The Graduate University for Advanced Studies,
SOKENDAI

DOCTORAL THESIS

Multiple Data Matching by Modeling Data Structures

Author:
Yuki SAITO

Supervisor:
Dr. Kenji FUKUMIZU

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

in the

School of Multidisciplinary Sciences
Department of Statistical Science

September 28, 2020

The Graduate University for Advanced Studies, SOKENDAI

Abstract

School of Multidisciplinary Sciences
Department of Statistical Science

Doctor of Philosophy

Multiple Data Matching by Modeling Data Structures

by Yuki SAITO

With the development of information technology in recent years, there is a need to apply machine learning to a wide range of industrial and academic fields, and emerging services and applications have started requiring matching multiple groups of data, namely, multiple data matching. Through the multiple data matching, we can investigate or infer the relationship between groups of data, such as common cluster structures or links. Furthermore, modeling the structure of the data is known as required to construct methods in some scenarios; however, several emerging use-cases are not well-studied, in which both modeling the structure of data and combining with new powerful functions (e.g., kernel functions and deep neural networks) is essential to match the multiple data.

This thesis considers matching up heterogeneous groups of objects by modeling the structures of data, involving various real-world applications. The problem scenarios divide into two cases: building methods to match (i) clusters, where the cluster structure commonly lies in heterogeneous domains, and (ii) two heterogeneous sets, where correct pairs are given as supervised information. This thesis focuses on extending the problems of multiple data matching onto the two different directions above.

(i) In the first case, we study a so-called supervised clustering to match common clusters that exist across two different domains, using given cluster assignments on one side of the domains as supervised information. The proposed method maximizes the similarity between the cluster structures within two domains in which kernel mean embeddings represent each cluster as probability distribution uniquely and nonparametrically. In the experiments, we use the datasets from meteoritics and planetary science and investigate taxonomical matching between the meteorites and asteroids. Here, the datasets consist of reflectance spectra of asteroids and meteorites, and also major chemical compositions of meteorites, where cluster assignments of the meteorites are known as the abovementioned supervised information, and the problem is to solve supervised clustering on the asteroidal domain. By comparing the clustering accuracy of the asteroid between with and without the guidance of the meteorite, we observe that the guidance of meteorite taxonomy improves the accuracy, either with the reflectance spectra or major chemical compositions of meteorites. This fact serves as a piece of evidence that there is a common taxonomic structure and links between meteorites and asteroids, implying a long-standing hypothesis of the taxonomy matching.

(ii) Second, we investigate heterogeneous set-to-set matching problem building novel deep neural networks. In this case, we are only given paired group data for training and inference, and the learned neural network models must classify whether an unknown paired data matches or not in the inference. The difficulties of the heterogeneous set-to-set matching are to extract features to match a correct pair of different sets and also preserve two types of exchangeability required for set-to-set matching: the pair of sets, as well as the items in each set, should be exchangeable. In this study, we propose a deep learning architecture for heterogeneous set-to-set matching to address the abovementioned difficulties. The proposed framework includes two novel modules: (1) a cross-set feature transformation (CSeFT) module and (2) cross-similarity (CS) function. The former provides the exchangeable set feature based on the interactions between two sets in intermediate layers, and the latter performs the exchangeable set matching by calculating the cross-feature similarity of items between two sets. Furthermore, we propose a novel loss function, *K*-pair-set loss, to train our model effectively. The effectiveness of our approach is demonstrated in two real-world applications. First, we consider fashion set recommendations via matching fashion outfits, where provided examples of the outfits are used as correct combinations of items. Since the paired sets include images of different fashion items, we regard this case as heterogeneous set matching. Next, we evaluate our methods through group re-identification experiments using two datasets, a new extension of the Market-1501 dataset (Market-1501 Group) and the Road Group dataset. Considering group membership change, we regard group re-identification as a heterogeneous set matching problem. In the experiment, we further introduce the novel data augmentation method that augments paired data (set-data augmentation). In these experiments, we show that the proposed method provides significant improvements and results compared with the state-of-the-art methods, thereby validating our architecture for the heterogeneous set matching problem.

Acknowledgements

The work presented in this thesis could not have been possible without the help of my advisors. I would particularly like to thank my advisor Kenji Fukumizu for his support and direction; he gives me thorough supports to grow as an independent researcher and guidance on how to tackle research problems logically, also in the way of mathematical and statistical thinking. I had the great pleasure of being under his direction. I also thank the doctoral dissertation committee, Hideitsu Hino, Daichi Mochihashi, and Tomoharu Iwata, for their insightful suggestions.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the development of information technology in recent years, there is a need to apply machine learning to a wide range of industrial and academic fields, and emerging services and applications have started requiring group-based data matching, namely multiple data matching. Multiple data matching has been studied to investigate relationships between groups of data, and we consider it an extended variant of an ordinary data matching that serves to match two data.

Matching up two data is one of the fundamental elements of many machine learning applications and has been investigated in various areas. Image matching (Thirion, 1998; Van den Elsen, Pol, and Viergever, 1993; Gruen, 1985) is a typical example of the leading research territories in computer vision, including person re-identification (Zheng et al., 2015), face identification (Chopra, Hadsell, and LeCun, 2005), and object retrieval (Li, Larson, and Hanjalic, 2015). These subjects have been well-studied and implemented as core functions in real-world applications, such as surveillance and robot navigation systems.

Multiple data matching includes but not limited to cluster matching for network data using infinite relational models (Iwata and Ishiguro, 2017), document matching across different languages by topic models (Iwata, Hirao, and Ueda, 2017), and latent factor models for a cross-domain recommendation (Gao et al., 2013). Furthermore, several researchers have studied matching in a view of dependence maximization based on the Hilbert-Schmidt independence criterion (Blaschko and Gretton, 2008), mutual information criterion (Faivishevsky and Goldberger, 2010; Kimura and Sugiyama, 2011), kernel canonical correlation (Blaschko and Lampert, 2008), kernel maximum mean discrepancy (Li et al., 2017), and discriminative approach using deep neural networks (Huang et al., 2019a). We can see several feature-based

image matching methodologies that match groups of local features on keypoints under geometric constraints as multiple data matching (Li, Larson, and Hanjalic, 2015; Tolias, Avrithis, and Jégou, 2016). The abovementioned methods are studied to discover a common latent structure forming clusters or groups that exist across multiple domains in an unsupervised manner.

In a supervised manner, several approaches for multiple data matching have been proposed. Given paired data, stochastic models have been proposed to investigate latent links between multiple entities (Chang and Blei, 2009; Airoldi et al., 2008; Nallapati et al., 2008). In clustering-based approaches, considering auxiliary information in a related domain as one of supervised information, several methods have been proposed to discover common clusters (Wang, Domeniconi, and Hu, 2008; Dai et al., 2007a), which are also related to works of literature of transfer learning (Raina, Ng, and Koller, 2006; Dai et al., 2007b; Yang et al., 2009) and self-taught clustering (Dai et al., 2008). Furthermore, so-called multitask learning (Caruana, 1997; Argyriou et al., 2008) and semi-supervised multitask learning (Ando and Zhang, 2005) have been developed, assuming that common labels or feature spaces exist across different domains.

Based on the property of matching use-case in applications, the multiple data matching scenarios can be grouped into two classes: homogeneous matching and heterogeneous matching. In the former, the groups comprising the same instances, such as the images of the face of the same person, are to be matched. In the example of face matching, except for variations such as differences in illumination or pose in the images, both groups contain similar instances. Homogeneous matching has been investigated in several studies (Gao et al., 2018; Lu et al., 2015; Feng, Karaman, and Chang, 2017; Liu, Yan, and Ouyang, 2017; Liu et al., 2019b; Xie, Shen, and Zisserman, 2018; Shakhnarovich, Fisher, and Darrell, 2002; Arandjelovic et al., 2005; Cevikalp and Triggs, 2010; Yamaguchi, Fukui, and Maeda, 1998; Liu et al., 2019a). In heterogeneous matching, the instances within paired groups can be considerably different. For example, group re-identification in surveillance systems (Lisanti et al., 2017; Xiao et al., 2018; Lin et al., 2019), which has recently started implementing a function to track known groups of suspicious persons or criminals, is a task that

can be simplified as heterogeneous matching, taking into account group membership change, which may change membership in paired groups under noisy condition. In addition, topic models for entity matching (Yang et al., 2015), graph matching for malware detection (Wang et al., 2019), and multiple-instance learning for anomaly detection (Sultani, Chen, and Shah, 2018) have been studied for heterogeneous matching. We consider that heterogeneous matching is a more difficult task and requires a strong framework to match different data.

