Xiaochen Wang. An Error Analysis on Representation Learning of Knowledge Graph. A Master's Paper for the M.S. in I.S degree. April, 2022. 31 pages Advisor: Yue Wang

Knowledge graph is a knowledge base containing integrated data in a graph-structure. Prior knowledge included in knowledge graph can make up for the insufficient reasoning ability of statistical machine learning methods. By utilizing representation of knowledge, researchers and information practitioners are capable of deepening algorithms' understanding of the real world by introducing plentiful common sense. However, a problem confusing researcher for a long period is that computers have difficulty in comprehending knowledge stored in knowledge graph, preventing the efficient usage of these graph-structured information. This study aims at presenting a comprehensive analysis on the knowledge representation generated by multiple methods. This analysis may inspire readers to reflect characteristics, advantages and disadvantages of knowledge representation learning models

Headings:

Knowledge Representation

Machine Learning

Artificial Intelligence

Data Mining

AN ERROR ANALYSIS ON REPRESENTATION LEARNING OF KNOWLEDGE GRAPH

by
Xiaochen Wang

A Master's paper submitted to the faculty
of the School of Information and Library Science
of the University of North Carolina at Chapel Hill
in partial fulfillment of the requirements
for the degree of Master of Science in
Information Science.

Chapel Hill, North Carolina

April 2022

Approved by

Yue Wang

Table of Contents

Contents

INTRODUCTION

For computer scientist and information practitioners, machine learning and other similar artificial intelligence methods have been proven to be effective for machine to perceive, understanding, reasoning, and predicting what happened and will happen in the real world. For instance, in the field of Natural Language Processing, a pre-trained deep learning model can learn linguistic information from exclusive corpora and complete downstream tasks such as Information Extraction, Reasoning, Question Answering and Machine Translation. However, these existing metrics are mostly based on statistical assumption, transform raw text or digital images into numerical data and usually does not leverage prior knowledge. A machine learning model can only obtain knowledge encoded within multiple datasets, which highly limits its capability in interpreting data flexibly. People often find text generation robots return ridiculous synthesized sentences, or have difficulties in understanding humors, which manifests the consequence of absence of prior knowledge and common sense.

Multiple methods have been developed in recent years to solve this problem. Knowledge representation is one of them that combine both prior knowledge and statistical information. It can make use of existing knowledge resources, encoding knowledge to forms that can be understood by machines, and enhance performances of current machine learning model. Its usage become prevalent in the domain of computer science as a necessary assisting tool to boost machine's intelligent degree.

Exhaustive experiments have been conducted in this field to examine the effectiveness of multiple methods. While several numerical evaluation metrics are applied, the evaluation process gradually becomes a test on black box: specialists in this domain become obsessed with the increasing number and show neglect interests in the specific results returned by their models. Admittedly, the emphasis on the evaluation metrics are understandable considering the nature of this task, however, the absence of analysis on specific cases may decelerate the development of representation learning of knowledge graph. As knowledge graph contains intuitive information that can be easily interpreted by human beings, the analysis on its representation learning cannot be time-consuming and will possibly insight practitioners as well. Therefore, this work focuses on the in-depth error analysis on knowledge graph representation learning, which has not been conducted in this domain.

LITERATURE REVIEW

Researches concerning knowledge representation have drawn more attentions in recent years, especially those published in top-tier conferences and journals. State-of-the-art techniques and perspectives are exclusively mentioned in these publications, on which novel research can be based and improvement may be made.

Overview of Knowledge Graph: How Should It Be Used?

Knowledge Graph is a concept came up with by Google in 2012, rapidly becoming ubiquitously used in several domains. It can be interpreted as an encoding formalism representing so-called real-world objects and relations in form of graph (Fensel et al., 2020). Considering computer's natural flaw in processing unstructured data and logical relations, knowledge graph can be a perfect medium to convey abstract concepts and correspondences to computer.

The knowledge representation learning is based on knowledge graphs and figuring out which knowledge graphs should be utilized is an initial step of a work specializing in this field. There are plenty of knowledge graphs for diverse domains (Lin et al., 2020). Language knowledge graph, commonsense knowledge graph and encyclopedic knowledge graph have different kinds of relationships, designing logic, structure and other properties serving for their domain characteristics. Understanding the inner composition of a knowledge graph may benefit analytical and practical works concerning it. Newly developed knowledge graph(Ilievski et al., 2020) in large scale can be useful in the knowledge representation research to encode more exhaustive information and output more generalizable representation results. A typical perspective of research regarding knowledge graph is that it demands a huge amount of manual work, which is not appropriate to be an individual research topic. However, researches on knowledge graph can be divided into two categories: application and construction (Zou et al., 2020). These two stages are relatively independent, hence, the application-relevant researches of knowledge graph can benefit from the construction work mentioned above without participating in the upper-stream works that are usually time-consuming and labor intensive.

