

Finding Concepts with Unexpected Multi-Label Objects Based on Shared Subspace Method

HongJie Zhai, Makoto Haraguchi, Yoshiaki Okubo, and HeYu Liu

Graduate School of Information Science and Technology
Hokkaido University
N-14 W-9, Sapporo 060-0814, Japan
mh@ist.hokudai.ac.jp

Abstract. We discuss a method of retrieving *unexpected* objects for a given query, where each data object is represented as a feature vector and assigned a multi-label as well. Given an object-feature matrix X_1 and an object-label matrix X_2 , we try to *simultaneously* factorize X_1 and X_2 as $X_1 \approx BV$ and $X_2 \approx SW$ by means of *Nonnegative Shared Subspace Method*, where the basis S is a part (subspace) of the basis B . With the help of the shared subspace, thus, we can predict a multi-label for a query feature-vector with unknown labels. Our unexpected object for the query is defined as an object which is similar to the query in the feature space, but is dissimilar in the label space. In order to obtain unexpected objects from several viewpoints of similarity, we formalize our retrieval task as a problem of finding formal concepts satisfying a constraint w.r.t. the unexpectedness. We present an efficient depth-first branch-and-bound algorithm for extracting our target concepts.

Keywords: formal concept, shared subspace method, nonnegative matrix factorization, unexpectedness of objects, multi-labels, recommendation

1 Introduction

Information Retrieval (IR) is a fundamental task in our daily life. In popular keyword-based IR, since it is not easy to get desirable data objects by providing query keywords just once, we *iteratively* input queries until we can meet satisfiable results. Particularly, in *Associative Search* [9], at each step we repeatedly input a query, our query is shifted to its sibling concept [10]. As the results, we often find an interesting search result which is *surprising* or *unexpected* for us but still keeps a certain degree of relevance to our initial query. The authors consider that such an aspect of associative search is strongly desirable especially for *recommendation-oriented IR systems*. This paper discusses a recommendation-oriented method for finding interesting objects for a given query, especially taking an *unexpectedness* of objects for the query into account.

A notion of unexpectedness in recommender systems has been discussed in [14]. In the framework, the unexpectedness of an item (object) is evaluated

based on a distance between the item and those a user already knows in some sense. Another related notions, *novelty* and *serendipity*, have also been investigated in [16]. An object is said to be novel for a user if he/she does not know it. For example, it can be evaluated based on user’s history of recommendations. On the other hand, since the notion of serendipity is emotional and difficult to be defined, it has been discussed in terms of *diversity* which is based on dissimilarity among objects [16]. It is noted here that those notions previously proposed are *subjectively* defined because we need some kind of *user-dependent* information.

In contrast with them, we propose an *objective* unexpectedness of objects. A data object is usually represented as a vector in a primary feature space. Then, the notions of novelty and serendipity have been formalized with the help of additional information specific to particular users. Nowadays, however, several kinds of additional information are also *commonly* available. In case of movie objects, for example, each movie would be primarily represented as a vector of feature terms extracted from their plots. In addition, each movie is often assigned some *genre labels* by commercial companies or many SNS users. Since those secondary features provide us valuable information about movies, they would make our IR systems more flexible and useful for a wide range of users.

In our framework, as such commonly-available additional features, we assume each object is assigned some *labels* (as a multi-label) beforehand. That is, our data objects are given by two data matrices, X_1 and X_2 , each of which represents an *object-feature* relation and *object-label* relation, respectively. Then, we propose our notion of unexpectedness *with respect to label-information of objects*.

More concretely speaking, in our recommendation-oriented IR, a query q is given as a feature vector and supposed to have no label. As a reasonable guess, we often consider that if an object x is similar to q in the feature space, q would also have a multi-label similar to that of x . Conversely, if we observe (by any means) their multi-labels are far from each other, we would find some unexpectedness of x for q because they have distant multi-labels even though their features are similar. Based on the idea of unexpectedness, we formalize our IR task as a problem of detecting *formal concepts* [12] each of which contains some unexpected objects in the extent. By finding those formal concepts, we can obtain our search results from various viewpoints of similarity among the query and objects.

The point in our IR is to predict a multi-label of the query represented as a feature-vector. For the task, our object-feature matrix X_1 and object-label matrix X_2 are *simultaneously* factorized as $X_1 \approx BV$ and $X_2 \approx SW$ by *Non-negative Shared Subspace Method* [1], where the basis S is a part (subspace) of the basis B . In a word, such a shared subspace associates the label-information with the feature-information of the original matrices. With the shared subspace, we can predict a multi-label for the query feature-vector with unknown labels.

