An Analysis of Knowledge Representation for Anime Recommendation Using Graph Neural Networks

Yuki Saito[1], Shusaku Egami[2,1][a], Yuichi Sei[1][b], Yasuyuki Tahara[1][c] and Akihiko Ohsuga[1][d]

[1]*Department of Informatics, University of Electro-Communications 1–5–1 Chofugaoka, Chofu, Tokyo, 182–8585, Japan*

[2]*National Institute of Advanced Industrial Science and Technology (AIST), 2–4–7 Aomi, Koto-ku, Tokyo, 135–0064, Japan*

Keywords: Knowledge Graph, Graph Neural Networks, Recommender System.

Abstract: In recent years, entertainment content, such as movies, music, and anime, has been gaining attention due to the stay-at-home demand caused by the expansion of COVID-19. In the content domain, research in the field of knowledge representation is primarily concerned with accurately describing metadata. Therefore, different knowledge representations are required for applications in downstream tasks. In this study, we aim to clarify effective knowledge representation through a case study of recommending anime works. Thus, we hypothesized how to represent anime works knowledge to improve recommendation performance from both quantitative and qualitative aspects and verified the hypotheses by changing the knowledge representation structure according to the hypothesis. Initially, we collected data about anime works from multiple data sources and integrated them to construct a knowledge graph (KG). We also prepared several KGs by varying the knowledge configuration. Subsequently, we compared the recommendation performance of each KG as an input to the graph neural networks. As a result, it was found that the amount of semantic relationships was proportional to the recommendation performance and that the properties that can characterize the work contributed to the recommendation.

1 INTRODUCTION

The COVID-19 pandemic has caused people to spend more time at home, such as telecommuting and distance learning. Accordingly, entertainment content including movies and music has gained significant attention due to the popularization of subscription-based services. For instance, in 2020, Netflix acquired approximately 37 million new subscribers worldwide (Soldo, Lana and Schagerl, Christopher, 2023). Among much research in the content domain, there is the field of knowledge representation, which covers the construction of systematic databases based on content metadata. Knowledge Graph (KG) is a core technology in the field of knowledge representation and can represent the relationships among various types of knowledge in a graph structure. In a KG, knowledge is represented as the structure of triplets: subject, predicate, and object. For example, the fact that Princess Mononoke is directed by Hayao Miyazaki can be represented as a triplet "*Princess_Mononoke, directed_by, Hayao_Miyazaki*" to provide the knowledge that humans understand conceptually in a machine-readable format. The standard format for sharing such representations on the Web is called Resource Description Framework (RDF); N-Triples and Turtle are used as notations for the practical use of RDF. Linked Data (Bizer et al., 2009) is a KG linked to other KGs using semantic web technologies such as RDF.

Existing studies on KGs for content primarily focus on conceptually accurate descriptions of content metadata. Thus, different knowledge representations are often required when applying KGs to downstream tasks. In this study, we focus on a recommendation task, which is considered important among downstream tasks. Furthermore, we narrow our focus to the domain of anime, which has gained attention as one of the rapidly growing content industries. This study aims to investigate effective knowledge representation through a case study of anime recommendation. The specific approach is to formulate the hypotheses on how to configure knowledge to improve

[a] https://orcid.org/0000-0002-3821-6507
[b] https://orcid.org/0000-0002-2552-6717
[c] https://orcid.org/0000-0002-1939-4455
[d] https://orcid.org/0000-0001-6717-7028

Saito, Y., Egami, S., Sei, Y., Tahara, Y. and Ohsuga, A.
An Analysis of Knowledge Representation for Anime Recommendation Using Graph Neural Networks.
DOI: 10.5220/0012359500003636

In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 2*, pages 243-252
ISBN: 978-989-758-680-4; ISSN: 2184-433X

recommendation performance and to verify them by changing the structure of the KG according to the hypothesis.

The remainder of this paper is organized as follows: Section 2 introduces prior studies on knowledge representation and recommendation models in the content domain; Section 3 formulates a hypothesis and then explains our approach to verify the hypothesis; Section 4 presents the construction process of the knowledge graph and the results of the recommendation experiment; Section 5 discusses the experiment results; Section 6 summarizes this study including prospects.

2 RELATED WORK

2.1 Knowledge Graph

2.1.1 Wikidata

Wikidata (Vrandečić and Krötzsch, 2014) is a typical example of Linked Data. Launched by the Wikimedia Foundation in 2012, Wikidata is one of the Wikimedia projects dedicated to structuring all knowledge within the project, including Wikipedia. Each entity on Wikidata is assigned a unique identifier. The entities encompass items that represent real-world objects, concepts, and events, along with properties that define relationships and characteristics between items and describe attributes related to the items. In accordance with the Linked Data principle (Heath and Bizer, 2011), URIs are assigned based on identifiers, enabling data retrieval from the URIs in JSON or RDF format. Furthermore, SPARQL Protocol and RDF Query Language (SPARQL) endpoint is available.

