

Article

Modified Conditional Restricted Boltzmann Machines for Query Recommendation in Digital Archives

Jiayun Wang ^{1,*}, Biligsaikhan Batjargal ², Akira Maeda ³, Kyoji Kawagoe ³ and Ryo Akama ⁴¹ Graduate School of Information Science and Engineering, Ritsumeikan University, Shiga 525-8577, Japan² Research Organization of Science and Technology, Ritsumeikan University, Shiga 525-8577, Japan³ College of Information Science and Engineering, Ritsumeikan University, Shiga 525-8577, Japan⁴ College of Letters, Ritsumeikan University, Kyoto 603-8577, Japan

* Correspondence: jiayunwong@hotmail.com

Abstract: Digital archives (DAs) usually store diverse expert-level materials. Nowadays, access to DAs is increasing for non-expert users, However, they might have difficulties formulating appropriate search queries to find the necessary information. In response to this problem, we propose a query log-based query recommendation algorithm that provides expert knowledge to non-expert users, thus supporting their information seeking in DAs. The use case considered is one where after users enter some general queries, they will be recommended semantically similar expert-level queries in the query logs. The proposed modified conditional restricted Boltzmann machines (M-CRBMs) are capable of utilizing the rich metadata in DAs, thereby alleviating the sparsity problem that conventional restricted Boltzmann machines (RBMs) will face. Additionally, compared with other CRBM models, we drop a large number of model weights. In the experiments, the M-CRBMs outperform the conventional RBMs when using appropriate metadata, and we find that the recommendation results are relevant to the metadata fields that are used in M-CRBMs. Through experiments on the Europeana dataset, we also demonstrate the versatility and scalability of our proposed model.

Keywords: query recommendation; digital archives; conditional restricted Boltzmann machines; machine learning

check for
updates

Citation: Wang, J.; Batjargal, B.; Maeda, A.; Kawagoe, K.; Akama, R. Modified Conditional Restricted Boltzmann Machines for Query Recommendation in Digital Archives. *Appl. Sci.* **2023**, *13*, 2435. <https://doi.org/10.3390/app13042435>

Academic Editors: Shoujin Wang and Qi Zhang

Received: 23 December 2022

Revised: 7 February 2023

Accepted: 9 February 2023

Published: 14 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Digital archives (DAs) are aimed at preserving human knowledge and artifacts in databases via converting and storing them as digital contents such as text, images, audio, and video. During the last few decades, to preserve these digital contents, tremendous efforts have been made to develop advanced techniques, including modeling 3D or 4D objects [1,2], processing historical documents [3], and archiving geospatial data [4]. Since search techniques have been regarded as the core information access method in DAs, they are developed and optimized extensively. Some of the representative search techniques in DAs are proposed for keyword searches, image searches [5], and semantic searches [6].

The recommender system (RecSys) is widely used as one of the information access methods in many fields, including the research and applications of personalization in cultural heritage listed in [7]. In DAs, RecSys is barely being paid attention to and is less developed. We focus on building RecSys in DAs that has the potential to gain more attention from the crowd and support its usage.

Researchers have become increasingly aware of the importance of RecSys in DAs as well as in other digital galleries, libraries, archives, and museums (GLAMs). For example, Wilson-Barnao [8] emphasized that algorithmic cultural recommendation creates commercial value that is embodied in historical and cultural values, and the significance of the Google Cultural Institute is that it bridges a commercial enclosure with the cultural collections of public institutions.

The main purpose of our proposed method is to use the queries formulated by high-level experts to help other users find the appropriate query (or queries) to narrow down the scope of the search results. The proposed model can be downloaded on GitHub (<https://github.com/blueorris/M-CRBMs>, accessed on 22 December 2022). According to the work in [9], researchers are accessing Europeana (<https://www.europeana.eu/portal/en>, accessed on 22 December 2022) [10], which is a web portal created by the European Union containing digitized museum collections of more than 3000 institutions across Europe for seeking materials, which is often considered the main purpose of the users coming to DAs. There are various types of users and various purposes for the users of Europeana. It is worth noting that the fact that various types of users (including cultural heritage enthusiasts, students, academics, teachers, cultural heritage professionals, and others) are accessing Europeana illustrates that users have different levels of understanding of the materials stored in DAs. The fact that the proportion of the purposes “create new work” and “personal interest” is larger than that of “professional activities” illustrates that seeking research-relevant materials is only part of the information needs in DAs. Therefore, it is necessary to rethink how to build the information access functions in DAs for supporting better usage. Since users who do not have a high level of understanding of the materials may fail to formulate appropriate queries for their purposes, we propose a method that recommends candidate queries to assist these users.

Query recommendation (or query suggestion) assists users to refine queries in order to satisfy their information needs. Often, users try different queries until they are satisfied with the results. This will be hindered by having little knowledge about the information for which they are searching. As mentioned before, the users might vary from different levels of expert knowledge of the contents in DAs, thus causing difficulties in formulating appropriate queries for information seeking.

For example, such difficulties occur when a user searches ukiyo-e (ukiyo-e is a kind of famous Japanese art of woodblock prints) databases. Users with basic background knowledge of ukiyo-e would formulate a simple search query, such as “美人” (meaning “beauty”, which is one of the most famous themes of ukiyo-e), but experts tend to use more specific queries to limit the result scope, such as “三美人” (meaning “three beauties”), “当世美人揃” (meaning “beauties of the present age”) or “見返り美人図” (meaning “beauty looking back”). In this example, the queries formulated by experts related to “美人” are often the whole title or the series name of an ukiyo-e print or the subwords of them.

In this article, we utilize the query-item pairs extracted from the access log of the Ukiyo-e Portal Database [11] of the Art Research Center (ARC-UPD, Ritsumeikan University) to train the proposed model and conduct our experiments. The ARC-UPD is one of the biggest DAs that stores ukiyo-e prints as digital versions and their extensive metadata (artist, title, genre, etc.). More than 200,000 public ukiyo-e prints can be browsed in the ARC-UPD as of August 2022.

The restricted Boltzmann machine (RBM) is a two-layer undirected graphical model with a visible layer of observable variables or features and a hidden layer of latent, representative units. It has been used for many different machine learning tasks, including image generation [12], dimensionality reduction [13], and representation learning of human motion [14]. It was first introduced to the recommendation task by Salakhutdinov et al. [15]. Compared with RBMs, conditional RBMs (CRBMs) take extra information into account and are often applied to temporal sequences of data, such as the CRBM model proposed by Taylor, G. W. et al. [14] and the CRBM model proposed by Salakhutdinov, R. et al. [16]. Both RBMs and CRBMs capture the dependencies between visible layer variables by associating energy to each configuration of those variables.

Our idea for query recommendation is that similar queries should exist in one configuration. For example, if the queries “美人” (meaning “beauty”) and “三美人” (meaning “three beauties”) are often used to search for one certain ukiyo-e print, then these two queries should be in one configuration. The weight matrices in the energy model extract the dependencies of each of the units (the units can represent queries or metadata values)

that finally figure out the configurations. In this case, the RBM model only learns the dependencies among the queries. The CRBM model utilizes extra information to help find the potential relative queries in one configuration. In M-CRBMs, the reduced weight matrix from conventional CRBMs is between the conditional layer and visible layer (i.e., the dependencies directly between the queries and metadata). The details of the methodology will be explained in Sections 3 and 4.

The novelty and contributions of this work are as follows:

1. We propose a method to recommend queries in DAs called modified conditional restricted Boltzmann machines (M-CRBMs). M-CRBMs provide users with different expert knowledge levels to seek information in the database when given an initial query.
2. We modify the conventional CRBMs and construct M-CRBMs by reducing the weight matrices in the model. This makes the model trainable for average-performing computers. Aside from that, we use free energy instead of energy to efficiently train the model.
3. The proposed M-CRBM model is able to predict the queries relevant to the user's query and predict the relevance degree (ranking) simultaneously.