Techniques Used in Representation Learning

Advances in hardware in the first decade of 21st century resulted in the boosting of deep learning and further triggered the revolution in artificial intelligence. Since the inception of deep learning in 2012, this novel technique has been widely applied in every scientific domain, including information science and its subbranch, knowledge representation. In 2013, a revolutionary knowledge representation learning method called TransE impressed most of researchers in this field by its innovation of modifying the loss function of relation-oriented deep learning model (Bordes et al., 2013). A relation represented as a directed arrow in the knowledge graph can be further translated into embedding space as a triple(head, label, tail). Optimizing the objective function:

$$f_r(h,t) = |\mathrm{h} + \mathrm{r} - \mathrm{t}|_2^2$$

the deep neural network is capable of learning more contextual information of entities, as well as understanding meanings of abstract relation according to the two entities it connects.

However, a shortage of this model is that a relation can only have one representation in the embedding space, which means that its objective function cannot perfectly reflect the 1-m or m-n relations between entities and may get model confused. As an improved version of it, a well-known variant of TransE was designed to simulate the complicated nature of relations in the knowledge graph, which was called TransH(Wang et al., 2014). The novel objective function they designed regarding this problem was:

$$f_r(h,t) = |h_{\perp} + \mathrm{r} - t_{\perp}|_2^2$$

in which entities were representated as vectors in hyperplane, and projection was used in the objective function to reflect the m-n relation between entities. By a modeling in the hyperplane instead of the normal embedding space that TransE used, TransH exceed all the previous works in terms of accuracy. Additionally, authors of TransE considered the fact that knowledge graphs were far from completed as most of them are domain-specific or limited in terms of scale, and further designed a false negative labeling method to cope with this problem. Besides sophisticated mathematical model, this work is inspiring as it paid attention on the characteristics of datasets themselves, which is an idea that can be borrowed in most of the quantitative researches. It can not only arouse novel thoughts but also help to review the quality of conclusion drawn before.

TransR is another variant of TransE focusing on solving the 1-m and m-n relation in the representation of knowledge graph. Instead of utilizing the concept of hyperplane, this work used a trick: relation and entity were represented in different embedding spaces, and a projection between these two spaces was applied to achieve to ensure the unity between them. The equation set

$$f_r(h,t) = |h + \mathrm{r} - t|_2^2$$

$$h_r = hM_r$$

$$t_r = tM_r$$

provided a brief explanation of how projections(M matrices above) help to shape these two independent embedding space. Experiments covered in this publication shows that this innovation can result in an apparent enhancement in multiple downstream tasks such as link prediction and triple classification. These three models mentioned above have a very similar design pattern: modeling knowledge graphs - figure out the goal of optimization - modify the objective function. Actually, there are a variety of directions that can be explored to improve the performance of a representation learning model, however, the most difficult but effective one is to dig into the bottom part of mode and modify basic operations contained in it.

As what can be observed in these three articles, resolving the ambiguity of relation and entity is a major difficulty that researchers should ponder in the knowledge representation study. Besides works concerning objective function and modeling mentioned above, additional information can also be utilized to identify the accurate meaning of a mention. For instance, given a triple containing two entities and one relation, a model may perceive their correlation in multiple ways. However, if the context and description of entities can be complemented, it will be much easier for deep neural network to comprehend meanings of entities and relation more precisely(An et al, 2018). Using existing tools designed for entity linking (Wang et al., 2016), a mention can be linked to a predefined entity in ontology, and description can be retrieved as additional features for neural networks. By encoding this kind of contextual information to sequential neural networks like Bidirectional Long Short Term Memory(Huang et al, 2015) or Attention Mechanism(Vaswani et al., 2017), problem of entity ambiguity can be alleviated more or less.