2.1.2 KG of Manga, Anime and Games

Oishi et al. (Oishi et al., 2019) focused on the multimedia franchise and adaptation relationships of manga, anime, and games (MAG). They constructed a KG of MAG to develop technology to associate bibliographic information with other works automatically. The KG was constructed by collecting information from several Web resources specialized in various adaptations and linking them to the Media Arts Database (MADB) as a whole. MADB (Agency for Cultural Affairs, 2015) is a database that holds metadata related to manga, anime, games, and media arts. It is developed and managed by the Agency for Cultural Affairs of Japan. To promote media arts in Japan, it provides a digital archive of media arts as an open database. The database contains approximately 330,000 manga titles and 120,000 game titles as of March 2022 (Media Arts Consortium JV, 2022). Users can access SPARQL query services and dataset dumps. In the study, they utilized the Superwork model (Lee et al., 2018) to describe the adaptation relationships between works. The quantitative quality of the constructed KGs was evaluated by random sampling of triplets and visual confirmation of the validity of the superordinate-subordinate relationships among entities.

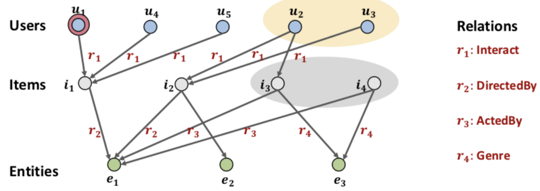

Figure 1: Example of CKG (Wang, et al., pp.951).

2.1.3 Japanese Visual Media Graph

Japanese Visual Media Graph (JVMG) is a project to build a KG for Japanese visual media such as manga, anime, games, and visual novels (Pfeffer and Roth, 2020). The project aims to provide researchers who focus on modern media, themes, and characters with a systematic database. JVMG collected data from the websites of several enthusiast communities. The enthusiast communities have the advantage of describing visual media at a more granular level of detail than general-purpose data sources. In addition, they tend to update information about visual media immediately. The project is currently in the process of integrating auxiliary data sources (e.g., Wikidata, MADB) into three main data sources. The database is expected to be further expanded in the future.

2.2 KG-Based Recommendation

2.2.1 Knowledge Graph Attention Network

Collaborative filtering is a recommendation algorithm for predicting a users' latent preference based on the interaction between users and items. However, it is difficult to capture the semantic relationships between items because collaborative filtering considers only the users' behavioral data.

Knowledge Graph Attention Network (KGAT) is a graph neural network recommender model that takes into account the feature and attribute of item (Wang et al., 2019). KGAT takes a graph as its input, which combines user behavioral (e.g., rating, purchasing, and browsing) data and auxiliary information about items. This graph is referred to as the Collaborative Knowledge Graph (CKG) in the paper (Figure 1). In general, KGAT assumes that a CKG is constructed from a single data source. The model consists of an embedding layer and attentive embedding propagation layers. The embedding layer performs pre-training embedding by applying TransR (Lin et al., 2015) to the CKG. The attentive embedding propagation layers perform propagating embedding by message passing (Gilmer et al., 2017) according to the graph structure. The embedding of each node is updated by aggregating information from neighboring nodes based on the attention mechanism (Vaswani et al., 2017). The output of the final layer can be used as an input for downstream tasks. In the paper, the model is quantitatively evaluated by recommending movies and music.

2.2.2 KG-Boosted Reinforcement Learning for Recommendation

Sakurai et al. proposed an artist recommendation method (Sakurai et al., 2022) that combines a KG of music with reinforcement learning by agents. The KG consists of four types of nodes: users, music, music genres, and artists. They defined the following three types of triplets.

$$\left(u_i, e_{(u_i,m_j)}, m_j\right) : \text{user } u_i \text{ listened to music } m_j$$

$$\left(a_k, e_{(a_k,m_j)}, m_j\right) : \text{artist } a_k \text{ created music } m_j$$

$$\left(m_j, e_{(m_j,g_l)}, g_l\right) : \text{music } m_j \text{ belongs to genre } g_l$$

Applying TransE (Bordes et al., 2013) to the KG constructed from the triplets described above results in obtaining embeddings attached to each node. In the KG with embeddings, agents are positioned at the target user nodes, allowing them to freely explore the graph. It is noteworthy that reinforcement learning for the agent to reach the artist's node from the user's node can ensure the explainability of the recommendation based on the arrival path. To evaluate the proposed method, they conducted experiments for artist recommendation using the Spotify Million Dataset (Chen et al., 2018), which contains 57,880 music, 1,006 users, 14,973 artists, and 2,517 genres. As a result, the proposed method improves the recommendation performance by about 1–2 % against the baseline.