This paper is structured as follows. Section 2 states the work related to our research. Section 3 introduces the basic methodology of RBMs and CRBMs. Section 4 states the methodology of our proposed M-CRBMs. Section 5 explains the DA dataset that we use in the experiments. Section 6 describes the experiments, and Section 7 concludes the paper.

2. Related Work

The related work can be divided into three types: (1) recommendations in GLAMs; (2) RBMs and CRBMs for the recommendation task, and (3) query recommendation.

2.1. Recommendation in GLAMs

Wang et al. [17] designed Cultural Heritage Information Personalization (CHIP), integrating the user's interest ratings with ontological reasoning by utilizing semantic web technologies to enrich the presentation of museum collections. This enabled the system to suggest more related items (i.e., suggest artworks not only from the same artists but also related artists).

Semeraro et al. [18] proposed a hybridization of collaborative filtering with a content filter using a fuzzy taste vector. With the traditional collaborative filtering recommendation algorithm, the content filter effectively solves the "cold start" problem which occurs in sparse library datasets. This method is proven to be suitable for libraries with relatively small collections.

These prior studies have made great contributions to the field of recommendation in GLAMs. However, to the best of our knowledge, no one has mentioned query recommendation in the field of recommendation in GLAMs until now.

2.2. RBMs and CRBMs for the Recommendation Task

The RBM has frequently been used in the recommendation field since it was first proposed in [15] by Salakhutdinov et al. Collaborative filtering (CF), the most famous algorithm in RecSys, has the idea that if two items get similar rating patterns then they are probably similar. This RBM-CF model inherits this idea that it treats the ratings assigned by each user for all items as a single training case, then each hidden unit could learn to model a significant dependency between the ratings of different items. This work also proposed CRBMs to incorporate a vector that indicates the user-rated items.

The aforementioned RBM-CF has the equivalent idea with user-based CF given users' ratings, and it can easily be converted to be equivalent to item-based CF by utilizing items' ratings. Georgiev et al. [19] proposed a unified framework combining the user-based RBM-CF and item-based RBM-CF. In this case, the visible units are determined by both

user hidden units and item hidden units. The studies focused on CRBMs often built the models by incorporating useful side information into the RBM, which was often found to have the ability to deal with the sparse dataset or “cold start” problem. Wu et al. [20] incorporated the trust information (i.e., trust statements explicitly given by users) into CRBMs. Pujahari et al. [21] added the preference relation as side information into the conditional layers of CRBMs. Chen et al. [22] proposed CRBMs considering the user’s preferred items, examined items, and sampled negative items.

There are many of variants of RBMs and CRBMs for the recommendation task built for different datasets. Although they are intuitively suitable algorithms for query recommendation, none of the existing studies applied them to the query recommendation task.

2.3. Query Recommendation

Baeza-Yates et al.’s proposed method [23] creates a term-weight vector for each query and then uses k-means clustering to find the semantically similar queries of the historical preferences of users registered in the query log of a search engine which also have sufficient popularity. However, we argue that popularity of queries is not such an important factor in DAs because many of the knowledgeable queries do not frequently occur.

Many past efforts proposed methods to not only utilize the query log but also other information that the system could collect. Huang et al. [24] used the current query as well as the current session of queries as the context for query suggestions. Song et al. [25] used Markov models of sequential queries to predict the user’s intent before making a recommendation. Feild et al. [26] leveraged the user’s information about the current search context, which refers to whether the leveraged queries address the user’s information needs.

With the development and integration of search engines, there are few examples of recent research on query recommendation algorithms based solely on the queries themselves and more based on the user information that could be obtained from the systems (e.g., session information or the virtual or physical context). Considering that the search functions of many DAs are constructed by various institutions themselves, it is difficult to obtain information as easily as large search engines. Due to this reason, we propose an algorithm purely based on the queries and accessed items, which can be simply obtained from the access log data.

3. RBMs and CRBMs

3.1. Restricted Boltzmann Machines

An RBM is a two-layer undirected graphical model constructed with a binary visible layer and a binary hidden layer, as shown in Figure 1. RBMs are a kind of energy-based model (EBM) [27] that captures the dependencies between variables by associating a scalar energy to each configuration of the variables. The energy function of RBMs is given by

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{h}^\top \mathbf{W} \mathbf{v} - \mathbf{b}_v^\top \mathbf{v} - \mathbf{b}_h^\top \mathbf{h}, \quad (1)$$

where \mathbf{v} is the visible layer variables, \mathbf{h} is the hidden layer variables, \mathbf{b}_v is the bias of the visible layer, \mathbf{b}_h is the bias of the hidden layer, and \mathbf{W} is the weight between \mathbf{v} and \mathbf{h} . Every joint configuration (\mathbf{v}, \mathbf{h}) has a probability of the form

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v}, \mathbf{h})), \quad (2)$$

where Z is the partition function given by enumerating over all possible configurations of (\mathbf{v}, \mathbf{h}) . This makes the energy function become the format of probability. Z is defined by the following formula:

$$Z = \sum_{\mathbf{v}, \mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})), \quad (3)$$

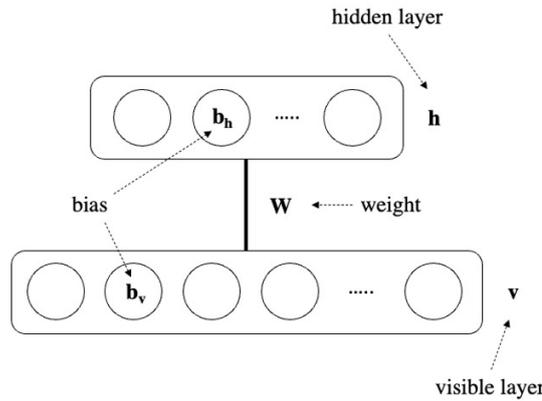


Figure 1. A restricted Boltzmann machine with a binary visible layer and binary hidden layer.

The marginal probability $p(\mathbf{v})$ could be calculated by summing all the hidden layer variables \mathbf{h} in $p(\mathbf{v}, \mathbf{h})$:

$$p(\mathbf{v}) = \frac{1}{Z} \sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})), \tag{4}$$

Because hidden units are conditionally independent in RBMs, we have

$$p(\mathbf{h} | \mathbf{v}) = \prod_i p(h_i | \mathbf{v}), \tag{5}$$

where h_i is the binary state of each unit. Then, by utilizing the joint probability in Equation (2), the marginal probability in Equation (4), and the independent rule in Equation (5), we could calculate the conditional probability of hidden variables given the visible variables:

$$\begin{aligned} p(\mathbf{h} | \mathbf{v}) &= p(\mathbf{h}, \mathbf{v}) / p(\mathbf{v}) \\ &= \frac{\exp(\mathbf{h}^T \mathbf{W} \mathbf{v} + \mathbf{b}_v^T \mathbf{v} + \mathbf{b}_h^T \mathbf{h}) / Z}{\sum_{\mathbf{h}} \exp(\mathbf{h}^T \mathbf{W} \mathbf{v} + \mathbf{b}_v^T \mathbf{v} + \mathbf{b}_h^T \mathbf{h}) / Z} \\ &= \frac{\prod_j \exp(h_j \mathbf{W}_j \cdot \mathbf{v} + b_j^h h_j)}{\prod_j (\sum_{h_j \in \{0,1\}} \exp(h_j \mathbf{W}_j \cdot \mathbf{v} + b_j^h h_j))} \\ &= \prod_j \frac{\exp(h_j \mathbf{W}_j \cdot \mathbf{v} + b_j^h h_j)}{1 + \exp(b_j^h + \mathbf{W}_j \cdot \mathbf{v})}, \end{aligned} \tag{6}$$

where b_j^h is the bias of each hidden unit and b_j^v is the bias of each visible unit. From Equation (6), we can see that in the forward inference phrase, to obtain the probability of activation of each unit in the hidden layer (which can also be considered the probability that one exists in each hidden unit), we just calculate

$$\begin{aligned} p(h_j = 1 | \mathbf{v}) &= \frac{\exp(b_j^h + \mathbf{W}_j \cdot \mathbf{v})}{1 + \exp(b_j^h + \mathbf{W}_j \cdot \mathbf{v})} \\ &= \text{sigmoid}(b_j^h + \mathbf{W}_j \cdot \mathbf{v}) \end{aligned} \tag{7}$$