To predict a multi-label of a given object, a method of *multi-label classification* has already been proposed in [8]. In the method, we need to obtain a subspace and its orthogonal basis for the original feature space which approximately reflects similarity among multi-labels assigned to the objects by solving

an eigenvalue problem. However, such an orthogonal basis yields negative components in the subspace which complicate interpretation of our search results. Moreover, orthogonality is not necessarily required in prediction purpose. As its simplified and efficient version, a prediction method has also been discussed in [17], where a subspace is just defined as a *real line*. However, the prediction is not so reliable as we will see later. This is the reason why we prefer nonnegative factorization in our framework.

In our recommendation-oriented IR, for a given query, we try to find formal concepts whose extents contains some unexpected objects for the query. We first create a formal context consisting of only objects similar (relevant) to the query in the feature space with a standard *Nonnegative Matrix Factorization* [2]. Then, we try to extract concepts with unexpected ones in the context. Since we often have a huge number of concepts, we evaluate a concept with its extent E by the average distance between each object in E and the query in the label-subspace, and try to extract concepts with the top- N largest evaluation values. We present a depth-first algorithm for finding those top- N concepts, where a simple branch-and-bound pruning based on the evaluation function is available. Our experimental result for a movie dataset shows our system can actually detect an interesting concept of movies whose plots are similar to a given query but some of them have genre-labels which are far from predicted genres of the query.

From the viewpoint of *Formal Concept Analysis (FCA)* [12], our study in this paper is closely related to several interesting topics. In order to reduce the size of formal context preserving important information, methods with non-negative matrix factorizations have been investigated in [3, 5]. Although our method is also based on such a factorization technique, the main purpose is not only to reduce the context but also to associate label information with the reduced context.

A smaller lattice with a reduced number of concepts is desirable as a practical requirement in FCA. Computing a subset of possible concepts, e.g., in [6], is a useful approach for that purpose. Interestingness measures of concepts can meaningfully restrict our targets to be extracted and several representatives are surveyed in [4]. Although our method also proposes a kind of interestingness based on unexpectedness of objects, we emphasize that it is a query-specific one.

2 Approximating Data Matrices Based on Nonnegative Shared Subspace Method

In this section, we discuss how to *simultaneously* approximate a pair of data matrices representing different informations of the same (common) objects. The approximation is based on *Nonnegative Shared Subspace Method* [1].

2.1 Approximating Object-Feature Matrix Reflecting Multi-Label Information

Let $X_1 = (f_1 \cdots f_N)$ be an $M \times N$ *object-feature* matrix and $X_2 = (\ell_1 \cdots \ell_L)$ an $M \times L$ *object-label* matrix, where each object is represented as a row-vector.

As will be discussed later, in our experimentation, movies are regarded as objects and terms (word) in their plots as features. In addition, each movie is assigned a set of genre-labels as its multi-label.

Since each object is often represented as a high-dimensional feature vector, it would be required to compress the matrix X_1 . Moreover, although the number of possible labels would be less than that of features, the matrix X_2 also tends to be sparse because each object has only a few labels in general. We, therefore, try to compress both X_1 and X_2 by means of *Nonnegative Matrix Factorization* [2].

More formally speaking, X_1 and X_2 are approximated as follows:

$$X_1 \approx (\tilde{f}_1 \cdots \tilde{f}_K) V, \quad \text{where } V = (v_{ij})_{ij}, \quad f_j \approx \sum_{i=1}^K v_{ij} \tilde{f}_i$$

$$X_2 \approx (\tilde{\ell}_1 \cdots \tilde{\ell}_{K_L}) W_L, \quad \text{where } W_L = (w_{ij}^L)_{ij}, \quad \ell_j \approx \sum_{i=1}^{K_L} w_{ij}^L \tilde{\ell}_i.$$