Furthermore, modeling the structure of the data is required to construct methods for multiple data matching in several use-cases, which have begun implementing functions towards emerging applications or services. For example, various methods have been proposed for representing a set of data and used to match the sets. A vector with a fixed length, such as a histogram of local features, has been introduced to analyze documents (Le and Mikolov, 2014) and images (Yang et al., 2007). Furthermore, different studies have suggested modeling a set as a hull (Cevikalp and Triggs, 2010; Hu, Mian, and Owens, 2011; Yang et al., 2013; Zhu et al., 2013), hyperplanes (Vincent and Bengio, 2002; Gionis, Indyk, Motwani, et al., 1999), linear subspace (Yamaguchi, Fukui, and Maeda, 1998; Kim, Kittler, and Cipolla, 2007; Wang et al., 2008; Hamm and Lee, 2008), convex cone (Sogi, Nakayama, and Fukui, 2018), exemplars (Hadid and Pietikainen, 2004), covariance matrix (Wang et al., 2012; Cai, Takala, and Pietikainen, 2010), Gaussian model (Shakhnarovich, Fisher, and Darrell, 2002; Arandjelovic et al., 2005), and kernel mean (Muandet et al., 2017), among others. In majority of the methods described above, specific measurements were needed, to measure the similarity/distance between the set models; for instance, (Peng, Zhang, and Li, 2016) requires optimizing two convex hulls on a set-to-set distance, using the Lagrangian multiplier method. Furthermore, various methods have been proposed for building a model to match graph data (Bai et al., 2019; Li et al., 2019; Guo et al., 2018), entities (Mudgal et al., 2018), and sequences (Si et al., 2018).

1.1 Our Goals

In various applications, I am interested in matching multiple data via modeling the structure of the data, towards matching up heterogeneous groups of objects. Although multiple data matching has been studied in a broad spectrum, as described above, several emerging use-cases are not well-studied, in which both modeling the

structure of data and combining with new powerful functions (e.g., kernel functions and deep neural networks) is essential to match the multiple data. This motivation leads to new research questions: with recent developments of (i) kernel functions, which uniquely and nonparametrically represent distributions, or (ii) deep neural networks, which provide powerful feature learning architecture,

- How can we extend matching problems via modeling the structure of data using the powerful functions?
- What are the new applications?

The goal of the work presented in this thesis is to extend the problems of matching multiple data by modeling the structure of data, involving various real-world applications. This thesis focuses on extending the problems onto the following two different directions:

1. **Modeling a cluster matching method that finds matched clusters, which commonly lie in heterogeneous groups of data:** the first goal of this thesis is to develop kernel mean embeddings to match clusters across two different domains. This problem scenario is based on assumptions that the two clusters consist of the same cluster structures, and we can match clusters based on calculating the similarity of distributions. Furthermore, in this scenario, we consider that cluster assignments in one side of domains are fully given as supervised information, and the remaining dataset does not contain any labels for the classification.
2. **Modeling a set-input function to match heterogeneous sets of data:** we study a problem scenario based on the situation for matching heterogeneous groups of data, where each group is defined as a set. In this case, we consider a deep learning-based method that learns models to discover matching between two sets via feature spaces. We investigate an application of this setting in a set-to-set matching task, requiring two types of exchangeability: the pair of sets, as well as the items in each set, should be exchangeable.

1.2 Thesis Overview

This thesis contains my works on addressing the abovementioned issues, which consists of two parts. The main parts of this thesis and the contributions are as follows:

1. In Chapter 2, we propose a novel clustering and cluster matching method using kernel mean embeddings and supervised information in one-side domain. In the experiments, we develop an application towards matching taxonomic structure of meteorites and asteroids and investigate a well-known and long-standing hypothesis that the links between meteorites and asteroids exist, by comparing the results of unsupervised and supervised clustering methods.

2. In Chapter 3, we develop a novel neural network architecture and effective learning techniques for heterogeneous set-to-set matching. We evaluate the methods through experiments based on two industrial applications: fashion set recommendation and group re-identification. In these experiments, we show that the proposed method provides significant improvements and results compared with the state-of-the-art methods, thereby validating our architecture for the heterogeneous set matching problem.

Furthermore, in Chapter 4, we summarize this thesis, and discuss the open problems for future research.

Chapter 2

Cluster Matching via Kernel Mean Embeddings and Its Application to Taxonomy Matching on Asteroid and Meteorite Domains

2.1 Motivation

Linking meteorites to asteroidal bodies is an important subject toward a better understanding of the origin, structure and history of the Solar System (Cloutis, Binzel, and Gaffey, 2014). It allows us to infer the property of asteroids from meteorite information without taking the sample from the asteroid directly. For decades, researchers have been discussing the relations of taxonomy classifications observed in the reflectance spectra of asteroids and meteorites (Britt et al., 1992; Gaffey, Burbine, and Binzel, 1993; Pieters et al., 2005). This is still challenging, however, since the clear matching of absorbing features is not possible for various reasons including space weathering (Noguchi et al., 2011) and diversity of terrains in asteroids. The largest difference in the spectrum between asteroids and meteorites are a scale of analyses. Asteroidal spectra are collected with the ground and/or space telescope that is observing entire bodies and incident angles are not perfectly controlled. On the other hand, meteorites are measured under controlled conditions, however, petrographical and mineralogical heterogeneity are existing due to measurements are conducted with a limited mass of meteorites. That indicates spectrum signatures of asteroids and meteorites should not match perfectly. Therefore, to compare asteroid and meteorites with spectrum signature, new taxonomic methods are required.

One of the earliest links between asteroids and meteorites is seen for 4 Vesta (V-type asteroids) and Howardite-Eucrite-Diogenite (HED) meteorites (McCord, Adams, and Johnson, 1970). The in situ analysis by the Dawn mission confirmed this link and provided further insights on the relations (Buratti et al., 2013). Another famous link is seen between S-type asteroids and ordinary chondrite meteorites, which had been predicted for long (Chapman, 1996; Hiroi et al., 1993), and was confirmed by the analysis of the sample taken from Itokawa by the Hayabusa mission (Nakamura et al., 2011); however, the spectrum of Itokawa are different from existent ordinary chondrites, and the discrepancy has been thought to be caused by the space weathering (Noguchi et al., 2011). Note that Gaffey et al. (Gaffey, Burbine, and Binzel, 1993) and Hardersen et al. (Hardersen et al., 2006; Hardersen et al., 2011) have shown that there is likely to be a significant amount of diversity in the meteorite analogs within given asteroid taxonomic classes, and that not all S-type asteroids seem to be corresponding to ordinary chondrite meteorites. However, the population of S-type asteroids and ordinary chondrites are largest among asteroids and meteorites, respectively, therefore, most of the ordinary chondrites should be linked with S-type asteroids. We also consider that the inconsistency has not been found for all links between S-type asteroids and ordinary chondrite meteorites, and assume that such mismatching cases are not dominant for our datasets. Even if our datasets contain such mismatching data, our data-driven approaches can be performed robustly taking into account the statistical trends within the clusters, based on the statistical analysis for the well-known link between S-type asteroids and ordinary chondrite meteorites. A less certain link is known between C-type asteroids and carbonaceous chondrites (Busarev, 2012). Current asteroid missions, Hayabusa-2 (JAXA) is observing C-type asteroid Ryugu and spectral signatures are similar to Carbonaceous chondrites. Other more uncertain links have been discussed also (Cloutis, Binzel, and Gaffey, 2014). For investigating these links, there are some works that take data-driven approaches (Britt et al., 1992); they provided a "map" to overview reflectance spectral similarity between meteorites and asteroids by using principal component analysis (Jolliffe, 1986). Britt et al. (Britt et al., 1992) have provided a description that there are similarities between asteroid and meteorite spectra by observing the visualized space of the principal components.