Since an RBM is symmetrical, we can calculate the conditional probability of visible variables given the hidden variables in the backward inference as follows:

$$p(v_k = 1 | \mathbf{h}) = \text{sigmoid}(b_k^v + \mathbf{h}^\top \mathbf{W}_{.k}) \tag{8}$$

The conditional probabilities $p(h_j = 1 | \mathbf{v})$ and $p(v_k = 1 | \mathbf{h})$ are used in the contrastive divergence (CD) training method of RBMs [28]. The details will be stated in Section 3.2.

3.2. Learning Rule of RBMs

Different from other EBM models that directly minimize the defined loss functions (or energy functions) in the learning phase, RBMs minimize the negative log-likelihood of the probabilities of the observed variables $\mathbf{v}^{(t)}$ that are clamped to the visible layer:

$$\begin{aligned} \mathcal{L}_E &= \frac{\partial -\log p(\mathbf{v}^{(t)})}{\partial \theta} \\ &= \partial - \left(\log \sum_{\mathbf{h}} \exp(-E(\mathbf{v}^{(t)}, \mathbf{h})) - \log \sum_{\mathbf{v}, \mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})) \right) / \partial \theta \\ &= \mathbb{E}_{\mathbf{h}} \left[\frac{\partial E(\mathbf{v}^{(t)}, \mathbf{h})}{\partial \theta} \mid \mathbf{v}^{(t)} \right] - \mathbb{E}_{\mathbf{v}, \mathbf{h}} \left[\frac{\partial E(\mathbf{v}, \mathbf{h})}{\partial \theta} \right], \end{aligned} \tag{9}$$

where θ is all the parameters in the case of RBMs, in which $\theta = \{\mathbf{W}, \mathbf{b}_v, \mathbf{b}_h\}$. Free energy is often utilized instead of the original energy function when training RBMs with a negative log-likelihood because free energy is fast to compute and can avoid the extremely small values that exist in the energy function. It is defined as follows:

$$\mathcal{L}_F = -\log p(\mathbf{v}^{(t)}) = F^c(\mathbf{v}^{(t)}) - F, \tag{10}$$

where both F and F^c are free energy and F^c is with the visible units clamped with the observed visible variables (or the training data points $v^{(t)}$). Thus, it is also referred to as the clamped free energy. The definition of free energy is below:

$$\begin{aligned} F(\mathbf{v}) &= -\log Z \\ &= -\mathbf{b}_v^\top \mathbf{v} - \sum_j \log \left(1 + \exp(\mathbf{b}_h + \mathbf{W}^\top \mathbf{v}) \right)_j \end{aligned} \tag{11}$$

The loss function L constructed by the energy or free energy is often denoted as follows:

$$\mathcal{L} = \langle \cdot \rangle_{data} - \langle \cdot \rangle_{model} \tag{12}$$

where $\langle \cdot \rangle$ represents the expectations under the distribution of the data and the model. The first term $\langle \cdot \rangle_{data}$ is called the “positive phase”, and the second term $\langle \cdot \rangle_{model}$ is called the “negative phase”. The positive phase would increase the probability of observing the configurations by reducing the energy (or free energy). Conversely, the negative phase would decrease the probability of samples generated by the model by raising the energy (or free energy).

In RBMs, if they use an energy function to construct the loss function, then the value of the positive phase is estimated by replacing the unknown hidden variables \mathbf{h} in $E(\mathbf{v}^{(t)}, \mathbf{h})$ by first applying the forward inference in Equation (7) to the training data, obtaining $\mathbf{h}^{(0)}$, and then computing the estimated energy expectation $E(\mathbf{v}^{(t)}, \mathbf{h}^{(0)})$. If they use free energy to construct the loss function, then there is no need to estimate \mathbf{h} because free energy does not use hidden layer variables for computation, as shown in Equation (11).

However, the negative phase term is intractable for both the energy function and free energy-constructed loss function because the visible variable \mathbf{v} is unobserved. It is impossible to find all the configurations over (\mathbf{v}, \mathbf{h}) in the energy function or all possible \mathbf{v} in the free energy, which are utilized to compute the expectation of the negative phase term. To make the computation tractable, estimation is carried out using Gibbs sampling [29] in the RBMs as shown in Algorithm 1.

Algorithm 1: Gibbs sampling in RBMs

Input: training data $v^{(t)}$
Result: estimate the negative phase term by a point $\tilde{\mathbf{v}}$
begin
 while $step < N$ **do**
 sample $\mathbf{h}^{(step)} \sim p(\mathbf{h} | \mathbf{v}^{(t)})$;
 sample $\mathbf{v}^{(step)} \sim p(\mathbf{v} | \mathbf{h}^{(step)})$;
 end
 set $\tilde{\mathbf{v}} = \mathbf{v}^{(step)}$
end

The training method CD of RBMs is stated in Algorithm 2. In RBMs, it is proven that one step (step $N = 1$) in Gibbs sampling is enough and effective.

Algorithm 2: CD in RBMs

Input: training data $v^{(t)}$
Result: estimate the negative phase term by a point $\tilde{\mathbf{v}}$
begin
 sample $\mathbf{h}^{(0)} \sim p(\mathbf{h} | v^{(t)})$;
 while $step < N$ **do**
 if $step = 0$ **then**
 sample $\mathbf{v}^{(step+1)} \sim p(\mathbf{v} | \mathbf{h}^{(0)})$;
 else
 sample $\mathbf{v}^{(step+1)} \sim p(\mathbf{v} | \mathbf{h}^{(current)})$;
 end
 sample $\mathbf{h}^{(step+1)} \sim p(\mathbf{h} | \mathbf{v}^{(step+1)})$;
 set $\mathbf{h}^{(current)} \leftarrow \mathbf{h}^{(step+1)}$
 end
 if using energy function constructed loss function \mathcal{L}_E **then**
 update all the parameters θ by:
 set $\theta \leftarrow \theta + \eta \cdot (\partial E(\mathbf{v}^{(t)}, \mathbf{h}^{(0)}) / \partial \theta - \partial E(\mathbf{v}^{(step+1)}, \mathbf{h}^{(step+1)}) / \partial \theta)$
 end
 else if using free energy constructed loss function \mathcal{L}_F **then**
 update all the parameters θ by:
 set $\theta \leftarrow \theta + \eta \cdot (\partial F^c(\mathbf{v}^{(t)}) / \partial \theta - \partial F(\mathbf{v}^{(step+1)}) / \partial \theta)$
end

3.3. Conditional RBMs

Derived from RBMs, conditional-RBMs (CRBMs) have the ability to incorporate extra information. There are varieties of CRBMs in the literature. Most of them are proposed to learn the pattern of temporal sequential data. We chose the simple CRBMs introduced

in [16] (see Figure 2). In this paper, the energy function of the CRBM model is defined as follows:

$$E(\mathbf{v}, \mathbf{h}, \mathbf{u}) = -\mathbf{v}^\top \mathbf{W}_{\mathbf{uh}} \mathbf{h} - \mathbf{v}^\top \mathbf{b}_{\mathbf{v}} - \mathbf{u}^\top \mathbf{W}_{\mathbf{uv}} \mathbf{v} - \mathbf{u}^\top \mathbf{W}_{\mathbf{uh}} \mathbf{h} - \mathbf{h}^\top \mathbf{b}_{\mathbf{h}} \tag{13}$$

The free energy function is defined as follows:

$$F(\mathbf{v}, \mathbf{u}) = -\sum_j \log\left(1 + \exp\left(\mathbf{b}_{\mathbf{h}} + \mathbf{v}^\top \mathbf{W}_{\mathbf{vh}} + \mathbf{u}^\top \mathbf{W}_{\mathbf{uh}}\right)\right)_j - \mathbf{v}^\top \mathbf{b}_{\mathbf{v}} - \mathbf{u}^\top \mathbf{W}_{\mathbf{uv}} \mathbf{v} \tag{14}$$

The training method of CRBMs is the same as that of RBMs: substitute the new energy function into Equation (9) or new free energy function into Equation (10) and apply the CD algorithm to update all the parameters.