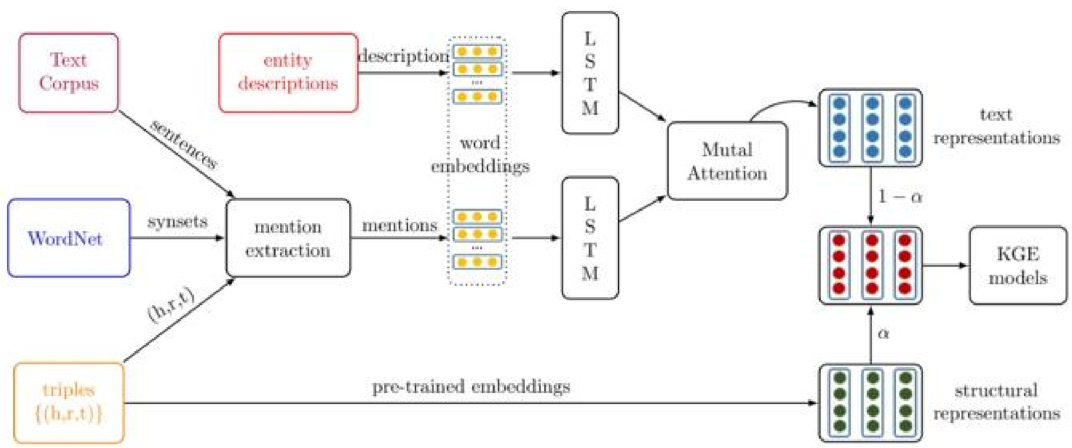

Fig. 1 Accurate Text-Enhanced Knowledge Graph Representation Learning Framework (An et al, 2018)

BERT(Devlin et al., 2018) is a millstone in the natural language processing area. Its capability of capturing contextual information and encoding it to adaptive word embedding makes it popular in every domain concerning linguistic application. It seems to be difficult to utilize this kind of sequence-based model in the domain of knowledge representation learning since there is no semantic continuity within each path in knowledge graphs. However, it can also be used in some alternative ways. K-BERT (Liu et al., 2019) is a variant specially designed for combining semantic information and prior knowledge stored in knowledge graphs. Instead of merely using sentences in corpora, researchers expanded information given by entities using domain-specific knowledge graph. For the purposing of avoiding that model confuses between information composed in sentences and knowledge graph, these two parts were encoded in different ways, making sure that the Mask-self-attention block (basic unit of BERT) can only make prediction based on semantic information, and prior knowledge was exploited in an assisting role. Another excellent feature of this model is that it can easily inject domain knowledge by equipped with a knowledge graph without pre-training process, which demands intensive computing resources that cannot be afforded by individual researchers. Its flexibility makes it a great tool to be considered in knowledge representation research.

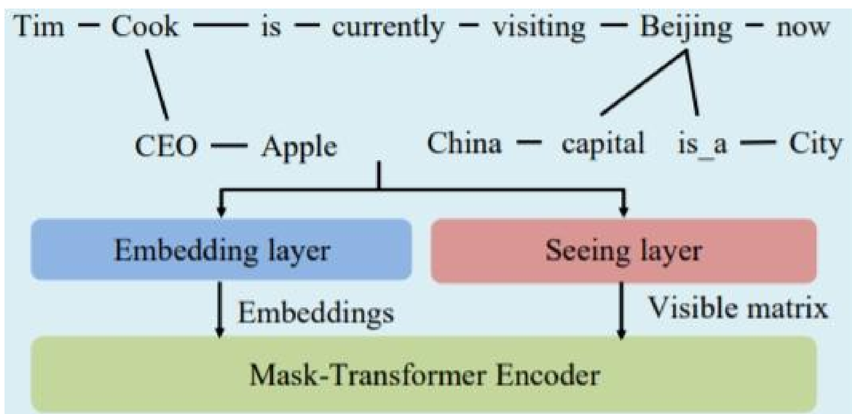

Fig 2. An instance of how K-BERT works

Although deep learning method is proven to be effective, extensive datasets, intensive computing resources and time consumption are needed in the application of this kind of heavyweight model. In more practical perspective, shallow machine learning model is also worth attention under some circumstances as it can be rapidly iterated and upgraded. Compared with deep neutral networks focusing on non-linear transformation, these shallow models always assume the linear relation between features and labels, reducing the cost of model as much as possible. FastText (Joulin et al., 2016) is a widely accepted open-sourced library in the natural language processing area, enabling users to train their own word embedding on multiple corpora. Based on it, researchers developed a knowledge graph-oriented variant (Joulin et al., 2017), aiming at obtaining knowledge representation under strict constraints of time and computing resources. According to their experiment results on knowledge graph completion and question answering tasks, their model was able to achieve competitive performances compared with deep learning models such as TransE and TransH. Its success indicates that deep learning is not the only solution of knowledge representation research, especially given the physical limitation in terms of experiments.