This chapter aims at pushing forward this line of data-driven approaches, and develops a data analysis method for matching the taxonomies between meteorites

and asteroids. By using the matching taxonomy system, checking whether the taxonomy information of meteorite improves classification for asteroid data, we examine the existence of common structure over the two domains. More precisely, the method classifies asteroid data with guidance of the known taxa of meteorites (Weisberg, McCoy, and Krot, 2006), and matches the taxa between the two domains of asteroids and meteorites. For this clustering and matching procedure, reflectance spectral data of asteroids, recorded with the visible and near-infrared wavelength, from 0.45 to 2.45 µm, are clustered so that the similarity between the cluster structures of the asteroids and meteorites is maximized. To represent the cluster structures, a nonlinear method is used to quantify the mutual relations among clusters. For the teaching data to guide the clustering of asteroid data, the reflectance spectral data and major chemical composition data of meteorites are used. As described later, for matching the taxa, our statistical analysis does not rely on spectrum difference between asteroids and meteorites but similarities among clusters between the two domains, so that it does not require to compare spectrum signatures of asteroids and meteorites.

Analysis in this chapter is asymmetric for asteroids and meteorites, as shown in Figure 2.1: the meteorite data are used as guidance, and the asteroid data are clustered so that they have similar taxonomic structures to the meteorite taxonomy. In this sense, the analysis procedure is regarded as a supervised clustering. This supervised clustering approach is taken because the classification of the meteorite data is expected to be much more reliable; the laboratory measurements have higher accuracy, and they are labeled with help of other features from a different perspective in addition to chemical compositions, reflectance spectra, and petrological analysis.

To validate our approach, the resulting clusters of asteroid data are compared with the known labeling (Tholen, 1984) provided by experts. The same asteroid data are clustered by standard clustering methods without guidance, and the results with and without guidance are compared. Given that the expert labeling is correct, if the clustering with guidance shows better agreement than that without guidance, it will give a piece of evidence that a common taxonomy structure exists and that the asteroids and meteorites have similar cluster structures.

The analysis of this chapter is confined only to the C-, S-, and V-type for asteroids (spectral classification), while the carbonaceous chondrites, ordinary chondrites, and HED for meteorites (petrological classification). As already described, these classes

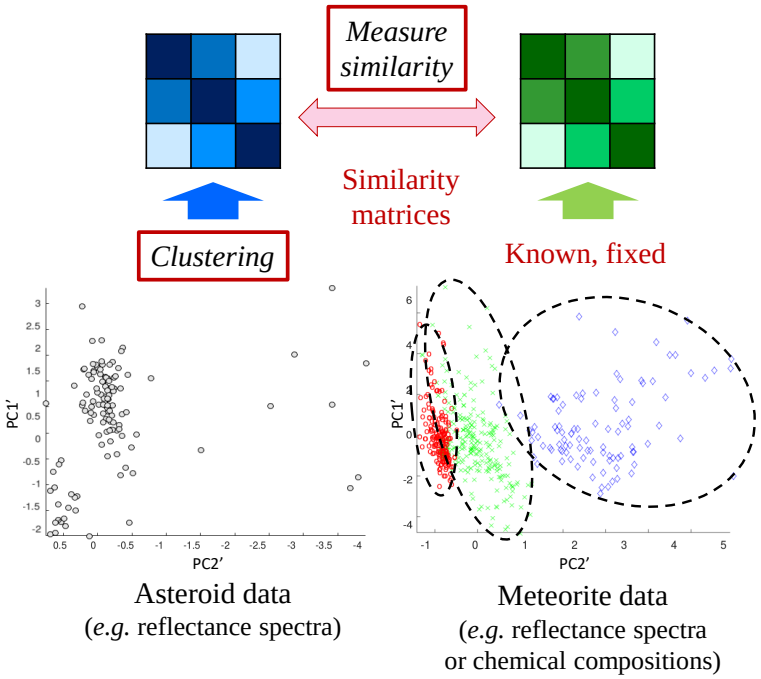


Figure 2.1. Analysis of taxonomy matching. First, unsupervised clustering finds cluster assignments of asteroids, independently for meteorites. Second, supervised clustering finds asteroid clusters with the expert's knowledge of meteorite to increase the similarity, using the unsupervised clustering results as initialized states. Third, compare the above two clustering accuracies; it will be confirmed that a common taxonomy structure exists if the supervised clustering accuracy outperforms the unsupervised clustering accuracy. Those three steps are the procedure of the taxonomy matching. The visualized spaces are individually computed by DeMeo's method with principal component analysis, and PC1' and PC2' are the first and second principal components, respectively.

are known to match each other more certainly than other types of asteroids and meteorites, and this choice is taken so that the analysis can be relatively easy for the first step and the obtained results can be confirmed by the knowledge. Furthermore, non-hierarchical clustering is considered here to focus on the simple cluster matching between meteorites and asteroids. The reflectance spectral data are used for asteroids, while for meteorites each of reflectance spectra and chemical compositions is served for teaching data. Note that, as explained in detail later, the nonlinear method for data analysis utilized in this chapter enables us to handle two data sets of different types: spectra and compositions.

2.2 Taxonomy Matching by Cluster Matching

Our method for supervised clustering consists of two steps. In the first step or initialization, we utilize a standard clustering method to find clusters by using only the data on domain X. In the second step, we use the domain Y data with supervised information to guide clustering procedures on the domain X data that referred to as the cluster matching. We also apply to the standard clustering in the first step as unsupervised clustering in contrast with supervised clustering, as explained below.

2.2.1 Unsupervised Clustering

For the unsupervised clustering, this chapter uses the spectral clustering algorithm (Ng, Jordan, and Weiss, 2002), a popular non-hierarchical clustering method as well as the K-means clustering (MacQueen, 1967). Spectral clustering, in general, takes a data-similarity-matrix as input. Each element of the matrix contains the similarity between two data indexed by the row and column.

The similarity between two data x and y is evaluated by Gaussian kernel function $\exp(-\frac{\|x-y\|^2}{2\sigma^2})$ with deviation parameter σ. This kernel extracts nonlinear similarity between x and y. This kernel function is composed of squared distance between x and y, a parameter $2\sigma^2$, and an exponential function exp. The parameter σ^2 can be interpreted as a variance parameter of the squared distance which is chosen by hand. It is used for data on domain X and Y with deviation parameter σ_X and σ_M, respectively. A different choice of these parameters may cause different clustering results. The results are in fact stable over a wide range of parameters as shown in Section 2.4. The normalized negative distance $-\frac{\|x-y\|^2}{2\sigma^2}$ is then fed into an exponential function, calculating the similarity between x and y. This means that if the distance between x and y is too far, then the similarity will be exponentially small. By definition, a range of similarity is between 0 and 1. For calculating the similarity using spectra data, note that the squared distance between the data of x and y is not only calculating the difference of its absorption bands of x and y but also the total difference in the spectrum.

The spectral clustering can be regarded as a relaxation of the graph cut algorithm for which the edge weights are given by the data-similarity matrix. In the case of the domain X data, the $N_X \times N_X$ data-similarity-matrix K_X contains similarity between each pair of the data on domain X, where N_X is the size of the dataset, calculated

by using kernel functions. For example, the (i, j)-component of K_X shows similarity between i- and j-th data x_i and x_j, calculated by $\exp(-\frac{\|x-y\|^2}{2\sigma^2})$. Among some variants of spectral clustering algorithms, in chapter, we use the one proposed by Ng et al. (Ng, Jordan, and Weiss, 2002), which uses the eigenvectors of the normalized graph Laplacian,

$$L_X := T_X^{-\frac{1}{2}} (T_X - K_X) T_X^{-\frac{1}{2}}, \tag{2.1}$$

where T_X is the diagonal matrix $T_{X,ii} = \sum_i K_{X,ii}$ containing the degree of the nodes in the graph. The algorithm further uses K-means clustering methods after projecting the data onto the eigenspaces corresponding to the least eigenvalues. For the details, see (Ng, Jordan, and Weiss, 2002). It is known that the results of the K-means clustering depend on initialization; K-means clustering starts clustering using initialized centers of clusters and converges to different results depending on the initial states. To avoid this issue, we take 50 initializations, by random K points from the data set, and take the best result in terms of the K-means objective function. We limit the iteration number by 100.

