2C2-3

Laplacian-Constrained Tri-factorization for Feature Association Learning

Hongjie Zhai*1 Makoto Haraguchi*1

*1 Graduate School of Information Science and Technology, Hokkaido University.

In this paper, we study a problem of feature association learning. That is, given two object sets with their own feature sets, the learning task is to find associations between features, where part of associations is already presented. The proposed method is based on an idea: With the given associations, all the objects can be clustered into object clusters. By representing features with the object clusters, some new associations between remaining features can be determined. By repeating this process, we are able to get all the feature associations. We formalized this problem as a kind of Non-negative Matrix Tri-factorization (NMF). The method consists of two main parts: (1) it performs Non-negative Matrix Tri-factorization on two object sets. (2) during factorization, it uses a laplacian constraint to guarantee the associated features to have close vectors after they are projected into the embedded spaces.

1. Introduction

In this paper, we study a problem of feature association learning. That is, considering two object sets $O^1 = \{o_1^1, o_2^1, \ldots, o_n^1\}$ and $O^2 = \{o_1^2, o_2^2, \ldots, o_m^2\}$, where objects in O^1 are described by feature set $F^1 = \{f_1^1, f_2^1, \ldots, f_u^1\}$ and objects in O^2 are described by another feature set $F^2 = \{f_1^2, f_2^2, \ldots, f_v^2\}$, a partial associations between two feature sets are assumed already known. The associated features are written as tuples (f_i^1, f_j^2), where $f_i^1 \in F^1$ and $f_j^2 \in F^2$. The target of feature association learning problem is to mining the association between remaining features. Generally speaking, in the feature association learning problem, the features known to be associated are usually a small part of the total features. Thus, the problem can not be simply solved just by link prediction techniques [Lu 11], because many link prediction methods can only handle the data with small number of missing samples. Meanwhile, the feature association learning problem can also be transformed into a type of transfer learning problem. However, even though transfer learning may also be able to find the associations, they usually strongly rely on data domain. As the result, they may only work on specific types of data [Pan 10].

In this paper, we proposed a novel method for feature association. For mining the missing associations, our method tries to find new associations by the knowledge from known associations. Once new associations are detected, they can be added to known association set. Thus, we are able to use the new known association set to furtherly find new associations. By repeating this process, finally all the possible associated feature pairs can be detected. This paper is organized as follows: In Section 2., we explain the basic idea of our method. Section 3. will introduce the tri-factorization techniques to realize the idea. Finally section 4. and section 5. will discuss the result of the preliminary experiment and future works.

Contact: Hongjie Zhai, Graduate School of Information Science and Technology, Hokkaido University, N-14, W-9, Kita-ku, Sapporo, Hokkaido, zhaihj@kb.ist.hokudai.ac.jp

2. Basic Idea

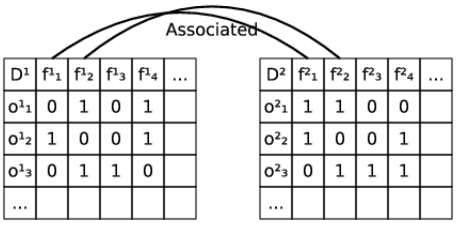
The basic idea of our method is illustrated in Figure 1. Here, we have two object-feature relation tables, where $f_1^1, f_2^1, f_3^1, \ldots$ are the features of objects $o_1^1, o_2^1, o_3^1, \ldots$. If a object contains the feature, the corresponding position will be 1 as shown in Figure 1a. Additionally, we also have two associated feature pairs: $f_1^1 - f_1^2$ and $f_2^1 - f_2^2$. Firstly, we merge the associated features into one. After that, by only focus on the merged ones, we can construct the vector representation of objects with the same dimensions. For example, in Figure 1b, f_1^1 and f_1^2 are merged as f_1 as well as that f_1^2 and f_2^2 are merged into f_2. With these constructed vectors, we perform clustering on objects. As the result, the objects in different sets may be clustered into one object cluster just like Figure 1b. Here, cluster c_1 contains object o_1^1, o_3^1 and o_3^2 while cluster c_2 contains object o_2^1 and o_2^2. Moreover, by representing features with object clusters, we can perform clustering on features to find new associations. This is illustrated in Figure 1c. It is easy to find that under the object cluster representation, f_4^1 and f_4^2 have the same vector. Thus, we found a new association $f_4^1 - f_4^2$. Once new associations are found, we merge into one and repeat the whole process until no new associations can be found.