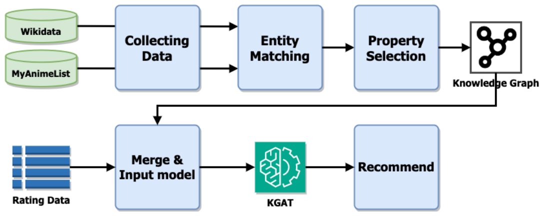

Figure 2: Overview of this study.

3 APPROACH

3.1 Overview

As mentioned in Section 1, this study aims to clarify effective knowledge representation for anime recommendation tasks. We hypothesize how to configure knowledge to improve recommendation performance and verify the hypotheses by changing the structure of the KG correspondingly. Specifically, we have defined the following two hypotheses based on the quantitative and qualitative aspects of knowledge considered in the KG construction.

(A) Quantitative aspect: Can we improve recommendation performance by enriching the semantics of the graph?

(B) Qualitative aspect: Can information that characterizes the work improve recommendation performance among metadata?

Regarding hypothesis (A), we construct KGs by collecting anime data from multiple sources, and then we evaluate the recommendation performance by varying the configuration of data sources. Regarding hypothesis (B), we construct KGs from triplets except for a single arbitrary property and compare the recommendation performance among them. In the manner of ablation study, we quantify what information contributes to a recommendation performance. Figure 2 shows the entire flow of approach from KG construction to recommendation.

3.2 Knowledge Graph Construction

3.2.1 Collecting Data

We considered several data sources with different characteristics to construct a KG of anime. First, we selected MyAnimeList [1] as a data source specific to the domain of anime. MyAnimeList is an international website dedicated to anime and manga. It is also a kind of enthusiast community as defined by

[1] https://myanimelist.net

Pfeffer et al. in Section 2.1.3. MyAnimeList was selected because it provides not only metadata but also a wealth of rating data from registered users. In this study, we use the constructed KG to recommend anime content; therefore, MyAnimeList is suitable for our study. Second, we selected Wikidata as a data source for general-purpose KG. Because it functions as a hub among many datasets and is suitable for entity matching due to its pre-defined properties for identifying the entities in external datasets. We expect to improve recommendation performance not only by simply increasing the amount of information but also by using complementary data sources with different characteristics.

Each data source assigns a unique identifier to an anime work. In MyAnimeList, the anime work is identified by an integer value under the namespace of "https://myanimelist.net/anime/". In Wikidata, the anime is described as an item identified by an integer value prefixed by Q under the namespace of "http://www.wikidata.org/entity/".

3.2.2 Entity Matching

In order to link metadata about an anime entity derived from two different data sources with a single anime entity, we applied entity matching and integrated the data sources. In this process, the integration means creating a dictionary of MyAnimeList identifiers and Wikidata identifiers for the anime mentioned in Section 3.2.1. Entity matching consists of two main steps. The first step is direct entity matching. When a Wikidata anime item has MyAnimeList anime ID[2] as one of its properties, we got the pair of the item's Wikidata identifier and MyAnimeList anime ID (MyAnimeList identifier). This allowed us to identify anime entities described in two separate data sources. The second step is indirect entity matching using a third-party data source as an intermediate dictionary. MyAnimeList entities often have third-party website links, such as Wikipedia or other streaming platforms. Similarly, Wikidata entities have identifiers of third-party data sources in addition to the MyAnimeList anime ID. We identified the anime entities by using third-party data sources that connect to both MyAnimeList and Wikidata as intermediate dictionaries.

Note that we had to deal with differences in the granularity of anime work descriptions between data sources. For instance, there is a case where a work W is defined as a single entity in Wikidata. In contrast, it is divided into two entities in the third-party, and in MyAnimeList, it is split into three entities. This

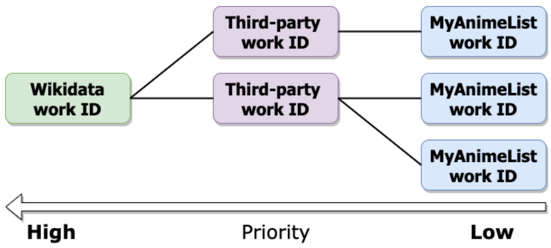

Figure 3: Differences in anime work description granularity and data source priority.