Downstream Works Concerning KG

For evaluation of quality of knowledge graph representation, various downstream tasks are taken into consideration by researchers. A better representation of knowledge graph usually implies a deeper understanding of knowledge in the real world, and is believed to enhance performance of models using it in relevant fields. Comparison between effectiveness of models before and after leveraging new representation is a general metric to examine whether the modification on representation learning model succeeds. By reviewing articles discussing representation's application in downstream tasks, one can get a clearer understanding of what a evaluation stage of representation learning is supposed to be.

Relation extraction is a general task in the field of information extraction, aiming at identity type and location of relation within raw text. It demands not only the representation of semantics that can be learned via unsupervised learning solely, but also the deep understanding of logic between entities and their relative position in the knowledge graph. These characteristics of this task makes it a perfect proving ground for examining quality of knowledge graph representation (Zhang et al., 2019).

In recent years, Question Answering (QA) has become the focus in the domain of natural language processing due to its wide application and technical difficulty. Artificial intelligence models are required to give reasonable feedback for specific questions, while a variety of linguistic phenomenon and causal inferences are demanded in the process. Mechanical memorizing that was widely used in statistical machine learning no longer work well in this task considering its requirement of moderate intelligence level. Instead, prior knowledge should be prioritized as it is the base of human beings' logical pattern and enable machine to simulate this process as well. Hence, knowledge graph has been generally used in this task, and the quality of its representation can also be embodied while evaluating the answers given by machine learning models(Zhang el al., 2018, Huang el al., 2019).

In addition to natural language understanding tasks mentioned above, natural language generation is another direction that can be explored to evaluate the quality of knowledge graph representation. Writing professional report requires authors to have abundant knowledge in the relevant field, making this work difficult for laypeople to undertake. It also happens to artificial intelligence models: algorithms are usually capable of generating summary in general field like news, however, creating a medical report can be much more tough compared with that. By utilizing knowledge graph representation and introducing external knowledge, models can be equipped with basic epistemology enabling them to generate report in specific domains. Recent work concerning Covid-19 report generation (Huang et al, 2019) gave an instance of combining domain-specific knowledge graph representation with Question Answering structure to generate a technical report. However, compared to other downstream tasks used as evaluation metric, text generation results do not have groundtruth that can be conveniently utilized to evaluate quality of models' output, which means that manual scoring may be needed for measuring the effectiveness of introducing new representation methods. Regarding this fact, it does not seems to be an appropriate downstream task for individual researchers focusing on knowledge graph representation learning if there is any other alternative.

DATA

Data Resources

In this study, two datasets prevalent in the domain of knowledge representation are utilized: WN11 and FB13. WN11 is an old version of WordNet, which is a large lexical database of more than language containing nouns, verbs, adjectives, and adverbs. Encoded relations are included in WordNet to indicates relations between these words. In this study, only the English part of this dataset is used for the knowledge representation learning. FB13(FreeBase13) is a knowledge representation dataset extracted from FreeBase, a large collaborative knowledge base whose data are provided by a wide range of communities. By using so-called entity-relationship model, this knowledge base encodes real-world knowledge in form of triples.

Although newer versions of these two datasets become available, only old version are used in this study. Even though FB15k and WN18 are more popular among the whole representation learning community due to their scale, these versions do not provide a translation from entity to its corresponding ids. In other words, since these datasets are overly well-packaged, it becomes impossible for researchers to truly understand what their model outputs. IDs without any textual description are meaningless in terms of error analysis, therefore, older versions are adopted in this study due to their better interpretability.

Data Exploration

Triple is the basic unit in these datasets. A triple representing knowledge should include the following items: head, relation, and tail (h, r, t). As the real-world knowledge exists in a very complex form, forms of mapping relations between head entity and tail entity include 1-1, 1-n, n-1, n-n. In this study, all these relations are covered for comprehensive investigation. By combining these triples, a directed knowledge network can be well-constructed, in which each node represents an entity, while relation serves as edge between these nodes.

Statistics of these two datasets show their compositions, revealing the scale of them as well:

	Number of Entity Types	Number of Relation Type
WN11	38194	11
FB13	75043	13

Table 1. Statistics of Two Datasets

An apparent observation is that compared with the types of entity, types of relation is in a very limited amount. In the representation learning process, each relation will be described using multiple entities, which may cause potential problems.