It is noted here that $(\tilde{f}_1 \cdots \tilde{f}_K)$ is a compressed representation of X_1 and $(\tilde{\ell}_1 \cdots \tilde{\ell}_{K_L})$ that of X_2 . As has been stated previously, we especially try to use the latter matrix $(\tilde{\ell}_1 \cdots \tilde{\ell}_{K_L})$ as a part of the former $(\tilde{f}_1 \cdots \tilde{f}_K)$ in order to associate the label-information with the feature-information in our approximation process. That is, assuming a $(K_F = K - K_L) \times N$ coefficient matrix V_F and a $K_L \times N$ coefficient matrix V_L , we try to perform approximations such that

$$X_1 \approx (\tilde{f}_1 \cdots \tilde{f}_{K_F} \tilde{\ell}_1 \cdots \tilde{\ell}_{K_L}) \begin{pmatrix} V_F \\ V_L \end{pmatrix} = (\tilde{f}_1 \cdots \tilde{f}_{K_F}) V_F + (\tilde{\ell}_1 \cdots \tilde{\ell}_{K_L}) V_L,$$

$$X_2 \approx (\tilde{\ell}_1 \cdots \tilde{\ell}_{K_L}) W_L.$$

Note that the original column-vector f_i of X_1 is approximated by a linear combination of basis vectors \tilde{f}_j in $F = (\tilde{f}_1 \cdots \tilde{f}_{K_F})$ and $\tilde{\ell}_j$ in $L = (\tilde{\ell}_1 \cdots \tilde{\ell}_{K_L})$ which are respectively *unaffected* and *affected* by label-compression.

In order to obtain a certain degree of quality in the approximation process, we have to care a balance between feature and label-compressions. Following [1], we take into account *Frobenius Norm* of the original matrices X_1 and X_2 and try to solve the following optimization (minimization) problem:

$$\min \frac{\left\| X_1 - (F|L) \begin{pmatrix} V_F \\ V_L \end{pmatrix} \right\|_F^2}{\|X_1\|_F^2} + \frac{\|X_2 - LW_L\|_F^2}{\|X_2\|_F^2},$$

where $\|X_1\|_F$ and $\|X_2\|_F$ can be treated as constants.

Based on a similar discussion to the standard formulation of NMF [2], we can obtain a set of multiplicative update rules for the optimization problem [1]. With element-wise expressions and $\lambda = \|X_1\|_F^2 / \|X_2\|_F^2$, we have

$$(L)_{ij} \leftarrow (L)_{ij} \times (S)_{ij}, \quad (1)$$

where $(S)_{ij}$ is given by

$$\frac{1}{(S)_{ij}} = \frac{(LV_LV_L^T + FV_FV_F^T)_{ij}}{(X_1V_L^T + \lambda X_2W_L^T)_{ij}} + \lambda \frac{(LW_LW_L^T)_{ij}}{(X_1V_L^T + \lambda X_2W_L^T)_{ij}} \quad (2)$$

for L defining the shared subspace. And, for $V = \begin{pmatrix} V_F \\ V_L \end{pmatrix}$, W_L and F , we have

$$(V)_{ij} \leftarrow (V)_{ij} \frac{((F|L)^T X_1)_{ij}}{((F|L)^T (F|L) V)_{ij}}, \quad (3)$$

$$(W_L)_{ij} \leftarrow (W_L)_{ij} \frac{(L^T X_2)_{ij}}{(L^T L W_L)_{ij}} \quad \text{and} \quad (4)$$

$$(F)_{ij} \leftarrow (F)_{ij} \frac{(X_1 V_F^T)_{ij}}{(LV_LV_F^T + FV_FV_F^T)_{ij}}. \quad (5)$$

2.2 Predicting Unknown Labels of Query

Based on the matrix factorization discussed above, we can predict a multi-label of a given query. We assume our query q with unknown-label is given as just an N -dimensional (feature) vector, that is, $q = (q_i)_{1 \leq i \leq N}$. A prediction about labels of q can be performed by computing a coefficient vector for the basis vectors $\tilde{\ell}_j$ reflecting label-information of objects in the factorization process. More precisely speaking, the query can be represented as

$$q = \sum_{i=1}^N q_i f_i, \quad \text{where } f_i \approx \sum_{j=1}^{K_F} v_{ji}^F \tilde{f}_j + \sum_{j=1}^{K_L} v_{ji}^L \tilde{\ell}_j.$$

Then we have

$$q \approx \sum_{j=1}^{K_F} \left(\sum_{i=1}^N q_i v_{ji}^F \right) \tilde{f}_j + \sum_{j=1}^{K_L} \left(\sum_{i=1}^N q_i v_{ji}^L \right) \tilde{\ell}_j.$$

Thus, the (K_L) -dimensional coefficient vector for $\tilde{\ell}_j$ can be given by $V_L q$.