Conclusively speaking, our method can be concluded as the following *two-phase* process:

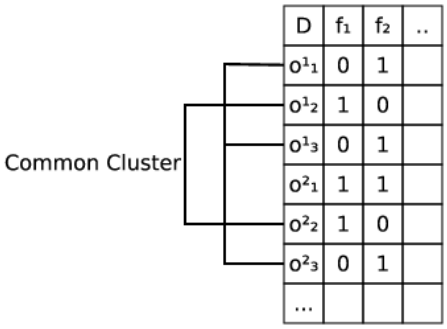
- Use associated features to build the common space, clustering objects in this common feature space.
- Use object clusters to build the common space, clustering features in this common object space.

3. Algorithm

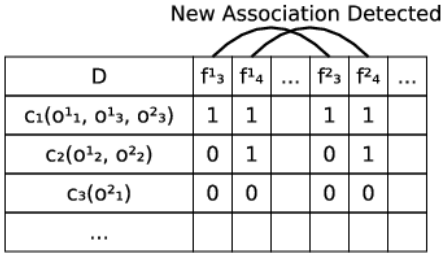
As previously described, our method is a two-phases algorithm by repeating performing clustering on objects and features. Technically speaking, when we focus on one object set as well as its features, by repeating the two-phases algorithm endlessly, we are actually performing a *co-clustering* algorithm on the object-feature set. Recently,



(a) Merge associated features



(b) Clustering in common feature space



(c) Mining new associations in common object cluster space

Figure 1: Illustration of Idea

non-negative matrix factorization (NMF) [Lee 01] methods have been proposed for co-clustering [Ding 06, Long 05]. Non-negative matrix factorization allows soft clustering, performs fast and its non-negative property has been shown to be a good constraints for natural information, especially text data. Thus, it becomes a popular choice for many applications. Generally speaking, the NMF decomposes the non-negative input data matrix D into two non-negative factors: W and H, where $D = WH$. However, NMF for co-clustering usually uses three non-negative factors: L, C and R, where $D = LCR$. Thus, the NMF for co-clustering is also called **Non-negative Matrix Tri-factorization**.

In our research, instead of the two-phases algorithm, we embed features into a common space by tri-factorization proposed by [Long 05]. In this common space, the associated features are guarantee to have the same vector representation. Thus, we formulated our idea into the algorithm 1.

Here, $D^1 \in \mathbb{R}+^{|O^1|\times|F^1|}$, $D^2 \in \mathbb{R}+^{|O^2|\times|F^2|}$ are the relation matrix, where $d_{ij} = 1$ if object o_i contains feature f_j. $W \in \mathbb{R}^{|F^1|+|F^2|\times|F^1|+|F^2|}$ is called feature relation matrix.

Algorithm 1: Tri-factorization for Association Learning

Data: Object-feature relation matrix: D^1, D^2, feature relation matrix: W, feature set F^1 and F^2, object set O^1, O^2, Dimension Parameter: N,M
Result: Associated feature pairs
Initialize random non-negative matrix L^1, C^1, R^1, L^2, C^2, R^2, where $C^{\{1,2\}} \in \mathbb{R}+^{N\times M}$;
$K = D - W$, where $d_{ii} = \sum_j w_{ij}$;
Solve the following tri-factorization problem:
$argmin_{L^1,C^1,R^1,L^2,C^2,R^2}|D^1 - L^1C^1R^1| + |D^2 - L^2C^2R^2| + \lambda(\frac{R^1}{R^2})^T K(\frac{R^1}{R^2})$;
Assign column vector in R^1 and R^2 to each feature in order;
for *each vector* v_j^1 *in* R^1 **do**
  Find the nearest vector v_j^2 in R^2;
  Print (F_i^1, F_j^2) as associated feature pair.
end

It is constructed from the following rules:

- if $i \leq |F^1| and j \leq |F^1|$, $w_{ij} = 0$
- if $i \geq |F^1| and j \geq |F^1|$, $w_{ij} = 0$
- if $i \leq |F^1| and j \geq |F^1|$, if f_i and f_j are known to be associated $w_{ij} = 1$, else $w_{ij} = 0$
- if $i \geq |F^1| and j \leq |F^1|$, $w_{ij} = w_{ji}$

By considering each feature as a vertex and connect the associated feature pairs, we can get a bi-graph G. It is easy to know that the matrix K is the laplacian matrix of graph G. According to [Cai 11], the laplacian constraint can make sure the associated features always have similar vector representation in the common space. It is not difficult to realize that if f_i^1 and f_j^2 have the same vector representation, the clusters they belongs to will be highly overlapped. As the result, we can know that the other words in this two clusters are associated. In addition, we also introduced λ as the parameter for adjusting the balance between co-clustering and the common vector representation constraint. In our algorithm, we followed [Chakraborty 15], which gave the detailed form of update rule as well as proved the convergence of laplacian-constrainted non-negative tri-factorization.