Some severe problems of the original CRBM model are the following: (1) The weight matrix $\mathbf{W}_{\mathbf{uv}}$ that connects the visible layer \mathbf{v} and conditional layer \mathbf{u} is extremely large in size if \mathbf{v} and \mathbf{u} are both large, which makes the model impossible to be built on small-scale devices, and (2) there is only one conditional layer that incorporates one kind of extra information, which hinders the model's incorporating more kinds of effective extra information. To address these two problems, we propose a modified CRBM (M-CRBM) in the following section.

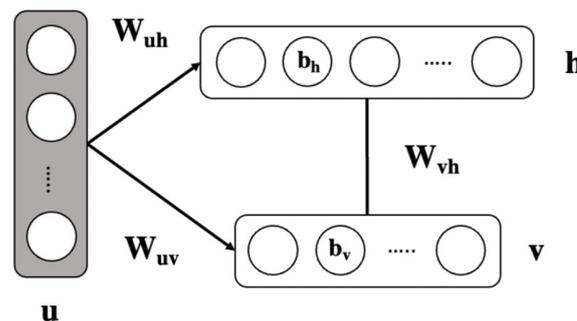


Figure 2. A CRBM with one conditional layer that connects to both the hidden layer and visible layer.

4. Modified CRBMs (M-CRBMs)

We propose M-CRBMs in this paper, which incorporate three conditional layers that can take extra information into account rather than only the observed visible variables, as shown in Figure 3. The conditional layers can easily be added or removed according to the number of types of extra information. In our experiment on a dataset of DAs, three types of extra information were used. Therefore, we implemented a three-conditional-layer M-CRBM in this paper.

Our proposed M-CRBMs are much more scalable on large-size training data and suitable for DAs compared with the conventional CRBMs described in Section 3.3. First, in the query recommendation task, the training data can be extremely large in size, because in query recommendation, the ordinary embedding method is to represent an item (or a URL) with the bag-of-words (BoW) encoding method, where the ones represent the formulated queries that have been used to access that item. Therefore, to represent all the items, the number of dimensions of the representative vector of the item would be the number of all the queries in the access log. Next, the queries were difficult to embed for many of the natural language processing (NLP) models, which often reported efficiently reducing the dimension of the representative vector. The reason for this is that the queries in DAs are special, often being from a unique field, and can easily fail to be embedded well

by those NLP models trained on a general text corpus. Moreover, our proposed M-CRBM has the ability to incorporate more information, such as the metadata, which are abundant in DAs. Lastly, the extra information is considered to alleviate the difficulty of dealing with the sparse query representative vector. This is also one of the key advantages of M-CRBMs when compared with RBMs.

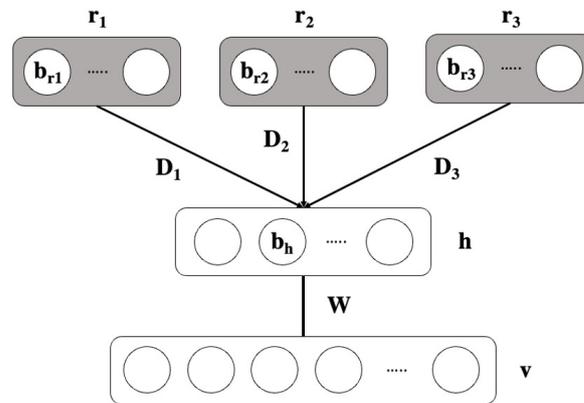


Figure 3. Our proposed M-CRBMs, incorporating three conditional layers that take different types of extra information into account.

Algorithm 3 states our proposed method. The idea of M-CRBMs is to capture the configurations (or dependencies) of the co-occurring queries formulated by high-level experts and low-level expertise users. Each M-CRBM represents an item (an ukiyo-e print), each visible layer unit represents a query, each hidden unit represents a latent feature of the queries, and each conditional unit represents a word from the corresponding metadata field. In this model, we regarded the queries that retrieved the same item as relevant, and these relevant queries were semantically similar. By learning the configurations of high-level and low-level expertise queries from all of the query log, the relevant queries could be detected.

Algorithm 3: CD in M-CRBMs with free energy

Input: training data $v^{(t)}$, conditional data r , learning rate η , initialized parameters θ

Result: update all the parameters θ

begin

while $step < N$ **do**

if $step = 0$ **then**

 sample $\mathbf{v}^{(step+1)} \sim p(\mathbf{v} | \mathbf{h}^{(0)})$;

else

 sample $\mathbf{v}^{(step+1)} \sim p(\mathbf{v} | \mathbf{h}^{(current)})$;

end

 sample $\mathbf{h}^{(step+1)} \sim p(\mathbf{h} | \mathbf{v}^{(step+1)}, \mathbf{r})$;

 set $\mathbf{h}^{(current)} \leftarrow \mathbf{h}^{(step+1)}$

end

 update all the parameters θ by:

 set $\theta \leftarrow \theta + \eta \cdot (\partial F^c(\mathbf{v}^{(t)}) / \partial \theta - \partial F(\mathbf{v}^{(step+1)}) / \partial \theta)$

end

The energy function of our proposed M-CRBMs is defined as follows:

$$\begin{aligned} E(\mathbf{v}, \mathbf{h}, \mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3) &= -\mathbf{v}^\top \mathbf{b}_v - \mathbf{h}^\top \mathbf{b}_h - \mathbf{v}^\top \mathbf{W} \mathbf{h} - \mathbf{r}_1^\top \mathbf{D}_1 \mathbf{h} \\ &\quad - \mathbf{r}_2^\top \mathbf{D}_2 \mathbf{h} - \mathbf{r}_3^\top \mathbf{D}_3 \mathbf{h} \end{aligned} \quad (15)$$

Thus, the free energy is computed as follows:

$$\begin{aligned} F(\mathbf{v}, \mathbf{h}, \mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3) &= -\log \sum_h \exp(-E(\mathbf{v}, \mathbf{h}, \mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3)) \\ &= -\sum_j \log \left(1 + \exp \left(\mathbf{b}_h + \mathbf{v}^\top \mathbf{W} + \mathbf{r}_1^\top \mathbf{D}_1 + \mathbf{r}_2^\top \mathbf{D}_2 + \mathbf{r}_3^\top \mathbf{D}_3 \right) \right)_j \\ &\quad - \mathbf{v}^\top \mathbf{b}_v \end{aligned} \quad (16)$$

We used the free energy constructed loss function as shown in Equation (10), because logarithm operation in free energy avoids extremely small energy values during the training phase. Extremely small energy values could exist due to the model learning too many frequent configurations. This would prevent the model from learning some relatively rare configuration that was meaningful.

