5, while when enhanced with KG, mPLUGbase+KG exceeds mT5. The possible reasons is that mPLUG base merely exceeds relatedScene (quality (质量), nice (不错)) and ⟨cushion (坐垫), size (大小), suitable (合适)⟩.

| Model                  | 1-Shot F1 | 5-Shot F1 |
|------------------------|-----------|-----------|
| UE                     | 57.20     | 66.80     |
| RoBERTabase+KG         | 59.60     | 67.90     |
| mPLUGbase+KG           | 40.51     | 50.96     |
| mPLUGbase+KG           | 57.84     | 61.55     |
| mPLUGlarge+KG          | 62.57     | 70.41     |

**D. Title Summarization**

1. **Task Definition:** Title summarization is designed to summarize the expatiatory titles of items (as item titles contain redundant information in most cases), similar to text summarization. For example, given the item title of “Lagogo Lagu Valley 2018 Summer New Women’s Word-neck Short-sleeved Floral Skirt Dress Beach Skirt Long Skirt Tide (Lagogo拉谷谷2018夏新款女一字领短袖碎花裙连衣裙沙滩裙长裙潮)”, the model can simplify it to “Lagogo One-Line Neck Short Sleeve Floral Dress (Lagogo一字领短袖碎花连衣裙)”.

2. **Baseline Models:** We select mT5 [45] as our baseline model for title summarization. mT5 [45] is a multilingual variant of T5 [42], the text-to-text transfer transformer, and demonstrates great performance on many multilingual benchmarks. We also pre-train mT5 with corpus in general domains.

3. **Dataset Details:** The dataset for title summarization is created by crowd-sourced annotations, including 20,000 samples in total. The training set and validation set is randomly splitted with the ratio of 8:2.

4. **Experimental Analysis:** We evaluate title summarization with the metric of ROUGE-L, and present experimental results in Table V. As seen, mPLUGbase already excels mT5, and mPLUGbase+KG enhanced with KG even more predominant. The probable reasons for the outperformance are mainly comprised two aspects: The first one is that we pre-train mPLUGbase with Chinese corpus, which can achieve better performance in Chinese datasets compared to mT5; the second one is that KG with attribute triples can help to identify phrases containing key attributes of items, so that making text summarization more meaningful. Besides, increasing the model capacity can also bring more significant improvement than introducing KG information in title summarization.

**E. IE for Reviews**

1. **Task Definition:** IE (Information Extraction) for reviews aims to extract structured information which describes and evaluates the item from the customer reviews. For example, given the item review of “The quality of the cushion is good, I bought it for my dad, the size is right, he likes it very much (坐垫质量不错，给老爸买的，大小合适，他很喜欢)”, the model should figure out some triples, such as ⟨cushion (坐垫), quality (质量), nice (不错)⟩ and ⟨cushion (坐垫), size (大小), suitable (合适)⟩.

2. **Baseline Models:** We also select mT5 [45] as our baseline model to implement IE for item reviews. We covert IE into a generative task and also pre-train mT5 model with general-domain corpus.

3. **Dataset Details:** The dataset of IE for item reviews is created by crowd-sourced annotations, including 20,000 samples in total. The training set and validation set is randomly splitted with the ratio of 8:2.

4. **Experimental Analysis:** We evaluate IE for reviews with the metric of precision (P), recall (R), F1 score (F), and present experimental results in Table V. As seen, mPLUGbase underperforms than mT5, while when enhanced with KG, mPLUGbase+KG exceeds mT5. The possible reasons is that taxonomy knowledge defined in OpenBG can help the model to distinguish the types of various entities in the texts, and attribute knowledge contains in OpenBG can help to implement attribute extraction of items.

**F. Salience Evaluation**

1. **Task Definition:** Salience evaluation for commonsense knowledge in business scenarios is defined at [46], where salience [25] reflects if a statement (triple) is characteristic enough, meaning that the relation in the triple can be regarded as a key trait of the entity. Given a triple t = ⟨e_h, r, e_t⟩, the model should to estimate if t is salient (score to 1) or not (score to 0). For example, ⟨running shoes, relatedScene, running⟩ is salient and should score to 1, while ⟨shoes, relatedScene, running⟩ is not and should score to 0.

2. **Baseline Models:** We select BERT [24] as the baseline model for commonsense knowledge salience evaluation in business scenarios, which stands for bidirectional encoder from transformers and gains satisfactory achievements in many NLP tasks. BERT is also pre-trained with general-domain corpus.

3. **Dataset Details:** The dataset for salience evaluation is also created by domain experts, including 23,000 samples in total. The training set and validation set is randomly splitted with the ratio of 8:2.