discrepancy is caused by different data sources often having different editing policies, such as whether to regard an anime series as a single work or as several different works. We handled the problem by assigning priorities to each data source. Specifically, priority is assigned based on the following order relation.

$$W_{\text{wikidata}} \succ W_{\text{third-party}} \succ W_{\text{myanimelist}} \quad (1)$$

Therefore, in the aforementioned case, multiple work entities described in MyAnimeList are aggregated into a single entity described in Wikidata as shown in Figure 3. Since Wikidata is a general-purpose dataset in a wide range of domains, it is difficult to capture detailed adaptation relationships specific to anime works. In contrast, MyAnimeList is a data source specialized in anime works; thus, it can describe works in detail. Consequently, we assigned priorities following the order relationship as shown in Equation 1.

3.2.3 Property Selection

Entity matching allowed us to link knowledge derived from different data sources to a single work entity. Among the knowledge, resources (such as categories or classes) that have unique identifiers enrich the semantics of KG. Therefore, properties that refer to them are required. For this reason, we selected properties by imposing constraints. We listed properties and their values that meet all of the following conditions.

- Conditions (a) on Wikidata properties:
  - A property whose subject belongs to the class "anime (Q1107)" in a chain
  - A property whose number of appearances in Wikidata is 1000 or more
  - A property whose data type is "Item"
- Conditions (b) on MyAnimeList properties:
  - A property whose value functions as a resource or category
  - A property whose value does not contain a URL

[2]https://www.wikidata.org/wiki/Property:P4086

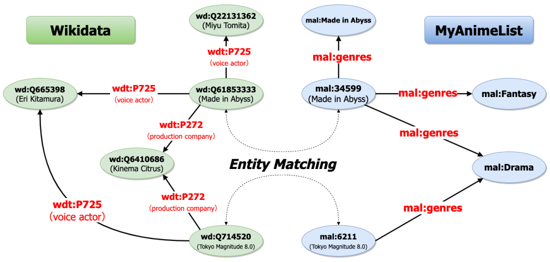

Figure 4: Example of entity matching and peripheral structure across “Made in Abyss” and “Tokyo Magnitude 8.0”.

For example, the partial structure of the KG around “Made in Abyss” and “Tokyo Magnitude 8.0” is shown in Figure 4. “Made in Abyss” is assigned the identifier “Q61853333” (Made in Abyss: Japanese anime television series)[3] in Wikidata, and the identifier “34599” in MyAnimeList. We applied entity matching by creating a key-value pair of Q61853333 and 34599 according to the above methodology, and properties were selected according to Section 3.2.3. “Tokyo Magnitude 8.0” also has Wikidata and MyAnimeList identifiers respectively, and entity matching is applied similarly. On the subgraph, these two works are linked to a common entity, because they share common elements in production company, voice actor, and genre. Such co-references are repeated in many other works, allowing KG to capture the higher-order relationships between works.

3.3 Recommendation

We performed an experiment for the anime recommendation task using the constructed KGs as a part of the model’s input. The recommendation model is KGAT introduced in Section 2.2.1. As mentioned, KGAT inputs a CKG that combines the User-Item bipartite graph used in collaborative filtering and the Item KG based on the auxiliary information. The User-Item bipartite graph corresponds to an existing dataset by MyAnimeList users. The Item KG means the KG based on the metadata constructed in Section 3.2. By linking the KGs and the rating dataset with MyAnimeList identifiers, we constructed a CKG as shown in Figure 1, and KGAT learned its structure to recommend the anime works to the users.

Table 1: Results of entity matching (EM) per step.

	Step 0	Step 1	Step 2
Number of EM work	0	2,989	5,740
Percentage of EM (%)	0	18	35

[3]http://www.wikidata.org/entity/Q61853333

4 EXPERIMENTS

4.1 Evaluation of KG Construction

4.1.1 Dataset

In this study, since we are planning to use the KG for the recommendation, we have selected Anime Dataset with Reviews - MyAnimeList (Marlesson, 2020). This dataset consists of three tables of data: “animes.csv”, “profiles.csv”, and “reviews.csv”. Since “animes.csv” contains metadata about anime works, we constructed KG based on this data. However, the metadata is outdated because it covers data up to 2020 and the MyAnimeList API changed in specifications. To update the data, from June 8 to June 9, 2023, we crawled and supplemented data for 16,216 works included in the dataset. As a result, we have obtained 16,082 works with detailed data. The 134 missing works are thought to be due to differences between the present and when the dataset was created, such as page deletion.