The prediction phase of this model is such that, given the input query vector and its extra metadata, it applies forward and backward inference once to reconstruct the input vector. This would assign a probability to each of the visible units that represents the probability that each of the queries is related to the current input.

5. Dataset

We conducted experiments on two datasets: (1) the access log dataset from May 2015 to November 2020 (about five and a half years) of ARC-UPD and (2) the ukiyo-e dataset obtained via the Europeana search API. The details are stated in Sections 5.1 and 5.2.

5.1. ARC-UPD Dataset

Query-item pairs were needed for preparing the training and test datasets for our proposed M-CRBMs. Thus, we only chose the access log records that showed a user formulating a certain query and accessing a certain item (a URL-oriented ukiyo-e print). After the preprocessing, about 150,000 access records remained to be utilized in the subsequent training and test dataset construction. In the experiments, the M-CRBMs that took three types of extra information had around 269 million variables.

5.1.1. Preprocessing

In the preprocessing phase, we performed the following operations on the raw access log data to obtain the query-item pairs, which were used to find configurations for training the M-CRBMs:

1. We removed all the web robots' access by recognizing the agent names that appeared in the access logs. If the agent name of the record was listed in that web robot list, then the record was removed. The suspicious web robot list was provided by the Apache HTTP Server, which is the front-end web server of the ARC-UPD.
2. We removed records from users who frequently accessed (accessed more than 50 times within a time interval of less than 10 s) the ARC-UPD, because these users were perhaps scraping the web page or checking the system. In this work, a unique user is defined by the same IP address and browser (user agent) information.
3. We only kept the records that indicated a user used the formulated query (or queries) to access a certain item. In the search engine of the ARC-UPD, a user can input search queries in multiple search fields, such as keyword, artist, publication year, and genre. Although all of the queries to different search fields include extensive expert

knowledge, in this research, we only used the queries input in the keyword search field. The reason for this is due to the difficulties of extracting the needed training data from other fields. In the search system's design, if the users click some metadata shown on the web page and then access a certain item from the access log, it will be recorded as the same as users making a search query in the corresponding metadata field and accessing a certain item. Therefore, only choosing the queries input in the keyword search field would ensure that most of the queries were really formulated by real users.

4. We constructed a dictionary to store the query-item pairs from the previous steps with the format of

$$\{itemID : \{query : frequency, \dots\}, itemID : \{query : frequency, \dots\} \dots\}$$

We call this a query dictionary. Here, "frequency" is the frequency of the query that has been used to search for the item.

5.1.2. Characteristics of the Dataset

Some of the characteristics of the dataset were as follows. First, if it was defined that two consecutive access instances were within a maximum of 15 min in one session, then the sessions of the ARC-UPD often included very few queries with clicked items (which means a user formulated some queries and clicked any items in a session). According to the statistics, 50.55% of the sessions included only one query with clicked item access, and 72.12% of the sessions included three or fewer queries with clicked item access.

Second, we can reasonably speculate that there are more users of the ARC-UPD with high-level expertise. The ARC-UPD was developed from the beginning for experts. Thus, in theory, most users are experts. We conducted a simple analysis to see the expertise level of all the queries. We first calculated the average number of characters contained in each keyword in the queries (if there was a space or spaces in a query, then it was regarded as multiple keywords), which was 3.52. For a simple calculation, the keywords with more than 3.52 characters were regarded as high-level expertise queries, and among the keywords with less than 3.52 characters, we checked whether these keywords existed in the ukiyo-e term dictionary (an expert-edited dictionary). If so, then they would also be regarded as high-level expertise queries. According to such a setting, the high-level expertise keywords accounted for 87.94% of all the keywords.

5.1.3. Training and Test Datasets

There were two types of datasets required for our experiments. One was the item embeddings that were constructed by their related queries, which were used as the input for the visible layers of the M-CRBMs. Another was the item embeddings that were constructed by the extra information, which were used as the input for the conditional layers of the M-CRBMs. We used at most three types of metadata as the extra information in this work: artists, genres, and series names of ukiyo-e prints. Each item had its query embedding and its corresponding extra information embedding(s). After these two types of datasets were prepared, they were randomly split, with 80% in the training dataset and 20% in the test dataset.

Query Embedding

To obtain the embeddings of the queries, we constructed a vector for each item. The number of dimensions of this vector was equal to the number of all the existing distinct queries in the query dictionary that we created in the preprocessing step. Each dimension represented the existence of a query. For a vector that represented an item, there would be numbers existing at the dimension(s) where the represented query (queries) was (were) used for searching the item. The numbers were the frequencies referred to in the query dictionary. The remainder dimensions would be zeros. Then, the vector was normalized by min-max normalization. An example is shown in Figure 4.

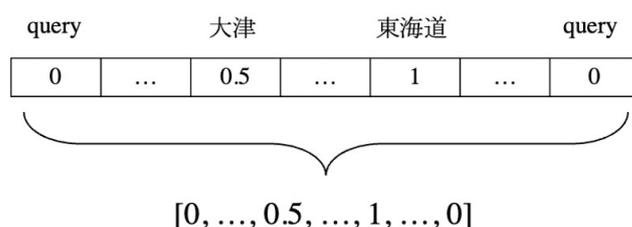


Figure 4. An example of the query embedding. Here, the item was searched and accessed by “大津” (Otsu) once and by “東海道” (Tokaido) twice.

Since this vector was not binary (the required input of all binary RBMs and CRBMs), we would first obtain the binary vectors sampled from a Bernoulli distribution as stated in Algorithm 3.

Artist Embedding

The operation used to obtain the embeddings of the field artist metadata was similar to the query embeddings. Each item had a binary encoded vector that represented its artist information. The dimensions of the vectors were the number of all the existing artists in the DA. Each dimension of a vector represents an artist. If the item was created by a certain artist, then the value at that dimension would be ones, and the remaining values would be zeros. Since the artist embedding was already binary, we did not need to perform a sampling operation from a Bernoulli distribution.

Genre Embedding

The embedding of the genre metadata field was the same as the embeddings of the artist’s metadata. The only difference was in using a different type of information.

Series Name Embedding

The embeddings of the series name metadata (In the ARC-UPD, the series names and title were actually saved in one metadata field. In this paper, we only used the series names under the series name field for convenience.) was slightly different from that for the artist or genre metadata. We first segmented the series name of an ukiyo-e print into tokens, because we considered that there was shared information in different series names. For example, there are many series names including “八犬伝” (Hakkenden), such as “見立八犬伝” (Mitate Hakkenden), “八犬伝犬のさうしの内” (Hakkenden ken no Sousinouchi), and “里見八犬伝” (Satomi Hakkenden). If we represented all those series names as individual dimensions in vectors, then they would share no similarities. By first separating the series names into words with an ukiyo-e dictionary, this would intuitively help the model find the appropriate configurations.

The tokenization was performed by utilizing Japanese morphological analysis system MeCab (<https://taku910.github.io/mecab/>, accessed on 22 December 2022) with the dictionary Kinsei kōgo Unidic (https://clrd.ninjal.ac.jp/unidic/back_number.html#unidic_kinsei-edo, accessed on 22 December 2022) and a user dictionary created by ukiyo-e experts that includes related ukiyo-e proper nouns in the field.

Aside from tokenization, the embedding of series names is similar to the embedding of other metadata fields, which uses binary encoding to represent whether a token exists or not.