2.2.2 Supervised Clustering

As an overview, the proposed method of supervised clustering is also based on similarity. We can calculate the similarity of cluster structure between two domains following two steps described below; first, data-similarity-matrix is constructed for each of domain X and Y, and then two *cluster-similarity-matrices* are computed from the respective data-similarity-matrices to represent the cluster structure in each domain. Second, a similarity measure between these cluster-similarity-matrices is used for quantifying the similarity of taxonomies, serving as an objective function for clustering the domain X data and matching the clusters of the two domains.

For more details, we first explain the domain X. Suppose that data-similarity-matrix K_X is already calculated in the same way of spectral clustering explained above. The cluster-similarity matrix, representing the similarity between two clusters, is then calculated by

$$S_X = (W_X D_X)^T K_X W_X D_X, \tag{2.2}$$

$$D_{\mathrm{X}} = \begin{bmatrix} \frac{1}{N_{\mathrm{X}}^1} & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & & 0 \\ 0 & \dots & 0 & \frac{1}{N_{\mathrm{X}}^C} \end{bmatrix} \tag{2.3}$$

where W_{X} is $N_{\mathrm{X}} \times C$ cluster assignment matrix of dataset on domain X, $C = 3$ is common cluster size on the domain X and Y, N_{X}^i is the number of data in the i-th cluster C_{X}^i of the dataset on domain X, and D_{X} is a normalizing term using the number of data N_{X}^i. W_{X} indicates cluster assignments of the data on domain X: each row of W_{X} contains only one 1 element and the others 0. If (i, j)-component of the cluster assignment matrix is 1, then it indicates that i-th data belongs to j-th cluster. Each data is thus assigned to only one cluster. The (i, j)-component of the cluster-similarity-matrix S_{X} is then equal to

$$S_{\mathrm{X},ij} = \left(W_{\mathrm{X},i} D_{\mathrm{X},ii}\right)^{\mathrm{T}} K_{\mathrm{X}} W_{\mathrm{X},j} D_{\mathrm{X},jj} = \frac{1}{N_{\mathrm{X}}^i N_{\mathrm{X}}^j} \sum_{a \in C_{\mathrm{X}}^i, b \in C_{\mathrm{X}}^j} K_{\mathrm{X},ab}, \tag{2.4}$$

which represents the average total similarities between the data in the two clusters. The cluster-similarity-matrix S_{Y} for the dataset on domain Y is obtained in a similar way.

Note that different data-formats, dimensionalities, and data sizes can be handled by the proposed similarity representations. Using any quantitative datasets, we can calculate cluster-similarity-matrices, and then measure the similarity between them; the datasets on domain X and Y can have different data formats to match the clusters. As an example, for major chemical compositions of meteorites, we can calculate Gaussian kernel function using the amount of chemical compositions; by calculating the kernel function for all pairs of the composition data, we obtain data-similarity-matrix K_{Y}. In accordance with the guidance provided by experts, we can calculate cluster-similarity-matrices S_{Y} from K_{Y} by assigning the data to given clusters.

Since the objective is to cluster the domain X data so that they have a similar cluster structure as the dataset on domain Y, the two cluster-similarity-matrices S_{X} and S_{Y} should be made similar. Note that S_{X} is variable and S_{Y} is fixed in the supervised clustering discussed in this chapter; whereas S_{X} can be changed by switching cluster assignments for the domain X data, S_{Y} is unchanged once it is calculated by using given parameter σ_{M} for domain Y. We can see that our objective is optimizing

cluster assignments for the domain X data to maximize the similarity of the cluster structure.

A standard method for measuring the similarity of two matrices is the inner product of the matrices. We further "centerize" the matrices to compare two cluster structures around the origin, and apply normalization with respect to the effect of cluster assignment matrix W_X. Finally, the proposed objective for our supervised clustering is to maximize

$$\frac{\mathrm{Tr}\left(S_X H S_Y H\right)}{\left\| \left(W_X D_X H\right)^{\mathrm{T}} W_X D_X H \right\|_{\mathrm{F}}} = \frac{\mathrm{Tr}\left(H W_X^{\mathrm{T}} D_X K_X D_X W_X H S_Y\right)}{\left\| \left(W_X D_X H\right)^{\mathrm{T}} W_X D_X H \right\|_{\mathrm{F}}}, \tag{2.5}$$

where H is the centering matrix defined by $H_{ij} = \delta_{ij} - \frac{1}{C}$ (Gretton et al., 2005) for centering the matrix, and $\|M\|_{\mathrm{F}}$ denotes the standard Frobenius (Euclidean) norm of matrix M. Note that $HH = H$ and that $\mathrm{Tr}(ABC) = \mathrm{Tr}(BCA) = \mathrm{Tr}(CAB)$ with symmetric matrices A, B and C.

In this chapter, to optimize W_X for Eq. (2.5), a greedy search is used in a similar way to the K-means (MacQueen, 1967) and CLUHSIC (Song et al., 2007): one row of W_X (data point) is selected and its assignment is optimized at one iteration. Before the search, W_X is initialized by the spectral clustering, W_Y is obtained by the label given by experts, and the cluster order is also optimized so that Eq. (2.5) is maximized by checking all of the six cluster permutations of W_X. If the number of clusters is large, it is difficult to test all the permutations and a more efficient method such as kernelized sorting (Quadrianto, Song, and Smola, 2009) is needed. The optimization procedure stops if W_X no longer changes or it reaches the maximum number of iterations, or the cost function Eq. (2.5) increases less than $1e-6$ while the each iteration. By the initialization, the greedy search usually converges quickly, and the maximum number of iterations is set as 800.

2.3 Datasets

We use three datasets: reflectance spectra of asteroids, reflectance spectra of meteorites, and major chemical compositions of meteorites. In the experiments, we consider the asteroid and meteorite domains as domain X and Y, respectively. The datasets contain quantitative data representing features of meteorites or asteroids.

We do not directly utilize qualitative data such as petrological data which is described by researchers using text descriptions; however, the petrological data is indirectly fed into the supervised clustering, serving as the petrological taxa for meteorites.

2.3.1 Reflectance spectra dataset of asteroids

The original dataset of reflectance spectra of asteroids contains 365 data provided by SMASS Data Sets (Bus and Binzel, 2002; DeMeo et al., 2009; Rayner et al., 2003).

We use the reflectance spectra containing the wavelength of 0.45 to 2.45 µm from the original dataset and utilize the spectra in the range of from 0.45 to 2.45 µm. The wavelength is divided by 401 bins of equal size. The dataset also includes the labels for the taxonomical types including C-, S-, V-, and others. The data were smoothed and sampled by using spline models (Reinsch, 1967) in accordance with DeMeo's method (DeMeo et al., 2009) described below.

For preprocessing, we utilize DeMeo's method (DeMeo et al., 2009), which normalizes each reflectance spectra with the value at 0.55 µm, removes the slope of the reflectance spectra, and applies principal component analysis (Jolliffe, 1986). The principal component analysis is used to reduce the dimensionality of the data linearly, and to extract feature representation in the lower dimensional space. However, it may also reduce rich feature representation, which could be embedded in higher dimensional space; thus, we do not apply dimensionality reduction with the principal component analysis of DeMeo's method to preserve rich information of the data, and full dimensional information is utilized for the numerical experiments. Note that we apply the DeMeo's method not only to asteroid data but also to reflectance spectra of meteorites. After applying DeMeo's method, we extract data of only C-, S-, and V-type asteroids for our numerical experiments. They consist of 122 data, in which the numbers of C-, S-, and V-type asteroids are 19, 95, and 8, respectively. Note that applying DeMeo's method may reduce fidelity by the normalization; however, it also may decrease the difference in spectrum within true clusters, by reducing the scale and slope diversity, so that understanding and comparing clusters could be easier in feature spaces.

2.3.2 Reflectance spectra dataset of meteorites

The original dataset of reflectance spectra of meteorites (Pieters and Hiroi, 2004) contains 731 data, from which 221 carbonaceous chondrites, 245 ordinary chondrites, 108 HED meteorites, 574 in total, are used for the analysis. We only use the data which obviously belongs to those three classes and contains the wavelength of 0.45 to 2.45 µm, which is the same as the asteroid dataset. In this chapter, the carbonaceous chondrites are composed of C-ung, CM, CO, CR, CV, CH, and CI. The format of data and preprocessing are the same as the asteroid data.