Data Sampling

Datasets in the domain of knowledge representation learning contain several valid triples indicating connection between real-world entities. However, invalid cases are also demanded in the model training process, as only models fed with both negative and positive samples can learn an appropriate way to represent knowledge. As these negative (invalid) cases do not exist in the dataset or the real-world, they should be artificially synthesized before the training process.

A popular solution of the lack of negative samples is to abandon one of items in a triple, and randomly select another relation/entity to replace the dropped one. By adopting this procedure, synthesized triples are relatively similar with the real ones, while they deliver fake information and are labeled as negative. In the practical process, the default setting in the OpenKE library is adopted, in which only the entities are swapped, while relations are always kept. The ratio between positive and synthesized negative cases is set to 1 by default as well, which provide a uniform distribution of data and may be beneficial for training a better model.

METHOD

Model Selection and Tasks

For the purpose of comparison, two knowledge representation learning methods, TransE and TransH, are adopted in this study. OpenKE is a framework for knowledge embedding implemented in PyTorch, in which well-investigated models like TransE, TransH and RotatE are available. To avoid the reinvention of wheels, I utilize OpenKE as the base of the analysis, and make some necessary modifications given the requirement of error analysis task.

Instead of some application scenarios mentioned in the literature review that can be used to evaluate the knowledge embedding, OpenKE provides two options applicable for examining the quality of knowledge representation. The first one is triple prediction: given a triple in form of (head, relation, tail), models will assign it with a score, and a threshold is set to make a binary division to judge whether the triple is valid or randomly synthesized. Compared with triple prediction which is hardly meaningful in the application of knowledge representation, the second one, entity predication, corresponds to how people understand and utilize knowledge representation well. Triples are presented to the model in form like ([MASK], relation, tail) or (head, relation, [MASK]), then models are required to find the most matching entity given the rest part of information in this triple. A correct prediction indicates that the model can successfully understand connect between abstract and tangible things in the real world. In this study, I adopt the tail prediction as the specific approach of examine the quality of representation.

In the original OpenKE implementation, models are evaluated using the following metrics: MR (Mean Rank), MRR (Mean Reciprocal Rank), Hit@10, Hit@3 and Hit@1. As there are thousands of entities in the knowledge base, it is too harsh to require the model to make the correct selection, therefore, Hit@10 and Hit@3 are applied to loose the constraint and acknowledge the performance of models if they assign a higher rank to the correct answer. These metrics provide intuitive way to measure effectiveness of models, however, one can never understand information delivered by the detailed output returned by these models though simply reviewing these numbers. Therefore, it is necessary to design an interface to present the output.

Interface Implementation and Output Formatting

Even though OpenKE is mainly implemented using Python language, its basic functions concerning embedding extraction, result generation and ranking are fully written in C++ for the purpose of accelerating the execution. The mixed usage of both languages and well-packaged characteristic of this library makes it difficult for people who are interested in the error analysis to obtain specific cases. This may be the reason why the error analysis on knowledge representation learning is extremely rare. In this study, some modifications on basic codes of these models are conducted, allowing users to intuitively view, understand, and analyze output of these models.

For the triple classification task, models' output is in form of three-columns tables, in which every triple will hold a score representing the quality of its representation, and binary label indicating whether it is a synthesized triple or a real one.

	Triple	Score	Label
1	(__chamaecyparis_lawsoniana_1, _type_of, __cedar_1)	12.68621	1
2	(__roman_empire_1, _domain_topic, __athrotaxis_1)	16.724201	0
……			

Table 2. Output in the Triple Classification Task

In the entity prediction task, there is no classifier assigning score to each triple, instead, all the possible entities will be ranked, and position of the correct answer in the ranked list will be used as the measurement of quality of the representation. In this task, only real triples are demanded, and all the synthesized negative samples will not be utilized.

	Triple	Ranks
1	(__chamaecyparis_lawsoniana_1, _type_of, __cedar_1)	1
2	('__medical_aid_1, _type_of, __irrigation_2)	5896
……		

Table 3. Output in the Tail Prediction Task

These two output presenting forms allow us to conduct error analysis case by case and investigate the reason why model perform ideally or unexpectedly. In the following section, reasoning based on models' output will be given along with some representative cases, from which better understanding of knowledge representation models can be obtained.

Statistics and Error Analysis for FB13

Before digging into specific cases, some general statistical works can be conducted to help us obtain a more comprehensive impression of what the output looks like, as well as what problem we are supposed to cope with in the error analysis.