5.2. Europeana Dataset

To test the versatility of our proposed model, we collected an ukiyo-e dataset via the Europeana search API (<https://pro.europeana.eu/page/search>, accessed on 22 December 2022) and conducted an experiment on it. The dataset was obtained and processed as indicated in the following steps. In this experiment, the M-CRBMs had around 9 million variables:

1. Send the query “ukiyo” to the Europeana search API and obtain all the returned records. In this dataset, only the metadata field title was considered suitable conditional information for the proposed M-CRBM model.
2. Filter the returned records by record language. Here, we chose to only use records in the German language because the size of this dataset was small, and most records were in the German language. Only using records in German ensured that the model could efficiently learn the patterns of the co-occurrence of words. After filtering, 4089 records remained.
3. Perform basic natural language preprocessing, remove punctuation marks, numbers, and stop words, remove the often-appearing words that are not related to the items, and convert all the characters to lowercase.
4. Because there were no real access logs for this dataset, we randomly generated the access logs. The random rule was that each item was accessed 1–10 times. The query (queries) to access a certain item was (were) assumed to be 30~50% of the words in the title of the item (less than one word was counted as one). We generated a total of 22,650 query-item access logs.
5. Query embedding and title embedding were created for the Europeana dataset. The embedding method was similar to artist embedding in the ARC-UPD. The only difference was that in the Europeana dataset, each dimension represented one word in queries or titles.

6. Experiments

As mentioned in Section 5, we conducted experiments on the ARC-UPD and Europeana dataset. The experiments on the ARC-UPD were mainly to check whether our proposed M-CRBMs outperformed the original RBM, whether utilizing metadata contributed to this task, and moreover, which metadata contributed the most. The experiment on the Europeana dataset was mainly to check the versatility of the proposed model for the other dataset of DAs.

6.1. Evaluation Metrics

In the experiments, we used the mean square error (MSE) to evaluate the performance of the utilized training method: CD with free energy. Since CD with free energy minimized the defined free energy but did not directly minimize the prediction error, to evaluate whether this training method was effective on our dataset, we first checked whether the MSE decreased or not. If the MSE decreased, then this showed that the proposed training method was effective at minimizing the prediction error. The formula for the MSE is stated below:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2, \quad (17)$$

where n is the total amount of training data, Y_i is the ground truth for every training data, and \hat{Y}_i is every prediction when the model takes the training data as input.

We used precision@k, recall@k and F1@k to evaluate the ability of the proposed method to retrieve the relevant queries. They are stated below:

$$\text{Precision@k} = \frac{|\text{hits in top } k|}{|k|}, \quad (18)$$

$$\text{Recall@k} = \frac{|\text{hits in top } k|}{|\text{all relevant items}|}, \quad (19)$$

$$\text{F1@k} = \frac{2 \cdot \text{Precision@k} \cdot \text{Recall@k}}{\text{Precision@k} + \text{Recall@k}}, \quad (20)$$

where k means k items from the top-ranked recommendation items were retrieved. In the experiments, $k = \{1, 5, 10, 20\}$. The term *hits in top k* in precision@ k and recall@ k is the number of correctly recommended items. The term *relevant items* is the size of the test data.

6.2. Experiments on ARC-UPD Dataset

For the ARC-UPD, we evaluated the models with the following metrics:

1. Mean square error (MSE), which measures the effectiveness of the training method of the proposed M-CRBM model;
2. Recall@ k , precision@ k , and F1@ k to measure the model performance in a recommendation task;
3. Ablation experiments that compared the conventional RBMs with M-CRBMs, which took different extra information in the conditional layer(s);
4. Recommendation examples, including the predictions of all the models by inputting the same query and typical bad cases.

Experimental Results

Figure 5 shows the MSE scores through the training epochs. The MSE score increased in the early epochs and then dropped sharply until it converged. The MSE score's objective was to indicate the training rule defined by us was effective.

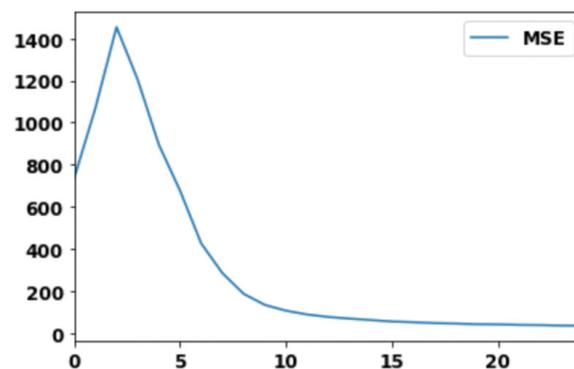


Figure 5. MSE through 25 training epochs of the proposed M-CRBMs that take three types of extra information in the conditional layers.

RBMs and CRBMs are mostly used as generative models. One of the general testing methods of the generative task is to input the original test data without masking part of them. This is to see whether the model has the ability to reconstruct the input data and also to measure whether the hidden units extract meaningful features that are related to the configurations. A characteristic existing in our dataset was that many of the items had only been searched by one query word. If they were masked in the test data, then the input would be null. Therefore, in this experiment, we did not mask part of the input data and saw whether the model could predict it but used the aforementioned method to calculate the recall, precision, and F1 scores of the test data. Then, we show some recommendation examples to prove that our proposed model found meaningful related queries.

In Figure 6, the recall, precision, and F1 scores for @1, @5, @10, and @20 are shown. We found that, although the precision@1 and F1@1 were not stable in the early training epochs, they became stable and outperformed the other metrics when the model converged. One of the reasons that the precision@1 and F1@1 scores were high in the experiments might also be that many of the inputs in the test data only included a single query, which means there could only be one hit. Despite that, this result still proves the ability of the proposed model for learning the important features of an input query (or queries) well.

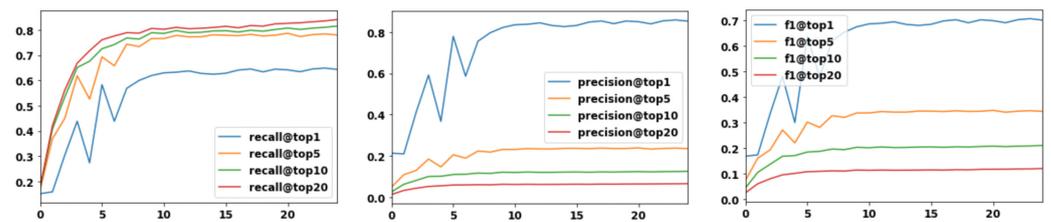


Figure 6. Recall@k, precision@k, and F1@k through 25 training epochs of the proposed M-CRBMs that take three types of extra information into account.

The ablation experiments were conducted to answer the questions “Is the extra information useful?” and “What kind of extra information will be more useful?”.

In the experiments, for all types of M-CRBM models (M-CRBM models utilize different extra information), the training data and test data that were clamped to the visible layer were the same, but different types of M-CRBM models take different extra information that corresponds to the visible layer. All types of M-CRBM models were trained with the same parameter settings, except for the number of hidden units. Because one hidden unit extracted one configuration among different input data, it should have been changed when the input changed. Especially when the model took genre information into account, it became hard to train and often had the gradient explosion problem. This might be because genre information is too weighted and does not contribute to detecting the configurations. We set the number of hidden units as the most appropriate for each model. All type of M-CRBM models were trained for 25 epochs.

We compared the recall@1, precision@1, and F1@1 of the M-CRBM models that utilized different extra information. The results for the metrics are listed in Table 1.

Table 1. Recall@1, precision@1, and F1@1 of M-CRBM models clamped with different extra information. The results were taken from the final training epoch. The bold text indicates the best results in the column, and the underlined values are the second- and third-best results in the column.