After the approximation, each object x is represented by its corresponding row-vector v_x^T in the compressed matrix $L = (\tilde{\ell}_1 \cdots \tilde{\ell}_{K_L})$ reflecting their original label-information. Therefore, a distance between v_x and $V_L q$ provides us a hint about which labels the query would have. If the vectors are close enough, q seems to have labels similar to those of x . In other words, we can evaluate a *farness/closeness* between labels of x and q by defining an adequate distance function for those vectors. In the following discussion, we assume a distance function, $dist_L$, based on *cosine similarity* between vectors. That is, for an object x and a query q , it is defined as $dist_L(x, q) = 1 - \frac{v_x^T V_L q}{\|v_x\| \|V_L q\|}$.

2.3 Evaluating Similarity among Features of Objects

As is similar to the case of labels, we can evaluate similarity among features of objects based on the compressed matrix ($F|L$). However, since the matrix is affected by not only the feature compression but also the label compression, it would not be adequate for evaluating only similarity of features. For our evaluation of feature similarity, therefore, we try to approximate the original matrix X_1 into an $M \times K_T$ matrix H_{X_1} with the standard NMF such that $X_1 \approx H_{X_1}W_F$, where each object x after the compression is given as its corresponding row-vector h_x^T in H_{X_1} . Therefore, we can evaluate similarity between features of x and q by computing a distance between h_x and W_Fq , denoted by $dist_F(x, q)$.

3 Extracting Formal Concepts with Unexpected Multi-Label Objects for Query

Towards a recommendation-oriented information retrieval, we present in this section our method for finding formal concepts whose extents include some unexpected objects for a given query. The reason why we try to detect formal concepts is that the extent of a concept can be explicitly interpreted by its intent. That is, the intent provides us a clear explanation why those objects are grouped together. By extracting various concepts, we can therefore obtain interesting object clusters from multiple viewpoints.

3.1 Unexpected Objects Based on Predicted Multi-Label of Query

We first present our notion of unexpectedness of objects for a given query. Especially, we propose here an objective definition for the notion.

As has been discussed, we can *implicitly* predict a multi-label of a given query q with unknown-label. More precisely, we can measure a farness/closeness between labels of an object x and the query. In addition, we can evaluate similarity of features between x and q . Suppose here that we find both x and q have similar or relevant features. In such a case, it would be plausible that we expectedly consider they also have similar/relevant multi-labels. However, if we observe their labels are far from each other, we seem to find some unexpectedness of x for q because they have distant multi-labels even though their features are similar.

With the distance functions, $dist_F$ in the feature-subspace and $dist_L$ in the label-subspace, we can formalize this kind of unexpectedness of objects for a given query q with unknown-label.

Definition 1. (Unexpected Object for Query)

For an object x , if x satisfies the following two constraints, then x is said to be *unexpected* for the query q :

Relevance/Similarity of Features : x is relevant/similar to q in the feature-subspace, that is, $dist_F(x, q) \leq \delta_F$, and

Farness of Multi-Labels : x and q are distant from each other in the label-subspace, that is, $dist_L(x, q) \geq \delta_L$,

where $\delta_F (> 0)$ and $\delta_L (> 0)$ are user-defined parameters. ■

3.2 Extracting Formal Concepts with Unexpected Objects

Constructing Formal Context for Query : Suppose the original object-feature matrix X_1 is approximated as $X_1 \approx H_{X_1}W_F$ with the standard NMF, where H_{X_1} is regarded as a compressed representation of X_1 . It is recalled that the i -th object x_i in X_1 is given as the i -th row-vector $v_i^T = (v_{i1} \cdots v_{iK_T})$ in H_{X_1} , where the j -th element v_{ij} is the value of the feature f_j for x_i .

In order to extract formal concepts with unexpected objects for a given query q , we first define a formal context \mathcal{C}_q consisting of the objects relevant to q . Formally speaking, with the parameter δ_F , the set of relevant objects is defined as $\mathcal{O}_q = \{x_i \mid x_i \in \mathcal{O}, \text{dist}_F(x_i, q) \leq \delta_F\}$. The set of features (attributes), \mathcal{F}_q , is simply defined as $\mathcal{F}_q = \{f_1, \dots, f_{K_T}\}$. Moreover, introducing a parameter θ as a threshold for feature values, we define a binary relation $\mathcal{R}_q \subseteq \mathcal{O}_q \times \mathcal{F}_q$ as

$$\mathcal{R}_q = \{(x_i, f_j) \mid x_i \in \mathcal{O}_q, v_i^T = (v_{i1} \dots v_{iK_T}) \text{ in } H_{X_1} \text{ and } v_{ij} \geq \theta\}.$$

Thus, our formal context is given by $\mathcal{C}_q = \langle \mathcal{O}_q, \mathcal{F}_q, \mathcal{R}_q \rangle$.