2.3.3 Chemical compositions dataset of meteorites

We utilize 481 chemical compositions data of meteorites, consisting of 30 carbonaceous, 388 ordinary and 63 HED meteorites provided by (Yanai and Kojima, 1995). The dataset contains the amount of chemical compositions, and we utilize major 11 elements of Fe, Si, Al, Ca, Mg, Na, P, K, Mn, Ni, and Cr, and an oxide Na_2O. We do not apply any preprocessing method except the centering in the supervised clustering.

2.4 Experimental Results

2.4.1 Experiments on Synthetic Datasets

First, we experiment with synthetic data to investigate the ability of cluster matching by the proposed method. We randomly generate Gaussian distributions with dimension size 10 that commonly lie in two domains and then sample data from it. We use the gallery function implemented in MATLAB software to generate the covariance matrices of the Gaussian that are random 10-by-10 correlation matrices with random eigenvalues from a uniform distribution and determine the center from -20 to 20 randomly. The number of data sampled from each distribution in each domain is determined randomly from 10 to 20. Then, we decide domain X and Y and use domain Y as supervised information. Using the supervised one, we apply our method to execute clustering and cluster matching on the domain X. We create the synthetic dataset 20 times and calculate an average and standard deviation of cluster matching accuracy described as follows.

Cluster matching accuracy. Assuming that the matching hypothesis between the clusters of two different domains is correct, or that we know the correct correspondences between the clusters on the two domains, we can define the cluster matching accuracy. It is the average ratio of each cluster's correct assignments on domain X, where the corresponding cluster on domain Y gives the cluster labels. For example, in the asteroid and meteorite experiments, if 30 data in the first cluster of the asteroid domain is C-type, and carbonaceous chondrite of the meteorite is matched to that cluster, then these 30 data are regarded as "correct" matchings. The sum of such correct matchings divided by the total size is the cluster matching accuracy. By the definition, the cluster matching accuracy is equivalent to the degree of certainty for the matching hypothesis.

Figure 2.2 shows the experimental results, where C indicates the number of generated clusters, and the blue, red, and green bars designate different parameter settings of the sigma of Gauss kernel as 1, 5, and 9, respectively. We can see that our model matches clusters comparably accurately with $C = 3$. By increasing the number of clusters, the accuracy was significantly decreased; however, it shows that we can still obtain comparable accuracy by selecting the Gauss kernel's optimal parameter.

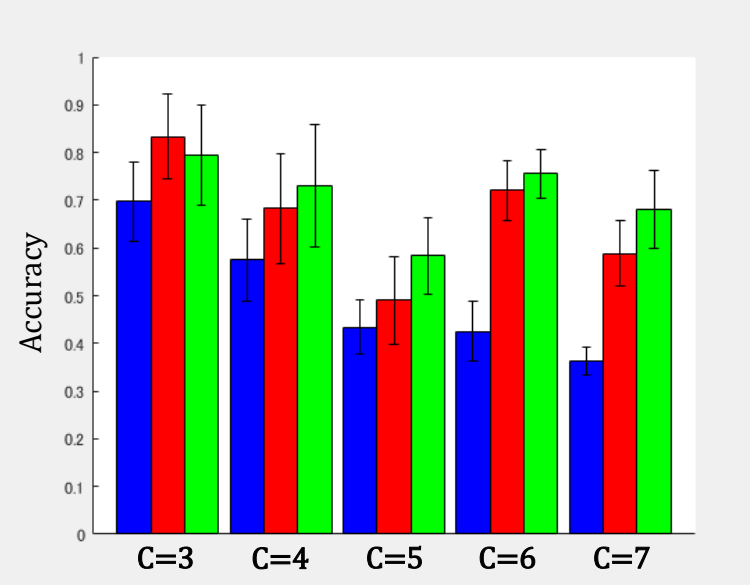


Figure 2.2. Supervised clustering results on the synthetic dataset. The bars show cluster matching accuracy described from 0 to 1.0.

2.4.2 Taxonomy Matching on Asteroid and Meteorite Domains

To evaluate the results of clustering and cluster matching, we use the label provided by experts for the asteroid data and calculate the cluster purity and cluster matching accuracy, described below. In this chapter, we validate the resulting clusters of asteroid data comparing with the well-known labeling (Tholen, 1984). We consider this asteroid letter-based taxonomic classifications are suitable for the first challenge, because of its simplicity, clarity, and importance. Note that the asteroidal label is not used in the clustering and cluster matching procedure, but used for evaluating the results.

Clustering parameters. The main parameters used for the clustering are the deviation parameter, σ_{A} for asteroids and σ_{M} for meteorites. We select each parameter in the range of from 0.5 to 7.0 in steps of 0.5 and perform the two clustering methods in sequel using the same parameters.

Cluster purity. The cluster purity of clustering is the portion of data with a dominant type in a cluster. It is also called global purity. To illustrate, suppose that asteroids of C-, S-, and V-type are dominant in the respective three clusters in an asteroid dataset of size 100, and that the sum of the numbers of the dominant cluster assignments is 70. Then the cluster purity is 70%.

Random resampling. Because clustering results may change under different settings, e.g., different parameters, or different datasets, statistically reliable analysis is needed. We show the results for randomly resampled datasets obtained by the following method, as well as for the whole dataset. We also try to obtain guidelines for selecting parameters showing its insensitivity. Note that the resampled datasets are shared between the unsupervised and supervised clustering with different deviation parameters to analyze under the same conditions (datasets).

By random resampling with a replacement for each ground-truth cluster, we independently generate 100 datasets. The sample size of each resampled dataset is 80% of the whole dataset; this is chosen to balance the randomness and a minimum number of data in a small class of V-type asteroids (it only has 8 data). The random resampling is performed after applying the data-preprocessing, e.g., DeMeo's method, to omit random effects on the preprocessing.

In the sequel, we first show the results for investigating the existence of common taxonomic structure over the meteorite and asteroid domains (datasets). As we described above, we examine the improvement of the cluster purity, applying the

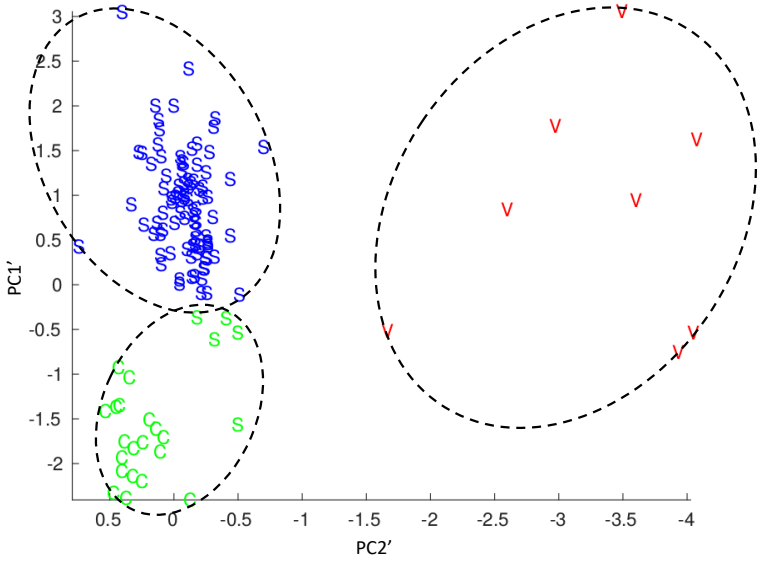


Figure 2.3. Unsupervised clustering results of asteroid reflectance spectra dataset without resampling. "C", "S", and "V" denote C-, S-, and V-type asteroids, respectively. The circles with dashed lines depict the clusters which are obtained by spectral clustering with $\sigma_A = 2.5$. Best view in color. The visualized space is computed by DeMeo's method with principal component analysis, and PC1' and PC2' are the first and second principal components, respectively.

supervised clustering to the results of the unsupervised clustering. Next, we investigate the links between the asteroid and meteorite, showing high cluster matching accuracy of the supervised clustering results. Before showing the results, we explain the pipeline for our taxonomy clustering and matching process in Figure 2.5.