Model	Recall@1	Precision@1	F1@1
RBM	0.685317	0.898908	0.744208
M-CRBMs (a)	<u>0.705432</u>	<u>0.925713</u>	<u>0.766092</u>
M-CRBMs (g)	0.684667	0.900585	0.744076
M-CRBMs (s)	0.723705	0.954496	0.787330
M-CRBMs (a+g)	0.687901	0.904879	0.747577
M-CRBMs (a+s)	<u>0.715931</u>	<u>0.938779</u>	<u>0.777305</u>
M-CRBMs (g+s)	0.663306	0.876736	0.722109
M-CRBMs (a+g+s)	0.642967	0.852156	0.700563

a = artist, g = genre, s = series name.

The results show that, compared with the RBM model, adding the artist or series name information contributed to the results, but the genre information made the results worse. The series name information contributed the most to all kinds of metrics. This might be because experts are prone to using series names and some subword included in the series names to construct the query (or queries). Using both the artist and series name information could improve from only using the artist information, but this was slightly lower than only using series name information. Adding genre information proved to be a bad influence on all single or combined types of information, which might be because experts rarely use genres to formulate a query (or queries).

We further checked some of the representative top 10 query recommendation results of a certain input of each model. The input of all the models was the typical query “月百姿” (meaning “one hundred aspects of the moon”, a famous book series of ukiyo-e prints that includes stories about moons, which was also used as a series name in the ARC-UPD). The recommendation queries predicted by each model are shown in Table 2. We regarded all the recommended queries that appeared together with “月百姿” in any metadata fields

of the same ukiyo-e as relevant. For example, there is an ukiyo-e print with the series name of “月百姿” and with the title of “淮水月 伍子胥” (meaning “moon of Huaishui River, Wu Zixu (who was a Chinese military general and politician of the Wu kingdom)”). Thus, “淮水月” is a relevant query to “月百姿”.

The recommendation results show that, similar to the recall, precision, and F1 score results, M-CRBMs (s) (the M-CRBMs using series names) performed the best. M-CRBMs (s) can give abundant relevant query recommendations. The results also show an interesting finding: the recommendation results were relevant to the metadata field(s) that was (were) used in the M-CRBMs. For example, M-CRBMs that utilize genre information can give some recommendations about the genre (such as “忠臣蔵” in M-CRBMs (g), M-CRBMs (a+g), and M-CRBMs (a+g+s)). M-CRBMs that utilize artist information can give some recommendations about the artist (such as “芳年...” in M-CRBMs (a)). M-CRBMs that utilize series name information can give some recommendations about the series name (such as “孝子の月” in M-CRBMs (s), M-CRBMs (a+s), and M-CRBMs (a+g+s)). There existed some noise recommendations, which might have been caused by random sampling in the training and prediction phase, but this could be removed by some processing before developing a system.

Table 2. Top 10 recommended queries for each model given the input query “月百姿”. The counts of the relevant recommended queries are shown in the relevant number column, and the underlined queries are the relevant ones.

Model	Query Recommendations	Relevant Number
RBM	月百姿, 大坂, 二十四孝, 五十三次之内, <u>金太郎</u> , 鬼, 人形, <u>孝子の月</u> , 井筒, 堀川	3
M-CRBMs (a)	月百姿, 玉手箱, 忠臣蔵, 芳年..., 目黒不動境内, 団扇絵, 祇園, 奥州安達, 季翫, 風俗	4
M-CRBMs (g)	月百姿, 春, 役者絵, 義士, 仮名手本忠臣蔵, 美人, 名所絵, <u>忠臣蔵</u> , 満堯, 之肖像	3
M-CRBMs (s)	月百姿, 淮水月, 武者絵, 祇園まち, 五条橋の月, <u>金時</u> , 小幡小平次, <u>孝子の月</u> , 月, 宝蔵院	9
M-CRBMs (a+g)	月百姿, 遊君五節生花会, 忠臣蔵, 大星, 仮名手本忠臣蔵, 八百万神..., 駿河国富士川合戦, 子別れ, 金剛神之図, 孝女	2
M-CRBMs (a+s)	月百姿, 邑増山..., 祇園まち, 平井保昌, 月, 淮水月, 小町, 大日本史略図会, 鳶, <u>孝子の月</u>	7
M-CRBMs (g+s)	月百姿, 講, 妻恋稲荷, 逢坂関, 月, 十兵衛, <u>孝子の月</u> , 高の師直, 東都宮戸川之図, 武者	2
M-CRBMs (a+g+s)	月百姿, 伊勢..., 徒然草, の紅葉手古那, 孝子の月, 真乳山山谷堀夜景, 染物, 江戸花柳橋名取, <u>忠臣蔵</u> , 面壳	3

Note: “...” stands for the omission of queries that were too long.

Two of the typical bad cases of M-CRBMs are listed in Table 3. In the first case, when the input query was “程義経”, we could obtain nine relevant output queries. However, many of them were partially duplicated. Such a case exists when there are several partially duplicated queries that are made in the access log. Sometimes it is good to recommend variants of similar queries, but sometimes it may be redundant. We need to find a filtering method to remove part of the redundant partial duplicated contents. Another case was when the model could not find any relevant queries. One of the reason for this result is that the input query “比叡山” may not appear with a large number of other queries, so there is no obvious co-occurrence relationship with other queries, and the recommendation yields very random results.

Table 3. Typical bad cases of M-CRBMs (s).

Index	Input	Outputs	Relevant Number
1	程義經	程義經恋源, 程義經, 義經, 義經恋, 武者, 横川の堪海, 小町, 弁慶, 牛若御曹司, 義經記	9
2	比叡山	比叡山, 忠臣蔵, 深草少将, 燕青, 中村芝翫, 小町, 藤原保友, 假名手本忠臣蔵, 衣紋坂, 美人画	1

6.3. Experiments on Europeana Dataset

For the Europeana dataset, we evaluated the models for the recall@k, precision@k, and F1@k to measure the model’s performance in the recommendation task. The results are shown in Figure 7.

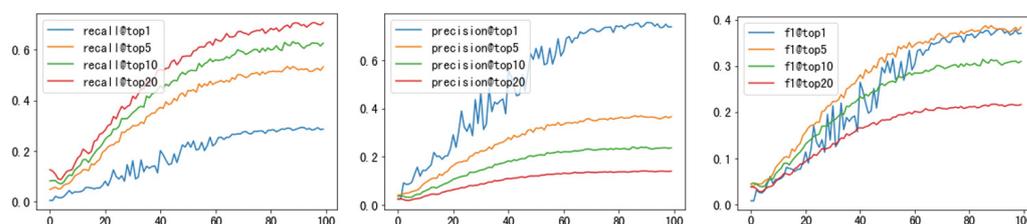


Figure 7. Recall@k, precision@k, and F1@k through 100 training epochs of the proposed M-CRBMs that take the Europeana dataset into account.

From the results, we can see that as the number of training epochs increased, the recall, precision, and F1 score all improved. This shows that our proposed model and training method are also applicable to the Europeana dataset. However, the performance on the Europeana dataset was not as good as that on the ARC-UPD. There are probably two reasons for this: (1) the amount of data in the dataset was too small, and (2) it lacked real user-formulated queries to extract real co-occurrence patterns. If these two difficulties can be overcome, then it is considered that our proposed model can be applied well to the Europeana dataset or other similar DA datasets.

7. Conclusions and Discussion

In this work, we proposed an M-CRBM for query recommendation in DAs. The experimental results prove that the model is able to give meaningful, semantically similar recommendations with the proposed M-CRBMs. The M-CRBM model has the potential to support users coming to DAs with non-expert-level knowledge.