4. **Experimental Analysis:** We evaluate salience of triples with the metric of accuracy, and present experimental results in Table V. As seen, vanilla mPLUG (mPLUGbase) and KG enhanced mPLUG models (mPLUGbase+KG, mPLUGlarge+KG) all excels BERT at accuracy. Moreover, the outperformance gained from KG enhancement is more prominent than from model capacity expansion, as mPLUGlarge+KG merely exceeds mPLUGbase+KG by 0.42%, and both of them can exceed BERT by more than 7%. This result is instinctive, because salience
evaluation is heavily dependent on commonsense knowledge in retail, which is exactly the crucial constituent in OpenBG and mainly expressed by concepts.

G. Other Applications and Online Cases

Cases on Online Systems. We deploy the pre-trained large model enhanced with OpenBG on some online systems of Alibaba Group, such as Taobao App (淘寶). As seen in Figure 7, when users enter the Channel of “Taobao Foodies (淘寶吃貨)” on Taobao App, they will find some slogan and tips with the item to provide shopping guide, which are summarized from item titles and reviews based on the pre-trained KG-enhanced large model. For example, at the Module of “Meals without Cooking (三餐不開火)”, an item of “konjac noodles (魔芋面)” is marked as “delicious soup and taste (汤汁鲜美)”, and an item of “cold noodles (凉面)” presents a short review of “convenient and suitable for summer (方便，适合夏天)”. These slogans and tips align with items can help users promptly pick out the items that they intend to purchase, making the shopping guide more intelligent and user-friendly.

(1) Item Alignment. The target is to identify different items referring to the same product. With OpenBG, items referring to the same product can be recognized more easily based on the product schema, such as the corresponding category and attributes. The evaluation metric is Gross Merchandise Volume (GMV) of aligned items, and larger is better. GMV of Alibaba e-commerce platforms has promoted about 45% after deploying the pre-trained KG-enhanced large model.

(2) Shopping Guide. The target is to guide users to purchase items. With OpenBG, items can be tagged with various concepts and linked with more enriched information, so that users can better understand items and intend to consume. The evaluation metric is Cost Per Mille (CPM), and larger is better. CPM of Alibaba e-commerce platforms has promoted about 28.1% after deploying the pre-trained KG-enhanced large model.

(3) QA-based Product Recommendation. The target is to recommend items to users in QA scenarios. With OpenBG, AliMe (a smart QA robot) [47] can better understand users’ intention and recommend items more precisely. The evaluation metric is Click-Through-Rate (CTR), and larger is better. CTR of Alibaba e-commerce platforms has promoted about 11% after deploying the pre-trained KG-enhanced large model.

(4) Emerging Product Release. The target is to release emerging products. With OpenBG, when emerging products release, their attribute information can be automatically filled in via inheriting from the categories, so as to improve efficiency and save time. The evaluation metric is duration time of emerging products release, and smaller is better. Alibaba e-commerce platforms has reduced about 30% duration time to release emerging products after deploying the pre-trained KG-enhanced large model.

V. RELATED WORK

A. Existing Business KGs (not open-source or comprehensive)

As business KGs have been pivotal in providing various knowledge-based services to e-commerce platforms, there exists some business KGs at present, but they are not publicly available and comprehensive enough.

AmazonKG has shown several subsets (not open-source) in different ways: AmazonKG_TExtract [48] with a hierarchical taxonomy contains 2 million products from 4,000 categories without images. AmazonKG_AutoKnow [49] contains >30M products in 11K unique types and >1B triples, and is also single-modal. AmazonKG_PAsk [50] is multimodal and contains 61,308 samples covering 14 product categories.

Alibaba Group has also introduced some subset KGs: AliCoCo [17] is a concept KG containing 57,125 types of concepts and 2 types of relations at the time of writing. AliCoCo2 [18] enriches AliCoCo with commonsense knowledge and contains 163,460 types of concepts and 91 types of relations at writing time. AliMe KG [51] contains thousands of items. AliMe MKG [52] enriches AliMe KG with millions of images. These business KGs are all well constructed but not directly open-source to the public.

WalmartKG [10] is derived from grocery.walmart.com, containing almost 140,000 common grocery products with short textual descriptions excluding images, and not open-source. Atlas [53] is a multimodal product taxonomy dataset focusing on clothing products, containing thousands of product images along with their corresponding product titles. Atlas is publicly available but only involves one category – clothing, and consequently not comprehensive enough.
B. Business KG Construction Approaches

As datasets in business scenarios are large-scale and complex, developing a high quality business KG is still laborious and challenging. Some previous works have focused on business KG construction from different aspects.