2.4.3 Analysis 1: Common taxonomic structure

Here, we show the results of the clusterings for the whole dataset, without applying the resampling. The results of the unsupervised clustering are shown in Figure 2.3. Some of the S-type asteroids were miss-clustered with C-type, and the cluster purity was 94.1% with $\sigma_A = 2.5$. Figure 2.4 shows the results of the supervised clustering using the meteorite guidance of reflectance spectra ($\sigma_A = 2.5$, $\sigma_M = 5.0$) and the reflectance spectra data of the meteorite used for giving the guidance are plotted in Figure 2.6. The initial clusters for the supervised clustering were given by the unsupervised clustering shown in Figure 2.3. Most of the misclassifications are corrected appropriately, and the cluster purity are improved from 94.1% to 99.2%. The results of the supervised clustering with the meteorite guidance of chemical compositions ($\sigma_A = 2.5$, $\sigma_M = 5.0$) are shown in Figure 2.7. The cluster purity is improved from

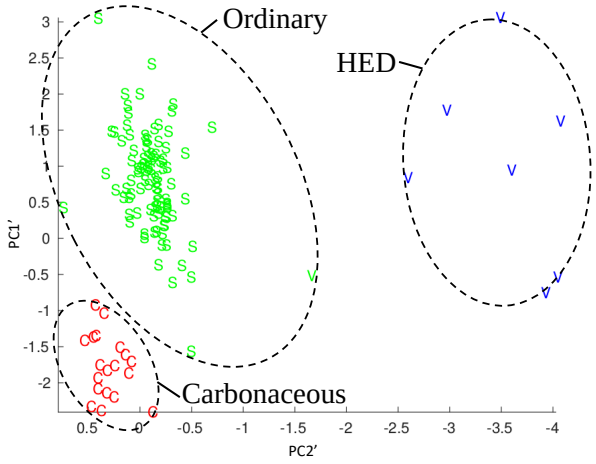


Figure 2.4. Supervised clustering results of asteroid reflectance spectra dataset using meteorite guidance of reflectance spectra dataset without resampling. The circles with dashed lines depict the clusters obtained by the supervised clustering with the parameters of $\sigma_{\mathrm{X}} = 2.5$, $\sigma_{\mathrm{M}} = 5.0$. Best view in color. The tags attached to the clusters show matched clusters of meteorites; "Carbonaceous", "Ordinary", and "HED" indicate carbonaceous chondrite, ordinary chondrite, and HED meteorite, respectively. The visualized space is computed by DeMeo's method with principal component analysis, and PC1' and PC2' are the first and second principal components, respectively.

94.1% to 98.4%. In other examples, we found that the results of supervised clustering are almost consistently accurate than unsupervised clustering results, i.e., the average cluster purity for different parameters is 97.4% by the supervised clustering, while by the unsupervised clustering it shown 89.3% purity on average with $0.5 \leq \sigma_{\mathrm{A}} \leq 7.0$ and $1.0 \leq \sigma_{\mathrm{M}} \leq 3.5$. In particular, the supervised clustering succeeds when results of the unsupervised clustering were failed; the supervised clustering recovers clustering purities from 84.4% to 93.4%, from 83.6% to 93.4%, and from 88.5% to 98.4% with $\sigma_{\mathrm{A}} = 0.5, 1.0$ and $3.5 \leq \sigma_{\mathrm{A}} \leq 7.0$, and with $1.0 \leq \sigma_{\mathrm{M}} \leq 3.5$, respectively. We consider the resulting differences are not induced by chance and the supervised information from meteorites improves the results of unsupervised clustering. We found that recovering purity from low-accurate results by the supervised clustering is commonly found in the following results. The meteorite guidance given by the chemical composition dataset is shown in Figure 2.8. Note that the proposed method of supervised clustering can incorporate a different data type for guidance. On average, using different parameters, we found that the cluster purity is 97.6% by supervised clustering whereas the unsupervised clustering gave only 89.3% with $0.5 \leq \sigma_{\mathrm{A}} \leq 7.0$ and $\sigma_{\mathrm{M}} = 4.0$.

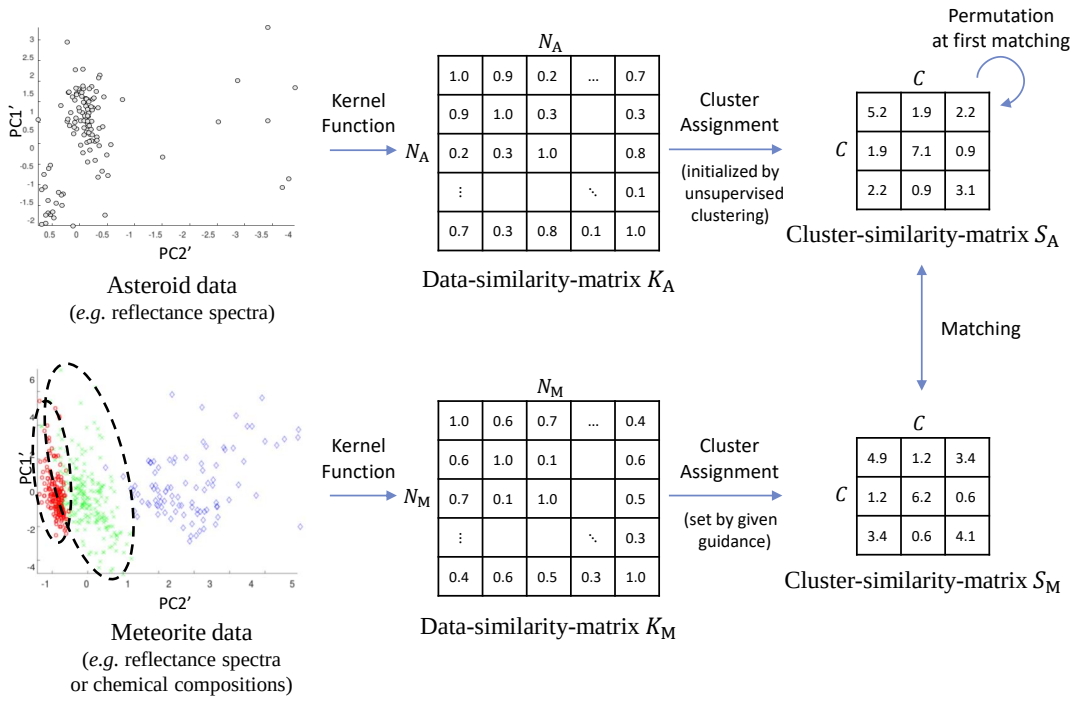


Figure 2.5. The pipeline of our clustering and matching process. Because the actual matrices are too big, we omit it and show examples.

In the cases of using the whole datasets above, we can see the improvements of cluster purity owing to the guidance of the meteorite, implying the existence of the common taxonomic structure.

Next, using the resampling, we show the average cluster purity as follows. The bars of "Unsupervised Clustering", "Supervised Clustering (spectra)", and "Supervised Clustering (compositions)" in Figure 2.9 show respective average cluster purities with different parameter settings of σ_A. The purities of the unsupervised clustering were at most 95.6% with $\sigma_A = 1.5$ or $\sigma_A = 2.0$, and the higher/lower parameters gave less accurate results, i.e., it was 84.5% with $\sigma_A = 0.5$. These results indicate that we need to select better parameters carefully to achieve better results of unsupervised clustering. The bars of "Supervised Clustering (spectra)" illustrate the average cluster purities of the supervised clustering results using the meteorite guidance of the reflectance spectra (σ_M is fixed as 2.0). The purities and its standard deviations are around 99% and 0.6, respectively, with $4.0 \leq \sigma_A \leq 7.0$. Most of the results outperform the unsupervised clustering results under a wide range of parameter settings, and the maximum purity is also significantly higher than the results of the unsupervised clustering. We also found that the supervised clustering always

improve the average cluster purity under the parameters at least with the range of $3.0 \leq \sigma_{\mathrm{A}} \leq 7.0$ and $0.5 \leq \sigma_{\mathrm{M}} \leq 7.0$. The bars of "Supervised Clustering (compositions)" in Figure 2.9 show the average cluster purities using meteorite guidance of chemical compositions (σ_{M} is fixed as 4.0); they show that the mean and the standard deviations are around 98.4% and 0.6, respectively, with $3.0 \leq \sigma_{\mathrm{A}}, \sigma_{\mathrm{M}} = 4.0$. The average purities also outperform the unsupervised clustering results except when the parameter setting was $\sigma_{\mathrm{A}} = 1.5$ in Figure 2.9; however, by setting $\sigma_{\mathrm{M}} = 4.3$ and $\sigma_{\mathrm{A}} = 1.5$, we found that the average purity is comparably 95.6% which is the same purity of the unsupervised clustering. We also found that the average purities are always improved under the parameters at least $2.5 \leq \sigma_{\mathrm{A}} \leq 7.0$ and $0.5 \leq \sigma_{\mathrm{M}} \leq 7.0$. In the results, with the random resampling, the supervised clustering improves the results of unsupervised clustering based on the meteorite guidance given by the reflectance spectra or the chemical compositions, under a wide range of the parameter settings. These results statistically support the existence of the common taxonomic structure between the asteroid and meteorite domains.