4.1.2 Evaluation of Entity Matching

According to Section 3.2.2, we applied entity matching in two steps. In the first step, we obtained a Wikidata item that has the property P4086 (MyAnimeList anime ID). In the second step, three third-party websites were selected as intermediate dictionaries: “AniDB”[4], “Anime News Network”[5] and “Official Website”. These websites were considered to be sufficiently active as intermediate dictionaries, because in the existing dataset (Marlesson, 2020), 12,348 external links were assigned to AniDB, 8,864 to the Anime News Network, and 10,927 to the Official Website. In both steps, we collected the data in Wikidata on June 12, 2023. The number and percentage of entity matching per step are shown in Table 1.

4.1.3 Selected Property

As for Wikidata, we used Wikidata Query Service to list the properties that meet the conditions (a) as of September 27, 2023. As for MyAnimeList, we listed the properties that meet conditions (b) based on the added detailed data. Table 5 and Table 6 show the properties listed by data source in the Appendix.

When changing the configuration of data sources, the prerequisites of the experiments should be standardized for all the KGs. Thereby, we filtered the works that have both MyAnimeList and Wikidata

[4]https://anidb.net
[5]https://www.animenewsnetwork.com

Table 2: Statistics of constructed KGs [7].

	# triplet	# entity	# property
KG-all	191,699	28,666	50
KG-mal	106,049	17,626	25
KG-wikidata	85,489	16,577	18

properties. Consequently, 5,621 works out of 5,740 ended up being included in the KG.

Eventually, we constructed a KG consisting of 191,699 triplets by integrating MyAnimeList and Wikidata. Including it, all the KGs were constructed with RDFLib, a Python library. The RDF exported in Turtle format was stored in GraphDB [6]. All datasets of this study are available at https://anonymous.4open.science/r/kgat-input-DB18 in a format that can be inputted for KGAT.

4.2 Quantitative Aspect: Recommendation Experiment I

4.2.1 Experimental Setup

Firstly, we evaluated the constructed KGs through the anime recommendation task. We constructed multiple KGs with different configurations of the data sources used, in order to evaluate the effect of the amount of knowledge on the recommendation performance. Specifically, we constructed the following 3 types of KGs as follows.

- **KG-All:** based on MyAnimeList and Wikidata
- **KG-Mal:** based only on MyAnimeList
- **KG-Wikidata:** based only on Wikidata

We compared their recommendation performances, as the inputs to the KGAT model. The statistics by knowledge considered in the KG construction are shown in Table 2.

As already mentioned in Section 2.2.1, the KGAT model takes a CKG as its input. In order to construct the CKG, we selected a dataset "reviews.csv" from the aforementioned dataset. We implemented a model based on KGAT-pytorch[8], which is designed for a problem known as Top-K recommendation. This problem predicts the top K items in each user's list of works to be rated. Recall@K and nDCG@K are ranking metrics to evaluate the quality of Top-K recommendation.

These metrics for the recommendation task are calculated as follows:

[6]https://graphdb.ontotext.com

[7]There are 7 other properties that are added automatically in the process of loading RDF into GraphDB.

[8]https://github.com/LunaBlack/KGAT-pytorch

Table 3: Typical hyperparameters of KGAT model.

# epoch	LR	Dimensions	@K value
100	1e-4	[64, 32, 16]	[20, 40, 60, 80, 100]

$$\text{Recall@K} = \frac{\text{Number of correct items in the top K}}{\text{Number of possible items in total}} \tag{2}$$

$$\text{nDCG@K} = \frac{\text{DCG@K}}{\text{IDCG@K}} \tag{3}$$

, where DCG@K can be calculated as follows if the i-th prediction is in fact the evaluated value rel_i.

$$\text{DCG@K} = \sum_{i=1}^{K} \frac{2^{rel_i} - 1}{\log_2(i+1)} \tag{4}$$

, where IDCG@K denotes the DCG for the ideal ranking order. That is, it expresses how much DCG can be obtained for the best possible ranking. nDCG@K is equivalent to normalized DCG@K by IDCG@K and it ranges from 0 to 1. We evaluated the quantitative quality of recommendation by means of these metrics.

In addition, we converted reviews.csv to the following dictionary format:

$$\{u_1 : [i_1, i_2, \cdots],\ u_2 : [i_1, i_3, \cdots], \cdots\} \tag{5}$$

, where u_n denotes a unique user id, and i_m denotes a unique item id that user rated. We split it into the train and test data by each user's item list. In the original KGAT, the split ratio is train:test = 8:2. Following this, the number of works rated by each user should be at least 5, so we extracted users who have five ratings or more as a threshold. As a result, 53,000 rating data by 4,330 users remained.