Although evaluation metrics available in the OpenKE library provide us a general view of performance of models, it is still difficult to figure out how models perform in representing different type of relations and entities. Do they show advantage in predicting a specific type of entity? Do they manifest incapability in coping with a type of relation? Stats of detailed output in the tail prediction task can give us this kind of information as the complementary of the original evaluation.

For the FB13 dataset, models' capability of representing relations differs apparently. The following table shows the average rank of triples grouped by their relation type, as well as their cohort size in the training set that may influence the representation learning process.

Type of Relation	Ave_rank (TransE)	Ave_rank (TransH)	Number of Cases in Training Set
cause_of_death	119.83	131.71	10816
ethnicity	50.37	46.27	4698
gender	0.23	0.39	59423
institution	207.64	177.62	15566
nationality	21.36	14.97	47581
profession	131.63	128.12	45931

religion	29.00	37.61	8112

Table 4. Evaluation Table for FB13

An obvious phenomenon that we can observe is that *gender* is an extremely easy relation for the tail prediction model to give the correct answer, seemingly manifesting the high quality of the representation of *gender.* How could this happen?

A simple investigation of the dataset may be useful to answer this question: in the training set, there are only two tails for the relation *gender (A's gender is B)*, male and female. Therefore, as these two words are clustered well in the feature space, the embedding of their corresponding relation is also well-described. According to the definition loss function, if the distribution of head is fixed, then the better tail embeddings are, the better relation representation is.

However, there are still some cases of this relation worth analyzing. Even though correct answers of most cases are ranked as 0 (predicted as correct gender) or 1 (predicted as the opposite gender), two outliers still have their correct answer assigned with rank 2 (predicted as entity other than binary gender) by both TransE and TransH. These two instances are ('hatshepsut', 'gender', 'female') and ('mikolaj_radziwill ', 'gender', 'male'). By browsing the training set, it is checked that *Hatshepsut*, the ancient female pharaoh, only has connections with her husband *Thutmose I* and her son *Thutmose II*, while Polish-Lithuanian noble *Mikołaj Radziwiłł* is merely connected to his son and father as well. Their isolation may be helpful to illustrate why they are not mapped to any gender: if the representation of character is in low-quality since it cannot be located well based on its neighbors in the network, then its corresponding tail will also shift in the embedding space, despite that the relation is well-learned.

According to Table 3, *institution* (A is affiliated to the institution B) is the most difficult relation for the model to tackle. One reason may be that compared with *gender* whose corresponding tails are very limited in number, there are hundreds of institutions, which set barrier for the model to learn an accurate embedding for this relation. Compared with two unpopular ancient historical characters mentioned above, the life of Pakistan politician *Huseyn Shaheed Suhrawardy* is much clearer. His religion, gender, nationality, active location and place of birth and death are exhaustively described in the training set. However, his corresponding triple ('huseyn_shaheed_suhrawardy', 'institution', 'grays_inn') gets failed in the tail prediction, which may be attribute to his institution *Gray's Inn* only has connections with four of its alumni. Here, the low-quality of tail's embedding rather than head's negatively influences the knowledge representation, and its co-occurrence with the ambiguity of relation makes the real-world knowledge even more misty.

Statistics and Error Analysis for WN11

Compared with FB13, entities in WN11 are connected in a more complex pattern, which reduces the performance of knowledge representation model. Table 4 shows how TransE and TransH perform in the tail prediction task. It is notable that the total amount of entities in WN11 is only the half of the number of entities in FB13, therefore, these larger number indicate that WN11 is a much trickier knowledge base in terms of embedding extraction.

Type of Relation	Ave_rank (TransE)	Ave_rank (TransH)	Number of Cases in Training Set
domain_region	11058.48	13226.49	4227
domain_topic	10714.03	13467.74	1099
has_instance	14601.27	16847.92	36178
has_part	10664.64	12654.96	6139
member_holonym	3065.52	3821.50	9146
member_meronym	12650.26	16560.54	9223
part_of	4370.12	5027.33	6600
similar_to	19345.04	21419.66	1659
subordinate_instance_of	987.32	592.97	3778
synset_domain_topic	693.07	793.11	3976
type_of	3045.77	3302.85	30556

Table 5. Evaluation Table for WN11

Intuitively, the relation *similar_to* should not be difficult for learning and representing, however, Table 4 shows that the quality of its representation is not satisfying. After checking low-score triples containing this relation, it is confirmed that it is not isolated entities triggering this type of failure, as entities in these low-score triples are connected to around 5 neighbors in average. Therefore, other explanation should be investigated.