Evaluating Formal Concepts : It is obvious that for each formal concept in the context \mathcal{C}_q , its extent consists of only objects relevant to q . The extent, however, does not always have unexpected ones. For a purpose of recommendation, since it would be desirable to involve some unexpected objects, we need to evaluate formal concepts in \mathcal{C}_q taking the fairness of multi-labels into account.

Let us consider a concept in \mathcal{C}_q with its extent E . We evaluate the concept by the average distance between each object in E and the query in the label-subspace. That is, our evaluation function, $eval$, is defined as $eval(E) = \frac{\sum_{x \in E} \text{dist}_L(x, q)}{|E|}$.

Although our formal context consists of only objects relevant to the query, we could have many concepts in some cases. We, therefore, try to obtain concepts with the top- N largest evaluation values.

Interpreting Intents of Formal Concepts : Our formal context is constructed from the matrix H_{X_1} which is a compressed representation of the original object-feature matrix X_1 approximated as $X_1 \approx H_{X_1}W_F$ with the standard NMF. That is, the intent of a concept in the context is given as a set of *compressed* features interpretable in terms of original features. The relationship among compressed and original features is represented as the matrix W_F . Each compressed feature is given as a row-vector in W_F in which larger components can be regarded as relevant original features. When we, therefore, interpret the intent of a concept, it is preferable to show relevant original features as well as each compressed one in the intent. As a simple way of dealing with that, assuming a (small) integer k , we can present original features corresponding to the k largest components for each compressed feature.

3.3 Algorithm for Extracting Top- N Formal Concepts

We present our algorithm for extracting target formal concepts.

```

[Input]  $\mathcal{C}_q = (\mathcal{O}_q, \mathcal{F}_q, \mathcal{R}_q)$  : a formal context obtained for a query  $q$ 
         $N$  : a positive integer for top- $N$ 
[Output]  $\mathcal{FC}$  : the set of formal concepts with the top- $N$  largest evaluation values


---


procedure MAIN( $\mathcal{C}_q, N$ ) :
   $\mathcal{FC} = \emptyset$  ;
   $\alpha = 0.0$  ; // the  $N$ -th (tentative) evaluation value of concepts in  $\mathcal{FC}$ 
  Fix a total order  $\prec$  on  $\mathcal{O}_q$  such that for any  $x_i, x_j \in \mathcal{O}_q$ ,  $x_i \prec x_j$  if  $dist_L(x_i, q) \leq dist_L(x_j, q)$  ;
  while  $\mathcal{O}_q \neq \emptyset$  do
    begin
       $x = head(\mathcal{O}_q)$  ;
       $\mathcal{O}_q = (\mathcal{O}_q \setminus \{x\})$  ; // removing  $x$  from  $\mathcal{O}_q$ 
      FCFIND( $\{x\}, \emptyset, \mathcal{O}_q$ ) ; //  $\mathcal{O}_q$  as candidate objects
    end
  return  $\mathcal{FC}$  ;


---


procedure FCFIND( $P, PrevExt, Cand$ ) :
   $FC = (Ext = P'', P')$  ; // computing FC
  if  $\exists x \in (Ext \setminus PrevExt)$  such that  $x \prec tail(P)$  then
    return; // found to be duplicate formal concept
  endif
  Update  $\mathcal{FC}$  adequately so that it keeps concepts with top- $N$  largest evaluation values found so far ;
   $\alpha =$  the  $N$ -th (tentative) evaluation value of concepts in  $\mathcal{FC}$ ;
  while  $Cand \neq \emptyset$  do
    begin
       $x = head(Cand)$  ;
       $Cand = (Cand \setminus \{x\})$  ; // removing  $x$  from  $Cand$ 
       $NewCand = (Cand \setminus Ext)$  ; // new candidate objects
      if  $NewCand = \emptyset$  then continue ;
      if  $eval(P \cup \{x\} \cup NewCand) < \alpha$  then continue ; // branch-and-bound pruning
      FCFIND( $P \cup \{x\}, Ext, NewCand$ ) ;
    end