4. Experiment

To validate the ability of proposed method, we performed a preliminary experiment. We take the english/french news articles between 1996-08-20 and 1996-08-25 from Reuters Corpora [Lewis 04]. Detailedly speaking, the english news articles are selected from RCV1 (i.e. Reuters Corpus Volume 1) while the french news articles are selected from RCV2 (i.e. Reuters Corpus Volume 2). Articles are the objects and words are treated as features. We use all the $2,000$ articles in french and randomly sampled $2,000$ english articles from total about $20,000$ articles to balance

the size of dataset. After morphological analysis by treetagger [Schmid 13], we only keep nouns. As the result, the number of french words is $4,783$ and for english, it is $6,020$. By using english-french dictionary, we select 500 pairs of word with similar meaning as the associated features. By applying the algorithm introduced in section 3., we get the vector representation of words in a common space. The parameters used in experiments can be found in table 1. After this, to show the ability of finding associations, we clustered english/french into common clusters. If our algorithm works, we should find associated words in the common cluster, excluding the given associated words. Because of the space limitation, here we only give one example of the common clusters. As shown in table 2, for the ease of reading, the common cluster are splitted into english words and french words.

Parameters	value
λ	300
Dimension N	250
Dimension M	250

Table 1: Parameters

French	English
température malaria	*temperature* malaria
cardiologue accordéon	cardiologist rigor
peau respirateur	skin respirator
élu degustation	chill pneumonia
ombre bijou	saint FSA
bras jambe	bouquet heartbeat
occurrence idylle	pacemaker investor
...	...

Table 2: Content of common cluster

In this cluster, the french word température and english word temperature are known to be associated. We can see that for other words, many associated words are embedded into the same cluster, such as cardiologue vs cardiologist, peau vs skin, etc. However, we should also point out that because the example is quite a big cluster, many other words are not matched. The reason is that the news articles are written by totally different journalists in different country. There are many localized topics for different countries. For example, the cluster contains french word "degustation" (tasting in english), because the article also talked about the relation between temperature and chocolate while this topic never appeared in the english version. Even though those words are not the associated words in "common meanings", in the context of english and french papers, they appeared with temperature (température). From the viewpoint of co-occurrence, the common cluster is the result that we expected.

5. Conclusion and Future Works

This paper proposed a general method for feature association learning and give a tri-factorization formulation of the method. The tri-factorization formulation enjoys both the performance from NMF and the generalness of our idea. We also showed that the algorithm works as expected through a preliminary experiment. However, because of the deviation of data, we failed to get the "common sense" associations (i.e. the associated words in dictionary). In future, to solve this problem, (1) we would like to construct a more consist dataset. For example, we can select in a longer duration as well as limit the categories. (2) Because many words have different meanings under different contexts, we also would like to take semantic information into our consideration.

References

[Lewis 04] Lewis David D., et al. "Rcv1: A new benchmark collection for text categorization research." Journal of machine learning research 5.Apr (2004): 361-397.

[Cai 11] Cai Deng, et al. "Graph regularized nonnegative matrix factorization for data representation." IEEE Transactions on Pattern Analysis and Machine Intelligence 33.8 (2011): 1548-1560.

[Ding 06] Ding Chris, et al. "Orthogonal nonnegative matrix t-factorizations for clustering." Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2006.

[Long 05] Long Bo, Zhongfei Mark Zhang and Philip S. Yu. "Co-clustering by block value decomposition." Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. ACM, 2005.

[Lu 11] Lu Linyuan and Tao Zhou. "Link prediction in complex networks: A survey." Physica A: Statistical Mechanics and its Applications 390.6 (2011): 1150-1170.

[Pan 10] Pan Sinno Jialin and Qiang Yang. "A survey on transfer learning." IEEE Transactions on knowledge and data engineering 22.10 (2010): 1345-1359.

[Chakraborty 15] Chakraborty Yulong Pei1 Nilanjan and Katia Sycara. "Nonnegative matrix tri-factorization with graph regularization for community detection in social networks." (2015).

[Lee 01] Lee Daniel D. and H. Sebastian Seung. "Algorithms for non-negative matrix factorization." Advances in neural information processing systems. 2001.

[Schmid 13] Schmid, Helmut. "Probabilistic part-ospeech tagging using decision trees." New methods in language processing. Routledge, 2013.