Checking the training set provides a novel insight into this kind of error. Different from other relations, *similar_to* is reversible. For instance, as (multiple_2, similar_to, double_4) exists in the training set, (double_4, similar_to, multiple_2) can also be found. Furthermore, it is observed that each triple containing *similar_to* in the WN11 training set has its corresponding reversible triple. Assume that we only want to train a TransE model using two triples mentioned above, an equation set based on the loss function is:

$$f_r(\text{multiple\_2}, \text{double\_4}) = |\mathrm{e}_{\text{multiple\_2}} + \mathrm{r}_{\text{similar\_to}} - \mathrm{e}_{\text{double\_4}}|_2^2$$

$$f_r(\text{double\_4}, \text{multiple\_2}) = |\mathrm{e}_{\text{double\_4}} + \mathrm{r}_{\text{similar\_to}} - \mathrm{e}_{\text{multiple\_2}}|_2^2$$

in which *e* is the embedding of entities, and *r* is the embedding of relation. As we would like to minimize the summation of these two losses, an intuitive optimal solution is to let

$$\mathrm{e}_{\text{double\_4}} = \mathrm{e}_{\text{multiple\_2}}$$

$$\mathrm{r}_{\text{similar\_to}} = 0$$

These equations show that if a relation is reversible, then its representation inclines to be zero-vector, while the model will intend to set two entities linked by it as close as possible. However, since these entities are well-connected in this knowledge network, it is impossible to assign them with the similar representation. So, an antinomy between the optimal objective of these equations and the whole network appears, hinting why model may get clumsy while dealing with reversible relations like *similar_to*.

Negative Samples: Expected Counterfeit

The triple classification task allows us to investigate how models treats real and synthesized samples. In the OpenKE Implementation, developers show that a lower score indicates that models are more certain about the correct answer. An expected observation is that synthesized triples should be assigned with lower scores compared with the real ones, while the lowness should be moderate to ensure that representation models can also learn from these synthesized samples. According to the statistics of triple classification results, synthesized samples get higher scores than real samples in average, revealing that real samples make more senses for all the datasets and methods, without any exceptions.

	WN11-TransE	WN11-TransH	FB13-TransE	FB13-TransH
Real Samples	16.33	17.04	13.88	14.29
Synthesized Samples	18.10	18.43	17.38	17.99

Table 6. Triple Classification for Real and Synthesized Samples

However, there are still some clues worth notation. Comparison between what presented in Table 3 and Table 4 shows that models perform better in the FB13 datasets rather than WN11. Are there any relation between the better performance and minimal differences between real and synthesized samples? It may be a direction of the future work. Previous work shows the potential of enhancing models' representing ability by improving the quality of negative samples [Wang et al., 2018], and a further error analysis may be conducted to investigate the detailed relation between ramdomly synthesized samples and those generated using adversarial neural networks.

CONCLUSION AND REFLECTION

In this study, an investigation based on knowledge representation learning library OpenKE is conducted. Multiple datasets (WN11, FB13) and models (TransE, TransH) are adopted for the comprehensiveness of the study. By presenting results in the tail prediction and triple classification task, model's output gets fully observed and analyzed. Specific cases manifesting models' traits and pattern of datasets are fully discussed. Concerning that there is no previous literature specifically focusing on the error analysis in this field, this work can be regarded as a pioneering study and may inspire researchers who are interested in profoundly understanding working pattern of knowledge representation models. Also, this work may be beneficial to those who aim at enhancing the interpretability of models as well as deconstructing the black box.

Future work based on this study may concentrate on the analysis on more complicated models, such as K-BERT and other transformer-based deep neural networks. As same patterns may be observed, phenomenon appearing due to the deeper and more complicated structure of neural network may also inspire novel reflection. Also, the investigation in the transform-based models may potentially solve the reversible relation issue mentioned in the analysis on WN11 dataset, since these models apply attention-mechanism and possess the capability of adjusting embeddings according to context (entities, in this scenario). As what discussed in the section of negative samples, case study on synthesized samples can also explicitly deepen researchers' understanding of how models work as well as bring unconventional innovation in this domain.

A reflection that I obtain during this study is that the dataset determinedly shape what we can expect from a machine learning model. Previously, my research only focuses on how the distribution of data effects the performance of model, however, while investigating the *similar_to* relation of the WN11 dataset, I find that there are multiple factors other than distribution that can be used to describe or measure a dataset. What is the data structure of a dataset? What is the ambiguity level of its language? What is the abstraction of entities contained in the dataset? How is the dataset collected, annotated, synthesized, or sampled? All these factors influence the final output of models along with models themselves. In an era that everyone never feel tired of discussing models, maybe paying more attention to what models learn can also benefit our application of artificial intelligence dramatically.