Taxonomy Construction. [54] proposes to append new concepts into existing taxonomies with an adaptively self-supervised and user behavior-oriented product taxonomy expansion framework. [55] proposes a new paradigm that categorizes e-commerce products into a multi-level taxonomy tree with thousands of leaf categories based on machine translation. [56] formulates a new task of taxonomy completion by discovering both the hypernym and hyponym concepts for a query, and proposes a triplet matching network to resolve this task. [48] proposes to organize thousands of product categories in a hierarchical taxonomy with a taxonomy-aware knowledge extraction model named TXtract.

Attribute Value Extraction. [50] presents PAM model, a unified framework for multimodal attribute value extraction from product profiles. [57] develops a novel deep tagging model named OpenTag to discover missing values of product attributes by leveraging product profiles. [14] proposes AdaTag, utilizing adaptive decoding to handle multi-attribute value extraction from product profiles.

[49] takes both taxonomy construction and attribute value extraction into account, and describes an automatic (self-driving) system: AutoKnow, to automate ontology construction, knowledge enrichment and cleaning for massive products.

C. Business KG Pre-Training and Embedding for Applications

Business KG Pre-training. [58] contributes a large-scale multimodal pre-training dataset named M5Product with self-harmonized contrastive learning, which contains more than 6 million multimodal samples from 6,232 categories. [11] proposes a pre-trained KG model named PKGM for a billion-scale product KG (single-modal), to provide knowledge services for e-commerce tasks. [59] proposes K3M model to introduce knowledge modality in multimodal pre-training w.r.t. modal-interaction of images, texts and structured knowledge.

Business KG Embedding. [10] presents a self-attention-enhanced distributed KG embedding method with an efficient multi-task training schema, aiming at learning intrinsic product relations. [60] discusses some experiences of deploying KG embedding in e-commerce applications, and identifies the necessity of attentive reasoning, explanations and transferable rules in business scenarios. [61] proposes an explainable KG embedding method named XTransE for link prediction with lifestyles in e-commerce, and generates explanations for item-lifestyle prediction results.

VI. DISCUSSION AND CONCLUSION

A. Lessons Learned

In this paper, we demonstrate that constructing large-scale (billion) KG in the vertical domain is practicable and illustrate how to design ontology, build multimodal KG and manage its quality. However, our work has limitations as there are still many wrong or missing facts in KG due to the noisy raw data in applications; thus, more work should be considered for quality management and knowledge graph completion. Naturally, there are three lessons to be noted when building a business KG. (1) First, everything follows the business logic. Note that practical applications impose constraints on the links in KG by defining an ontology coupled with business logic. Building an accurate, unambiguous, easy-to-maintain, and scalable ontology not only requires developers to have a deep understanding of the business logic but also to have a specific prediction of possible future changes so that the design can be close to the status quo and efficient. (2) Second, practical and minimal. Since not all knowledge is necessary for downstream tasks and there exist many wrong facts in the raw corpus; thus, it is important to integrate valuable nodes (e.g., images are valuable for product alignment) rather than all information to KG. (3) Third, quality control is a lifelong work. There are many considerations around the integrity of new knowledge, checking quality, and correcting wrong facts. Moreover, it is critical to manage changes to the ontology without creating inconsistencies with the knowledge already in the system.

B. Conclusion

In this work, we construct and publish OpenBG, an open business knowledge graph derived from a deployed system. To the best of our knowledge, it is the most comprehensive business KG available on the Web. OpenBG is integrated with fine-grained taxonomy and multimodal facts and enables linking to a variety of external data. As an ongoing business KG, OpenBG currently contains more than 2.6 billion triples, with more than 88 million entities covering over 1 million core classes/concepts and 2,681 types of relations. We also release the OpenBG benchmark and run up an online competition based on it. We report experimental results of KG-centric tasks on the OpenBG benchmark and KG-enhanced tasks on full OpenBG, illustrating the potential advantages of multimodal structured knowledge in business scenarios. We release all the data resources for the community to promote future work.

In the future, we will keep developing and improving OpenBG, and hopefully, it can benefit KG construction in other vertical domains, not limited to business. Interesting future directions include: (1) apply OpenBG to other tasks, such as product recommendation, (2) explore the OpenBG for cross-discipline science of economics, economics, and sociology.