We experimented with three different KGs and preprocessed rating data as a common input for the recommendation. Table 3 shows the typical hyperparameters used in training the model. Hyperparameters are kept at the default values in the implementation. We used the same configuration for all experiments in this paper.

4.2.2 Experimental Result

Table 4 shows the results of the evaluation experiment by recommendation. We compared the recommendation performance for each KG configuration. As a baseline input, the recommendation performance of collaborative filtering without KG was set to "**None**". The @K values of metrics are listed only for K = 20, 60, and 100 as representative.

Table 4: Recommendation performance by each KG configuration.

	Recall			nDCG		
	@K = 20	@K = 60	@K = 100	@K = 20	@K = 60	@K = 100
KG-all	**0.0959**	0.2052	0.2828	**0.0487**	**0.0752**	**0.0910**
KG-mal	0.0916	0.2007	0.2717	0.0435	0.0702	0.0845
KG-wikidata	0.0911	**0.2054**	**0.2869**	0.0450	0.0730	0.0897
None	0.0779	0.1650	0.2323	0.0376	0.0588	0.0722

4.3 Qualitative Aspect: Recommendation Experiment II

Secondly, we constructed new KGs that exclude an arbitrary property in the manner of ablation study. To accurately measure the effect of the excluded properties in the recommendation, a KG constructed from a single data source is set as a baseline. When building a KG from multiple data sources, the information about a work described by properties may not necessarily be unique. We can quantify the effect of unique information described excluded an arbitrary property by setting a single data source as the baseline.

In particular, we set KG-wikidata as the baseline. Specifically, all 18 properties of Table 5 were excluded one by one, and the KGs constructed in each case were used as input for the recommendation experiment. Calculating the recommendation performance ratio of the KG for each excluded property to the baseline KG, we quantified how much the excluded properties contributed to the recommendation. The results are shown in Figure 5. In more details, results for all the metrics are shown in Figure 6 in the Appendix.

5 DISCUSSION

5.1 Effects of Data Source Variation

First, in the evaluation experiment regarding recommendation, we varied the data source configuration considered in KG construction. Table 4 reveals that when "None" is used, the recommendation performance is at its lowest. This indicates that taking some knowledge into account would lead to improved recommendation performance. When comparing "KG-all" with "KG-mal" and "KG-wikidata", KG-all demonstrated the best recommendation performance across four metrics. Additionally, Table 2 shows that the recommendation performance is roughly proportional to the number of triplets in the KG. Despite KGAT typically assuming knowledge from a single data source, "KG-all" achieved the best performance. This result can be attributed not only to the increased volume of data but also to the mutually complementary role of the two data sources with different characteristics. Specifically, it appears that Section 3.2.2 dealt with the description granularity gap, and Section 3.2.3 selected properties with rich semantics, both of which played an effective role.

In summary, Hypothesis A, as outlined in Section 3.1, is substantiated. This is because the experiments confirm that recommendation performance improves as the number of triplets increases. In other words, the experiments highlight the positive effect of the quantitative aspect of knowledge considered during KG construction on recommendation performance.

5.2 Effects of Property Exclusion

Second, we excluded properties from KG-wikidata in the manner of ablation study. Figure 5 shows the Recall@20 ratio concerning the baseline for each excluded property. To interpret the graph, red bars signify properties with a negative effect on the recommendation, as their exclusion led to performance improvements. Conversely, blue bars indicate properties with a positive effect on the recommendation, as their exclusion resulted in performance declines.

According to Figure 5, the properties that had a negative effect by being excluded are "genre (P136)", "has part(s) (P527)" and "country of origin (P495)". The first property describes the work's genre, the second describes the hierarchical relationships between the works, and the third describes the country where the work was produced. Compared within the metadata, these properties are information that can characterize the work. In particular, "genre" most directly characterizes the work. The other two properties do not necessarily directly characterize the work. However, works from the same series or produced in the same country are likely to have similar characteristics. This result seems to support Hypothesis B in Section 3.1.

On the other hand, the property that had a large positive effect by being excluded is "characters (P674)". This is a property that describes the characters appearing in the work. Since it describes a character's name specific to an anime work, the associated