BIBLIOGRAPHY

An, B., Chen, B., Han, X., & Sun, L. (2018, June). Accurate text-enhanced knowledge graph representation learning. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (pp. 745-755).

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. Advances in neural information processing systems, 26.

Cunningham, P., Cord, M., & Delany, S. J. (2008). Supervised learning. In Machine learning techniques for multimedia (pp. 21-49). Springer, Berlin, Heidelberg.

Davis, R., Shrobe, H., & Szolovits, P. (1993). What is a knowledge representation?. AI magazine, 14(1), 17-17.

Fensel, D., Şimşek, U., Angele, K., Huaman, E., Kärle, E., Panasiuk, O., ... & Wahler, A. (2020). Introduction: what is a knowledge graph?. In Knowledge Graphs (pp. 1-10). Springer, Cham.

Ghahramani, Z. (2003, February). Unsupervised learning. In Summer School on Machine Learning (pp. 72-112). Springer, Berlin, Heidelberg.

Gidaris, S., Singh, P., & Komodakis, N. (2018). Unsupervised representation learning by predicting image rotations. arXiv preprint arXiv:1803.07728.

Grover, A., & Leskovec, J. (2016, August). node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 855-864).

Huang, X., Zhang, J., Li, D., & Li, P. (2019, January). Knowledge graph embedding based question answering. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (pp. 105-113).

Huang, Z., Xu, W., & Yu, K. (2015). Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint arXiv:1508.01991.

Ilievski, F., Szekely, P., & Zhang, B. (2021, June). Cskg: The commonsense knowledge graph. In European Semantic Web Conference (pp. 680-696). Springer, Cham.

Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759.

Joulin, A., Grave, E., Bojanowski, P., Nickel, M., & Mikolov, T. (2017). Fast linear model for knowledge graph embeddings. arXiv preprint arXiv:1710.10881.

Lafferty, J., McCallum, A., & Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Lin, J., Zhao, Y., Huang, W., Liu, C., & Pu, H. (2021). Domain knowledge graph-based research progress of knowledge representation. Neural Computing and Applications, 33(2), 681-690.

Lin, Y., Liu, Z., Sun, M., Liu, Y., & Zhu, X. (2015, February). Learning entity and relation embeddings for knowledge graph completion. In Twenty-ninth AAAI conference on artificial intelligence.

Liu, W., Zhou, P., Zhao, Z., Wang, Z., Ju, Q., Deng, H., & Wang, P. (2020, April). K-bert: Enabling language representation with knowledge graph. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 03, pp. 2901-2908).

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

Tiwari, P., Zhu, H., & Pandey, H. M. (2021). DAPath: Distance-aware knowledge graph reasoning based on deep reinforcement learning. Neural Networks, 135, 1-12.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

Wang, P., Li, S., & Pan, R. (2018, April). Incorporating gan for negative sampling in knowledge representation learning. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 32, No. 1).

Wang, Z., Li, J., Liu, Z., & Tang, J. (2016). Text-enhanced representation learning for knowledge graph. In Proceedings of International Joint Conference on Artificial Intelligent (IJCAI) (pp. 4-17).

Wang, Z., Zhang, J., Feng, J., & Chen, Z. (2014, June). Knowledge graph embedding by translating on hyperplanes. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 28, No. 1).

Wang, Q., Li, M., Wang, X., Parulian, N., Han, G., Ma, J., ... & Onyshkevych, B. (2020). COVID-19 literature knowledge graph construction and drug repurposing report generation. arXiv preprint arXiv:2007.00576.

Zhang, N., Deng, S., Sun, Z., Wang, G., Chen, X., Zhang, W., & Chen, H. (2019). Long-tail relation extraction via knowledge graph embeddings and graph convolution networks. arXiv preprint arXiv:1903.01306.

Zhang, Y., Dai, H., Kozareva, Z., Smola, A. J., & Song, L. (2018, April). Variational reasoning for question answering with knowledge graph. In Thirty-Second AAAI Conference on Artificial Intelligence.

Zou, X. (2020, March). A survey on application of knowledge graph. In Journal of Physics: Conference Series (Vol. 1487, No. 1, p. 012016). IOP Publishing.