ACKNOWLEDGMENT

We would like to express gratitude to the anonymous reviewers for their kind and helpful comments. This work was supported by the National Natural Science Foundation of China (No.62206246, 91846204 and U19B2027), Zhejiang Provincial Natural Science Foundation of China (No. LGG22F030011), Ningbo Natural Science Foundation (2021J190), Yongjiang Talent Introduction Programme (2021A-156-G), and NUS-NCS Joint Laboratory (A-0008542-00-00).
types,” in KDD. ACM, 2020, pp. 2724–2734. [Online]. Available: https://doi.org/10.1145/3394486.3403323

[50] R. Lin, X. He, J. Feng, N. Zalmout, Y. Liang, L. Xiong, and X. L. Dong, “PAM: understanding product images in cross product category attribute extraction,” in KDD. ACM, 2021, pp. 3262–3270. [Online]. Available: https://doi.org/10.1145/3447548.3467164

[51] F. Li, H. Chen, G. Xu, T. Qiu, F. Ji, J. Zhang, and H. Chen, “Alime KG: domain knowledge graph construction and application in e-commerce,” in Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2020, pp. 2581–2588. [Online]. Available: https://doi.org/10.1145/3340531.3412685

[52] G. Xu, H. Chen, F. Li, F. Sun, Y. Shi, Z. Zeng, W. Zhou, Z. Zhao, and J. Zhang, “Alime MKG: A multi-modal knowledge graph for live-streaming e-commerce,” in CIKM. ACM, 2021, pp. 4808–4812. [Online]. Available: https://doi.org/10.1145/3459637.3481983

[53] Y. Umaashankar, G. S. S, and A. Prakash, “Atlas: A dataset and benchmark for e-commerce clothing product categorization,” in Proceedings of ACM SIGIR Workshop on eCommerce (SIGIR eCom’20), Virtual Event, China, July 30, 2020. ACM, 2020. [Online]. Available: https://sigir-eecom.github.io/ecom20Papers/paper9.pdf

[54] S. Cheng, Z. Gu, B. Liu, R. Xie, W. Wu, and Y. Xiao, “Learning what you need from what you did: Product taxonomy expansion with user behaviors supervision,” in 38th IEEE International Conference on Data Engineering, ICDE 2022, Kuala Lumpur, Malaysia, May 9-12, 2022. IEEE, 2022, pp. 3280–3293. [Online]. Available: https://doi.org/10.1109/ICDE53745.2022.00310

[55] L. Tan, M. Y. Li, and S. Kok, “E-commerce product categorization via machine translation,” ACM Trans. Manag. Inf. Syst., vol. 11, no. 3, pp. 11:1–11:14, 2020. [Online]. Available: https://doi.org/10.1145/3382189

[56] J. Zhang, X. Song, Y. Zeng, J. Chen, J. Shen, Y. Mao, and L. Li, “Taxonomy completion via triplet matching network,” in Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021. AAAI Press, 2021, pp. 4662–4670. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16596

[57] G. Zheng, S. Mukherjee, X. L. Dong, and F. Li, “Opentag: Open attribute value extraction from product profiles,” in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018. F. Guo and F. Farooq, Eds. ACM, 2018, pp. 1049–1058. [Online]. Available: https://doi.org/10.1145/3219819.3219839

[58] X. Dong, X. Zhan, Y. Wu, Y. Wei, M. C. Kampffmeyer, X. Wei, M. Lu, Y. Wang, and X. Liang, “M5product: Self-harmonized contrastive learning for e-commercial multi-modal pretraining,” in IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022. IEEE, 2022, pp. 21 220–21 230. [Online]. Available: https://doi.org/10.1109/CVPR52688.2022.02057

[59] Y. Zhu, H. Zhao, W. Zhang, G. Ye, H. Chen, N. Zhang, and H. Chen, “Knowledge perceived multi-modal pretraining in e-commerce,” in MM ’21: ACM Multimedia Conference, Virtual Event, China, October 20 - 24, 2021, 2021, pp. 2744–2752. [Online]. Available: https://doi.org/10.1145/3474085.3475648

[60] W. Zhang, S. Deng, M. Chen, L. Wang, Q. Chen, F. Xiong, X. Liu, and H. Chen, “Knowledge graph embedding in e-commerce applications: Attentive reasoning, explanations, and transferable rules,” in IJCKG’21: The 10th International Joint Conference on Knowledge Graphs, Virtual Event, Thailand, December 8 - 10, 2021. 2021, pp. 71–79. [Online]. Available: https://doi.org/10.1109/IS02223.3502232

[61] W. Zhang, S. Deng, H. Wang, Q. Chen, W. Zhang, and H. Chen, “Xtranse: Explainable knowledge graph embedding for link prediction with lifestyles in e-commerce,” in Semantic Technology - 9th Joint International Conference, JIST 2019, Hangzhou, China, November 25-27, 2019. Revised Selected Papers, 2019, pp. 78–87. [Online]. Available: https://doi.org/10.1007/978-981-15-3412@8