In the future, some similar deeper models can be tested for this task, such as deep Boltzmann machines or deep belief networks, which could be considered better for extracting the dependencies (or configurations) among the units. Taking trust weights among the users into consideration is also an expected direction. Some studies have mentioned how to efficiently utilize trust weights in recommender systems [30–32]. In the context of query recommendation in DAs, different users are interested in different topics, so when the same query is input, the obtained query recommendations should be different according to the user’s preferences. Different from simply adding collaborative filtering (CF), which only considers whether users have similar preferences for certain items, trust-based recommendation allows users interested in different topics, which could tolerate different preferences for different items. In the ARC-UPD used in this paper, the users’ interest in different topics and trust weights can be extracted from the access log data. Adding these to the current recommendations can enable users to obtain more personalized recommendation results. For DAs associated with online social networks, the user relationship information can be extracted more easily. The user relationships can be used as learning data for trust weights to improve personalization.

Author Contributions: Conceptualization, J.W., A.M. and K.K.; Methodology, K.K. and J.W.; Software, J.W.; Resources, B.B. and R.A.; Writing—original draft, J.W.; Writing—review & editing, B.B. and A.M.; Funding acquisition, A.M. and R.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by JSPS KAKENHI Grant Number 20K12567.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Pavlidis, G.; Koutsoudis, A.; Arnaoutoglou, F.; Tsioukas, V.; Chamzas, C. Methods for 3D digitization of cultural heritage. *J. Cult. Herit.* **2007**, *8*, 93–98. [CrossRef]
- Johnson, P.S.; Doulamis, A.; Moura Santo, P.; Hadjiprocopi, A.; Fritsch, D.; Doulamis, N.D.; Makantasis, K.; Stork, A.; Ioannides, M.; Klein, M.; et al. Online 4D reconstruction using multi-images available under Open Access. In Proceedings of the SPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Strasbourg, France, 30 July 2013.
- Philips, J.; Tabrizi, N. Historical Document Processing: A Survey of Techniques, Tools, and Trends. In Proceedings of the 2th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, Online, 2 November 2020.
- Clark, J. H. The long-term preservation of digital historical geospatial data: A review of issues and methods. *J. Map Geogr. Libr.* **2016**, *12*, 187–201. [CrossRef]
- Resig, J. Using computer vision to increase the research potential of photo archives. *J. Digit. Humanit.* **2014**, *3*, 5–36.
- Phillips, S. C.; Walland, P. W.; Modafferi, S.; Dorst, L.; Spagnuolo, M.; Catalano, C. E.; Oldman, D.; Tal, A.; Shimshoni, I.; Hermon, S. GRAVITATE: Geometric and Semantic Matching for Cultural Heritage Artefacts. *GCH* **2016**, *16*, 199–202.
- Ardissono, L.; Kuflik, T.; Petrelli, D. Personalization in cultural heritage: The road travelled and the one ahead. *User Model. User-Adapt. Interact.* **2012**, *22*, 73–99. [CrossRef]
- Wilson-Barnao, C. How algorithmic cultural recommendation influence the marketing of cultural collections. *Consum. Mark. Cult.* **2017**, *20*, 559–574. [CrossRef]
- Clough, P.; Hill, T.; Paramita, M. L.; Goodale, P. Europeana: What users search for and why. In Proceedings of the International Conference on Theory and Practice of Digital Libraries, Thessaloniki, Greece, 18–21 September 2017.
- Europeana. Available online: <https://www.europeana.eu/portal/en> (accessed on 6 November 2022).
- ukiyo-e Portal Database1 of Art Research Center (ARC-UPD). Available online: https://www.dh-jac.net/db/nishikie/\search_portal.php?enter=portal&lang=en (accessed on 6 November 2022).
- Hinton, G.E.; Osindero, S.; Teh, Y.W. A fast learning algorithm for deep belief nets. *Neural Comput.* **2006**, *18*, 1527–1554. [CrossRef] [PubMed]
- Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [CrossRef] [PubMed]
- Taylor, G.W.; Hinton, G.E.; Roweis, S.T. Modeling human motion using binary latent variables. *Adv. Neural Inf. Process. Syst.* **2007**, *19*, 1345–1352.
- Salakhutdinov, R.; Mnih, A.; Hinton, G. Restricted Boltzmann machines for collaborative filtering. In Proceedings of the 24th International Conference on Machine Learning, Buenos Aires, Argentina, 20 June 2007.
- Mnih, V.; Larochelle, H.; Hinton, G.E. Conditional restricted boltzmann machines for structured output prediction. *arXiv* **2012**, arXiv:1202.3748.
- Wang, Y.; Stash, N.; Aroyo, L.; Gorgels, P.; Rutledge, L.; Schreiber, G. Recommendations based on semantically enriched museum collections. *J. Web Semant.* **2008**, *6*, 283–290. [CrossRef]
- Semeraro, G.; Lops, P.; De Gemmis, M.; Musto, C.; Narducci, F. A folksonomy-based recommender system for personalized access to digital artworks. *J. Comput. Cult. Herit.* **2012**, *5*, 1–22. [CrossRef]
- Georgiev, K.; Nakov, P. A non-iid framework for collaborative filtering with restricted boltzmann machines. In Proceedings of the International Conference on Machine Learning, Atlanta, GA, USA, 26 May 2013.
- Wu, X.; Yuan, X.; Duan, C.; Wu, J. A novel collaborative filtering algorithm of machine learning by integrating restricted Boltzmann machine and trust information. *Neural Comput. Appl.* **2019**, *31*, 4685–4692. [CrossRef]
- Pujahari, A.; Sisodia, D.S. Modeling side information in preference relation based restricted boltzmann machine for recommender systems. *Inf. Sci.* **2019**, *490*, 126–145. [CrossRef]
- Chen, Z.; Ma, W.; Dai, W.; Pan, W.; Ming, Z. Conditional restricted Boltzmann machine for item recommendation. *Neurocomputing* **2020**, *385*, 269–277. [CrossRef]

23. Baeza-Yates, R.; Hurtado, C.; Mendoza, M. Query recommendation using query logs in search engines. In Proceedings of the International Conference on Extending Database Technology, Berlin, Heidelberg, 14 March 2004.
24. Huang, C.K.; Chien, L.F.; Oyang, Y.J. Relevant term suggestion in interactive web search based on contextual information in query session logs. *J. Am. Soc. Inf. Sci. Technol.* **2003**, *54*, 638–649. [[CrossRef](#)]
25. Song, Y.; He, L.W. Optimal rare query suggestion with implicit user feedback. In Proceedings of the 19th International Conference on World Wide Web, Raleigh, NC, USA, 26 April 2010.
26. Feild, H.; Allan, J. Task-aware query recommendation. In Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin Ireland, 28 July 2013.
27. LeCun, Y.; Chopra, S.; Hadsell, R.; Ranzato, M.; Huang, F. Energy-Based Training: Architecture and Loss Function. In *Predicting Structured Data*; Bakir, G., Hofman, T., Scholkopf, B., Smola, A., Taskar, B., Eds.; The MIT Press: London, UK, 2007; pp. 197–205.
28. Hinton, G.E. Training products of experts by minimizing contrastive divergence. *Neural Comput.* **2002**, *14*, 1771–1800. [[CrossRef](#)]
29. Gelfand, A.E. Gibbs sampling. *J. Am. Stat. Assoc.* **2000**, *95*, 1300–1304. [[CrossRef](#)]
30. Dadgar, M.; Hamzeh, A. How to boost the performance of recommender systems by social trust? Studying the challenges and proposing a solution. *IEEE Access* **2022**, *10*, 13768–13779. [[CrossRef](#)]
31. Meo, P.D. Trust prediction via matrix factorisation. *ACM Trans. Internet Technol.* **2019** *19*, 1–20. [[CrossRef](#)]
32. Nikolakopoulos, A.N.; Ning, X.; Desrosiers, C.; Karypis, G. Trust your neighbors: A comprehensive survey of neighborhood-based methods for recommender systems. In *Recommender Systems Handbook*; Francesco, R., Lior, R., Bracha, S., Eds.; Springer: New York, NY, USA, 2021; pp. 39–89.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.