```

Fig. 1. Algorithm for Finding Top- N Formal Concepts with Unexpected Objects

Let $\mathcal{C}_q = \langle \mathcal{O}_q, \mathcal{F}_q, \mathcal{R}_q \rangle$ be a formal context constructed for a given query q . As the basic strategy, generating extents of concepts, we explore object sets along the *set enumeration tree*, rooted by \emptyset , based on a total ordering \prec on \mathcal{O}_q . More concretely speaking, we expand a set of objects P into $Px = P \cup \{x\}$ by adding an object x succeeding to $tail(P)$ and then compute $((Px)'', (Px)')$ to obtain a formal concept, where we refer to the last (tail) element of P as $tail(P)$. Such an object x to be added is called a *candidate* and is selected from $cand(P) = \{x \mid x \in (\mathcal{O}_q \setminus P''), tail(P) \prec x\}$. Initializing P with \emptyset , we recursively iterate this process in depth-first manner until no P can be expanded.

Our target concepts must have the top- N largest evaluation values. Let us assume the objects x_i in \mathcal{O}_q are sorted in *ascending order* of $dist_L(x_i, q)$. Based on the ordering, along our depth-first expansion process of concepts, the evaluation values of obtained concepts increase *monotonically*. It should be noted here that for a set of objects $P \subseteq \mathcal{O}_q$, the extent E of each concept obtained by expanding P is a subset of $P'' \cup cand(P)$. Due to the monotonicity of evaluation values, therefore, $eval(P'' \cup cand(P))$ gives an upper bound we can observe in our expansion process from P . This means that if we find $eval(P'' \cup cand(P))$ is less than the tentative N -th largest evaluation value of concepts found so far, there is no need to expand P because we never meet any target concept by the expansion. Thus, a branch-and-bound pruning is available in our search process.

A pseudo-code of the algorithm is presented in Figure 1. The head element of a set S is referred to as $head(S)$. Although we skip details due to space limitation, the code incorporates a mechanism for avoiding duplicate generations of the same concept, as **if** statement at the beginning of **procedure** FCFIND.

4 Experimental Result

In this section, we present our experimental result with our system.

We have prepared a movie dataset consisting of 17,000 movies with their plots and genres. Our dataset has been created from *CMU Movie Summary Corpus* [13]¹. After preprocessing, the plot of each movie is represented as a boolean vector of 6,251 feature terms with medium frequency. That is, our movie-plot matrix X_P has the dimension of $17,000 \times 6,251$. Moreover, each movie is assigned some of 364 genre-labels as its multi-label. Then, our movie-label matrix X_L is given as a boolean matrix with the dimension of $17,000 \times 364$. Applying Nonnegative Shared Subspace Method, we have compressed X_P into a $17,000 \times 500$ matrix $(F|L)$, where dimensions of F and L are $17,000 \times 450$ and $17,000 \times 50$, respectively, and L is also a compressed matrix of X_L .

In addition, as candidates of our queries, we have also prepared 783 movies with only their plots, hiding their genres. Thus, our system is given a 6,251-dimensional boolean vector as a query obtained from those candidates.

Example of Extracted Formal Concept for “*Shark Bait* (2006)”

For a query vector obtained from the plot of a candidate movie “*Shark Bait*”, we present here a formal concept actually detected by our system.

“*Shark Bait*” is a computer animated family movie released in 2006. The story is about Pi, the main fish character, his relatives and friends while fighting against a mean tiger shark terrorizing Pi’s reef community.

For the query vector (plot), an example of formal concept our system detected is shown in Figure 2.

Similarity of Movie Plots : The extent of the concept consists of 5 movies all of which are concerned with marine animals. For example, “*Jaws*” is a very famous movie about a great white shark wrecking havoc in a beach resort. “*Open Water*” is a suspense movie based on a real story about a couple of scuba divers left behind in the shark-infested sea due to an accident. Moreover, “*Free Willy*”, a family-oriented adventure movie, is about the friendship between a street child and a killer whale in a local amusement park. As the intent shows, all of them are commonly associated with 5 features compressed with NMF each of which is related to several relevant feature terms used in the original plots. Actually, we can observe a certain degree of similarity among the plots of the movies in the extent. It is, furthermore, easy to see that they are also similar to the query plot.