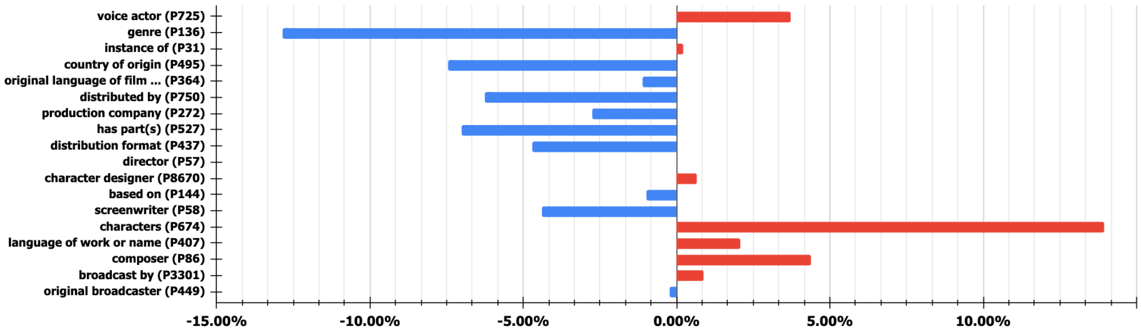

Figure 5: The effect of excluded property in recommendation (Recall@20 ratio to baseline).

triplets frequently contain named entities. Almost all named entities rarely link to entities of other works, thus they often exist as leaf nodes in the KG. When the graph neural networks propagate the embeddings according to the structure of KG, the leaf nodes are dead ends, thus preventing effective propagation of the embeddings. Therefore, properties of named entities, such as "characters", are thought to have impaired recommendation performance.

The "voice actor (P725)" property also has a positive effect due to its exclusion. Referring to Table 5 in the Appendix, "voice actor" appears the most frequently among the Wikidata properties. Nevertheless, the exclusion of "voice actor" does not significantly affect the recommendation performance in Figure 5. This result emphasizes that qualitative factors such as KG's topology and property semantics should also be considered, as quantitative factors do not solely determine recommendation performance.

6 CONCLUSION

In this study, we investigated effective knowledge representation through a case study of recommending anime works. Our approach is to first hypothesize how to configure knowledge in order to improve recommendation performance, and then verify the hypotheses by changing the KG's structure according to the hypothesis. We aimed to identify which aspects of knowledge about anime contributed to the recommendation by inputting multiple KGs into a graph neural network model and comparing their recommendation performance. Our findings revealed that: in terms of quantity, incorporating more data sources into the KG led to improved recommendation performance, in terms of quality, the knowledge that characterized the works more effectively contributed to the better recommendation. This study offers a novel methodology for constructing downstream-task-aware knowledge representation by evaluating the effects of structural changes in KGs on the performance of recommendation. Our method is generalizable to other content domains: inductive studies that investigate differences for every domain, and the development of systems that automatically discover important properties for KG-based recomendation.

However, there is still room for discussion regarding property exclusion results. It is important to standardize the prerequisites for all properties, such as equalizing the number of triplets for each property. Additionally, it is advisable to introduce new metrics or utilize existing ones to evaluate the contribution of each property to KG-based recommendation.

As an alternative methodology, text-based KG construction is worth considering. For example, the automatic KG construction from a work's synopsis text is feasible. As such texts provide detailed descriptions of a work's storyline or worldview, they have the potential to enrich properties that characterize the work. In recent years, there have been numerous studies on the construction of KGs from a natural language using Large Language Models (Mihindukulasooriya et al., 2023) (Giglou et al., 2023); thus, we consider enriching our KG by using such advanced technologies.

ACKNOWLEDGEMENTS

This research was supported by JSPS Grants-in-Aid for Scientific Research JP21H03496, JP22K12157, JP23H03688, JP22K18008 and Sumitomo Electric Group Social Contribution Fund.

REFERENCES

Agency for Cultural Affairs (2015). Media arts database.

Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked data - the story so far. *Linking the World's Information*.

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., and Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 2787–2795, Red Hook, NY, USA. Curran Associates Inc.

Chen, W., Lamere, P., Schedl, M., and Zamani, H. (2018). Recsys challenge 2018: Automatic music playlist continuation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 527–528.

Giglou, H. B., D'Souza, J., and Auer, S. (2023). Llms4ol: Large language models for ontology learning.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. (2017). Neural message passing for quantum chemistry. In Precup, D. and Teh, Y. W., editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272. PMLR.

Heath, T. and Bizer, C. (2011). *Principles of Linked Data*, pages 7–27. Springer International Publishing, Cham.

Lee, J. H., Jett, J., Cho, H., Windleharth, T., Disher, T., Kiryakos, S., and Sugimoto, S. (2018). Reconceptualizing superwork for improved access to popular cultural objects. *Proceedings of the Association for Information Science and Technology*, 55(1):274–281.

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. (2015). Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 2181–2187. AAAI Press.

Marlesson (2020). Kaggle: Anime Dataset with Reviews - MyAnimeList.

Media Arts Consortium JV (2022). *Implementation Report on the Project to Promote the Development of Media Arts Collaboration Infrastructure, etc.* Number FY2022. Media Arts Consortium JV.