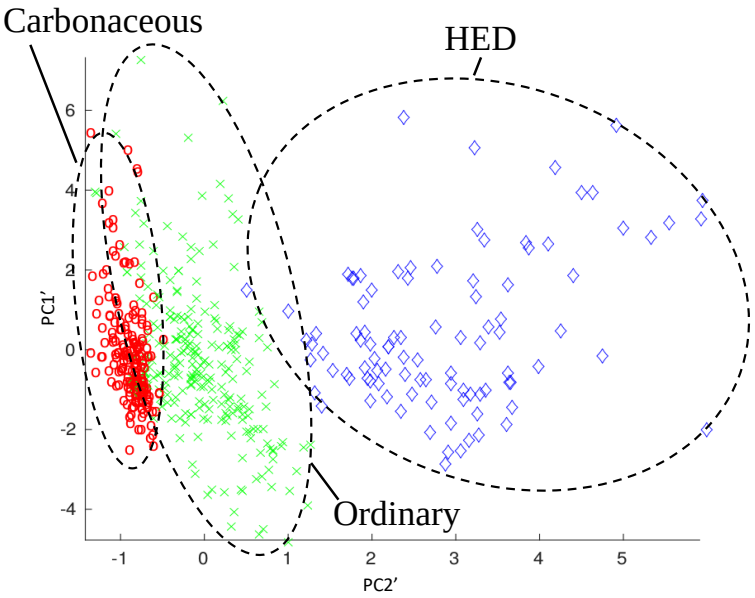


Figure 2.6. Meteorite reflectance spectra dataset without resampling. "o", "×", and "◇" denote carbonaceous chondrite, ordinary chondrite, and HED meteorite, respectively. The dashed circles indicate the clusters which are drawn by hand to separate the data belonged to different classes. Best view in color. The visualized space is computed by DeMeo's method with principal component analysis, and PC1' and PC2' are the first and second principal components, respectively.

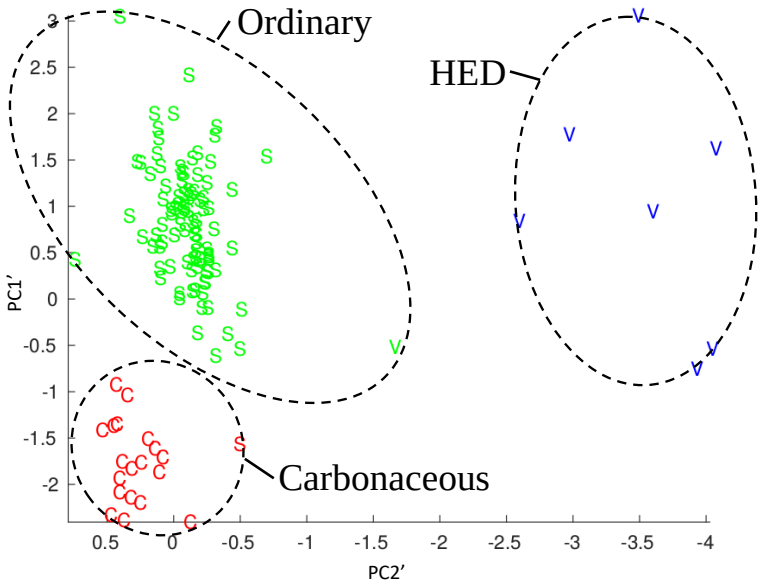


Figure 2.7. Supervised clustering results of asteroid reflectance spectra dataset using meteorite guidance of chemical compositions dataset without resampling. The circles with dashed lines depict the clusters obtained by the supervised clustering with the parameters of $\sigma_{\mathrm{A}} = 2.5$, $\sigma_{\mathrm{M}} = 5.0$. Best view in color. The tags attached to the clusters show matched clusters of meteorites; "Carbonaceous", "Ordinary", and "HED" indicate carbonaceous chondrite, ordinary chondrite, and HED meteorite, respectively. The visualized space is computed by DeMeo's method with principal component analysis, and PC1′ and PC2′ are the first and second principal components, respectively.

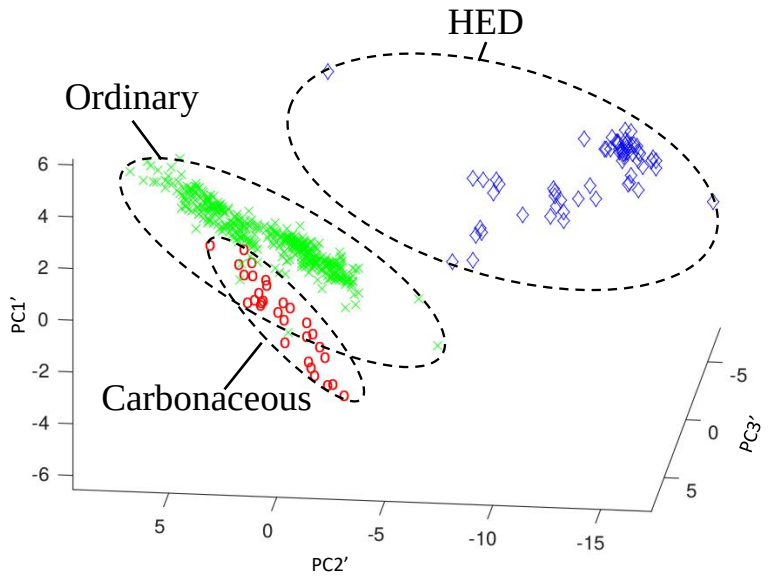


Figure 2.8. Meteorite chemical compositions dataset without resampling. "o", "×", and "◇" denote carbonaceous chondrite, ordinary chondrite, and HED meteorite, respectively. The dashed circles indicate the clusters. Best view in color. The visualized space is computed by using principal component analysis, and PC1', PC2', and PC3' are the first and second and third principal components, respectively.

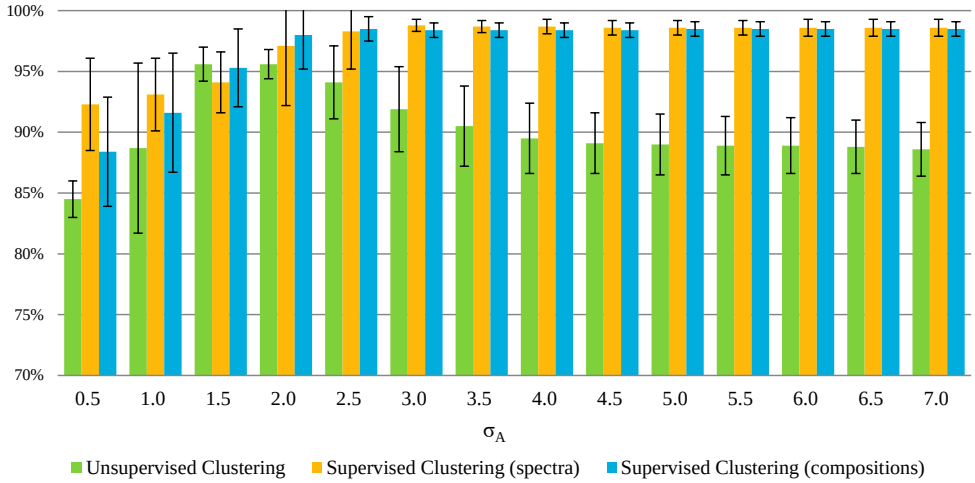
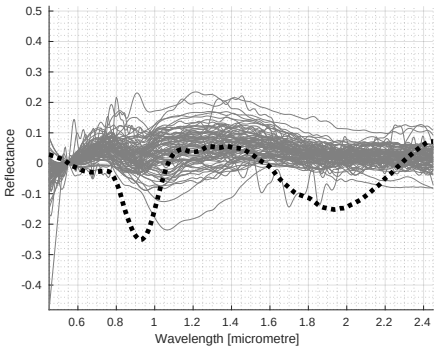
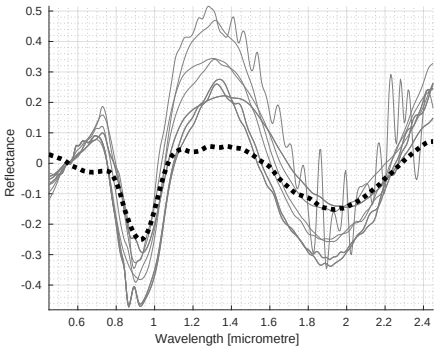