¹ <http://www.cs.cmu.edu/ark/personas/>

Extent (movie : genres as multi-labels)	
<i>Jaws</i> (1975)	: Thriller
<i>Open Water</i> (2003)	: Thriller, Natural Horror Films, Horror, Indie, Drama, Suspense, Disaster
<i>Jaws 3-D</i> (1983)	: Thriller, Natural Horror Films, Horror, Action/Adventure, Mystery, Suspense, Action
<i>Free Willy</i> (1993)	: Adventure, Children's/Family, Animal Picture, Drama, Family Film, Family-Oriented Adventure
<i>Help! I'm a Fish</i> (2000)	: Science Fiction, Adventure, Animation, Children's, Fantasy, Drama, Family Film
Intent (compressed feature : related original feature (words))	
<i>F1</i>	: tank, fuel, leak, gasoline, oxygen
<i>F2</i>	: surface, beneath, descend, underwater, flood
<i>F3</i>	: ocean, Pacific, liner, Ocean, drift
<i>F4</i>	: swim, swimming, underwater, fish, dive
<i>F5</i>	: owe, loan, borrow, shark, fund

Fig. 2. Example of Extracted Formal Concept

Farness of Multi-Labels : As is shown in the figure, the movies in the extent have various multi-labels. More precisely speaking, the movies are listed in descending order of distance between the (implicitly) predicted multi-label of the query and that of each movie. That is, upper movies are expected to have multi-labels further from that of the query as unexpected ones. On the other hand, multi-labels of lower movies would be similar to that of the query. According to our problem setting, although the correct multi-label of the query was intentionally hidden, its actual genre labels are “Family Film” and “Animation”. It is easy to see that the lowest movie, “*Help! I’m a Fish*”, is categorized into genres very similar to the actual genres of the query. In other words, the multi-label of the query can reasonably be predicted with the help of Nonnegative Shared Subspace Method. As the result, based on our evaluation function for formal concepts, we can find some movies in the extent with unexpected (further) genre labels, like “*Jaws*” and “*Open Water*”. Inspired by such a concept, users could newly try and enjoy some movies with those unexpected genres. Thus, our method has an ability to stimulate us to try unexperienced movies based on unexpectedness of multi-labels.

Quality of Multi-Label Prediction : Quality of multi-label prediction would be important in our method because our unexpectedness is based on the prediction. We here observe quality of prediction by comparing two object rankings, one is based on predicted multi-labels of the query and the other based on its correct multi-label (hidden in our retrieval process). More concretely speaking, for the former, each movie in the dataset is ranked in ascending order of similarity between its multi-label and the predicted label of the query. The obtained ranking is referred to as R_{pred} . Similarly, we also rank each movie in ascending order of similarity between its multi-label and the correct label of the query. We refer to the ranking as $R_{correct}$. Then we can compute the Spearman’s rank

correlation coefficient between R_{pred} and $R_{correct}$. If we can observe a certain degree of correlation, we can expect that our prediction would be reasonable.

For the above example, we actually have the coefficient of 0.35 showing a weak positive correlation. As has been mentioned previously, a prediction method has been discussed in [17], where prediction is performed based on a subspace defined as a real line. For comparison, the correlation coefficient between the ranking according to the previous method and $R_{correct}$ is 0.03 showing little correlation. Although the value of 0.35 seems to be a little bit small, it can be increased to 0.84 when we focus on Top-10% ranked (1,700) objects in the ranking. It is noted here that our main purpose of prediction is just to identify objects whose multi-labels are far from that of the query. In this sense, precise lower ranks in R_{pred} are not matter. Therefore, we consider that the prediction of multi-labels in our method can work well for our purpose.

5 Concluding Remark

We discussed our method of finding interesting formal concepts with unexpected objects for a given query with no label. We defined our unexpected object for the query as object which is similar to the query in the feature space, but is dissimilar in the label space. In order to predict a multi-label of the query, the original object-feature and object-label matrices are simultaneously factorized by means of Nonnegative Shared Subspace Method, where the obtained subspace associates the label-information with the feature-information of the original matrices. Our retrieval task was formalized as a problem of enumerating every formal concept with Top- N evaluation values w.r.t. unexpectedness.

At the moment, we still leave quantitative evaluation of our method. As the unexpectedness in [14] has been quantitatively evaluated for another movie dataset according to [15], we can attempt a similar comparison. In addition, it would be worth verifying actual usefulness of our system through user trials.

We also need to investigate a prediction method of multi-labels. In our current framework, although the basis of label space is assumed to be a part of the basis of feature space. In order to further improve quality of multi-label prediction, it would be required to carefully distinguish two classes of labels, ones which can be well explained in terms of features and the others. By a correlation analysis for labels and features, we can define an adequate regularization term in the objective function for our matrix factorization. This kind of analysis is also very important in the framework of *feature association* which identifies reasonable similarities among features in different data matrices [11].