Mihindukulasooriya, N., Tiwari, S., Enguix, C. F., and Lata, K. (2023). Text2kgbench: A benchmark for ontology-driven knowledge graph generation from text.

Oishi, K., Mihara, T., Nagamori, M., and Sugimoto, S. (2019). Identifying and linking entities of multimedia franchise on manga, anime and video game from wikipedia. In Jatowt, A., Maeda, A., and Syn, S. Y., editors, *Digital Libraries at the Crossroads of Digital Information for the Future*, pages 95–101, Cham. Springer International Publishing.

Pfeffer, M. and Roth, M. (2020). Japanese visual media graph: Providing researchers with data from enthusiast communities. *International Conference on Dublin Core and Metadata Applications*, pages 136–141.

Sakurai, K., Togo, R., Ogawa, T., and Haseyama, M. (2022). Explainable artist recommendation based on reinforcement knowledge graph exploration. In *International Workshop on Advanced Imaging Technology (IWAIT) 2022*, volume 12177, pages 91–96. SPIE.

Soldo, Lana and Schagerl, Christopher (2023). Impact of the Covid-19 Pandemic on Netflix. *MAP Education and Humanities*, 3(1):75–82.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Vrandečić, D. and Krötzsch, M. (2014). Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Wang, X., He, X., Cao, Y., Liu, M., and Chua, T.-S. (2019). Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pages 950–958, New York, NY, USA. Association for Computing Machinery.

APPENDIX

Table 5: Wikidata properties in KG (18 in total).

Property label	Identifier	# triplet
voice actor	P725	28681
genre	P136	9386
instance of	P31	7485
country of origin	P495	5560
original language of film ···	P364	5350
distributed by	P750	3417
production company	P272	3112
has part(s)	P527	3110
distribution format	P437	3013
director	P57	2680
character designer	P8670	2146
based on	P144	2063
screenwriter	P58	1928
characters	P674	1842
language of work or name	P407	1724
composer	P86	1670
broadcast by	P3301	1220
original broadcaster	P449	1102

	Recall@20	Recall@60	Recall@100	nDCG@20	nDCG@60	nDCG@100
voice actor	3.73%	-1.04%	-2.68%	4.67%	1.16%	-0.33%
genre	-12.84%	-6.96%	-4.50%	-8.00%	-5.64%	-4.35%
instance of	0.22%	-5.13%	-3.42%	-1.11%	-3.81%	-3.23%
country of origin	-7.46%	-2.56%	-0.73%	-5.33%	-2.82%	-1.67%
original language of film or TV show	-1.10%	-4.92%	-3.07%	2.67%	-1.07%	-0.78%
distributed by	-6.26%	-6.96%	-6.59%	-5.56%	-6.38%	-6.24%
production company	-2.74%	-8.27%	-6.48%	-1.78%	-5.97%	-5.35%
has part(s)	-7.03%	-8.80%	-6.34%	-3.11%	-5.93%	-4.91%
distribution format	-4.72%	-5.23%	-4.84%	-2.89%	-3.84%	-4.01%
director	0.00%	-2.12%	-1.05%	-0.44%	-1.67%	-1.11%
character designer	0.66%	-6.79%	-6.59%	-0.22%	-4.85%	-5.24%
based on	-0.99%	-6.67%	-5.82%	-0.44%	-4.17%	-4.12%
screenwriter	-4.39%	-5.86%	-4.84%	-5.33%	-5.72%	-5.13%
characters	13.94%	3.96%	2.75%	9.11%	4.02%	3.01%
language of work or name	2.09%	0.28%	-1.53%	1.11%	0.37%	-0.78%
composer	4.39%	-6.96%	-4.25%	3.56%	-6.38%	-2.68%
broadcast by	0.88%	-6.24%	-3.62%	2.67%	-2.76%	-1.78%
original broadcaster	-0.22%	-8.19%	-8.78%	1.11%	-4.52%	-5.80%

Figure 6: All the effect of excluded property in recommendation in heatmap.

Table 6: MyAnimeList properties in KG (25 in total).

Property label	# triplet
type	15941
genres	13451
producers	12179
themes	6245
studios	5767
status	5621
source	5621
title	5621
year	5621
rating	5621
season	5621
licensors	3911
demographics	2019
relations:Adaptation	3229
relations:Other	1660
relations:Sequel	1613
relations:Side story	1324
relations:Prequel	1194
relations:Alternative version	858
relations:Parent story	846
relations:Alternative setting	633
relations:Summary	510
relations:Spin-off	408
relations:Character	294
relations:Full story	241