Moreover, a recommendation system based on query-specific FCA-based bi-clustering has been proposed in [7]. We need to clarify relationship between the method and ours.

References

1. Gupta, S.K., Phung, D., Adams, B., Tran, T. and Venkatesh, S.: Nonnegative Shared Subspace Learning and Its Application to Social Media Retrieval, Proc.

- of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD'10, pp. 1169 – 1178 (2010).
2. Lee, D. D. and Seung, H. S.: Algorithms for Non-negative Matrix Factorization, Proc. of NIPS 2000, pp. 556 – 562 (2000).
 3. Kumar, C. A., Dias, S. M. and Vieira, N. J.: Knowledge Reduction in Formal Contexts Using Non-negative Matrix Factorization, Mathematics and Computers in Simulation, 109(C), pp. 46 – 63, Elsevier (2015).
 4. Kuznetsov, S. O. and Makhalova, T.: On Interestingness Measures of Formal Concepts, Information Sciences, 442(C), pp. 202 – 219, Elsevier (2018).
 5. Akhmaturov, M. and Ignatov, D. I.: Context-Aware Recommender System Based on Boolean Matrix Factorisation, Proc. of the 12th International Conference on Concept Lattices and Their Applications - CLA'15, pp. 99 – 110 (2015).
 6. Stumme, G., Taouil, R., Bastide, Y., Pasquier, N. and Lakhal, L.: Computing Iceberg Concept Lattices with TITANIC, Data & Knowledge Engineering, 42(2), pp. 189 – 222, Elsevier (2002).
 7. Alqadah, F., Reddy, C. K., Hu, J. and Alqadah, H. F.: Biclustering Neighborhood-Based Collaborative Filtering Method for Top- N Recommender Systems, Knowledge Information Systems, 44(2), pp. 475 – 491, Springer (2015).
 8. Sun, L., Ji, S. and Ye, J.: Hypergraph Spectral Learning for Multi-Label Classification, Proc. of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD'08, pp. 668 – 676 (2008).
 9. A. Takano, Y. Niwa, S. Nishioka, M. Iwayama, T. Hisamitsu, O. Imaichi, and H. Sakurai: Information Access Based on Associative Calculation, Proc. of SOFSEM 2000: Theory and Practice of Informatics, LNCS 1963, pp. 187 – 201 (2000).
 10. H. Zhai, M. Haraguchi, Y. Okubo, K. Hashimoto and S. Hirokawa: Shifting Concepts to Their Associative Concepts via Bridges, Proc. of the 9th International Conference on Machine Learning and Data Mining in Pattern Recognition, LNAI 7988, 586 – 600 (2013).
 11. H. Zhai and M. Haraguchi: A Linear Algebraic Inference for Feature Association, Proc. of the 12th Int'l Conf. on Knowledge, Information and Creativity Support Systems, IEEE Conference Publishing Services, pp. 102 – 107 (2017).
 12. Ganter, B. and Wille, R.: Formal Concept Analysis - Mathematical Foundations, 284 pages, Springer (1999).
 13. Bamman, D., O'Connor, B. and Smith, N. A.: Learning Latent Personas of Film Characters, Proc. of the 51st Annual Meeting of the Association for Computational Linguistics, pp. 352 – 361 (2013).
 14. Adamopoulos, P. and Tuzhilin, A.: On Unexpectedness in Recommender Systems: Or How to Better Expect the Unexpected, ACM Transactions on Intelligent Systems and Technology, 5(4), Article No. 54 (2015).
 15. Murakami, T., Mori, K. and Orihara, R.: Metrics for Evaluating the Serendipity of Recommendation Lists, New Frontiers in Artificial Intelligence, JSAI 2007 Conference and Workshops, Revised Selected Papers, LNAI-4914, pp. 40 – 46 (2008).
 16. Herlocker, J. L., Konstan, J. A., Terveen, L. G. and Riedl, J. T.: Evaluating Collaborative Filtering Recommender Systems, ACM Transactions on Information Systems, 22(1), pp. 5 – 53 (2004).
 17. Okubo, Y., Haraguchi, M. and Liu, H.: Finding Interesting Formal Concepts with Unexpected Objects with respect to Multi-Labels, Proc. of the 6th Asian Conference on Information Systems - ACIS 2017, pp. 98 – 105 (2017).