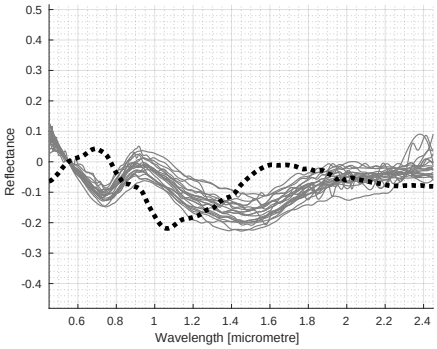
Figure 2.9. Cluster purities of clustering results using resampled dataset with different parameters. Each bar shows the average and standard deviation of the cluster purities. Each parameter of meteorite data is fixed as $\sigma_{\mathrm{M}} = 2.0$ for spectra data, and $\sigma_{\mathrm{M}} = 4.0$ for compositions data.



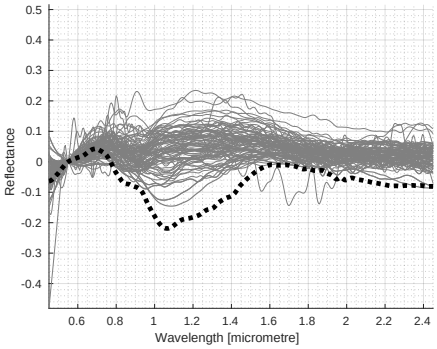
(a) S-type asteroid data (solid lines) and the misclassified V-type asteroid data (dashed line)



(b) V-type asteroid data (solid lines) and the misclassified V-type asteroid data (dashed line)



(c) C-type asteroid data (solid lines) and the misclassified S-type asteroid data (dashed line)



(d) S-type asteroid data (solid lines) and the misclassified S-type asteroid data (dashed line)

Figure 2.10. Reflectance spectra in the respective clusters of (a) S- and (c) C-type asteroids estimated by the supervised clustering with reflectance spectra of meteorite. The dashed lines and solid lines denote the misclassified data in the respective clusters which are labeled another type of the cluster, and the other spectra data in the respective cluster, respectively. (b) and (d) show the respective misclassified data comparing with the data in the "true" cluster. Note that those reflectance spectra are obtained by applying DeMeo's method.

2.4.4 Analysis 2: Resultant links of the asteroid and meteorite

In this analysis, we investigate the results of the supervised clustering, specifically the resulting links between the clusters of the asteroid and meteorite.

We show the resultant links obtained by using the meteorite guidance of the reflectance spectra, or the chemical compositions, with the whole dataset. Figure 2.4 illustrates the former results. The tags attached to the respective asteroidal clusters in Figure 2.4 indicate the matched cluster of the meteorite, described in Figure 2.6. Note that the cluster matching accuracy of 99.2% indicates high certainty of the matching results. Because the cluster matching accuracy is 99.2%, we can say these matching results are supported with a degree of certainty of 99.2%. The same links are found

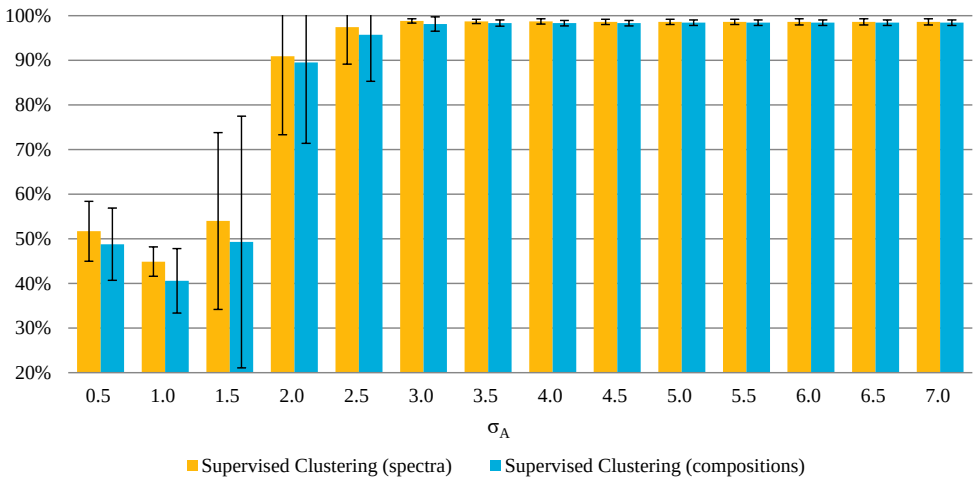


Figure 2.11. Cluster matching accuracies of supervised clustering results with different parameters of reflectance spectra dataset. Each bar shows averages and standard deviations of cluster matching accuracies of supervised clustering results. Each parameter of meteorite data is fixed as $\sigma_{\mathrm{M}} = 2.0$ for spectra data, and $\sigma_{\mathrm{M}} = 4.0$ for compositions data.

in the results of the composition guidance. The results in Figure 2.7 and Figure 2.8 show a cluster matching accuracy of 98.4%. Based on these results, the hypothesis linking the asteroid and meteorite is supported with high certainty.

Here, for further analysis, we inspect the misclassified data observed in the above results. Figure 2.10 depicts the reflectance spectra of the asteroid data comparing "successfully-classified" and "misclassified" cases, and the dashed lines denote the misclassified data. In the sequel, the asteroid clusters that match with the carbonaceous chondorite, ordinary chondorite, and HED, are called C-type, S-type, and V-type clusters, respectively. We first show the details of the misclassified data obtained by using the meteorite guidance of the reflectance spectra. One of the V-type asteroids in Figure 2.4 is misclassified to the S-type cluster and the dashed line in Figure 2.10a shows the spectrum in comparison with the successfully-classified data. Note that, for simplicity of the visualization, we randomly selected 80 data of the S-type asteroids. We compare the misclassified V-type asteroid (dashed line) with the other V-type asteroids (solid lines) in Figure2.10b. The dashed line has different features from the others: there is no absorption under 0.8 µm, and it is very flat around 1.4 µm. While it has the typical absorption bands of V-type asteroid around 0.9 and 2.0 µm, the absorption features are weak. As our asteroidal dataset has very few V-type data, the deviated data was classified into an incorrect cluster. In addition, we suspect that this misclassification was caused by the spectral diversity within a given

asteroid taxonomic type; Hardersen et al. (Hardersen et al., 2014; Hardersen, Reddy, and Roberts, 2015; Hardersen et al., 2018) show that there are varied asteroid spectra within a given asteroid type, so this misclassified data might be one of such spectra data. Second, we investigate the misclassified data for chemical composition guidance. In Figure 2.7, we found that there are two "misclassified" asteroids; one is the same V-type asteroid as the above, and the other is an S-type asteroid misclassified to the C-type cluster. Its spectrum is shown in Figure 2.10c (dashed line) in comparison with "successfully-classified" data (solid lines). We also compare the misclassified S-type asteroid (dashed line) with the other S-type asteroids (solid lines) in Figure 2.10d, in which the dashed line has very low reflectance comparing with the solid lines, causing misclassification. We note that reflectance spectra can show different features, e.g., some CV-chondrite meteorites have an olivine feature (Gaffey, 1976), so treating all asteroids in a given taxonomy as spectrally or mineralogically equivalent might cause misclassification.

Next, using the resampling, we describe the average cluster matching accuracy. The bars of "Supervised Clustering (spectra)" in Figure 2.11 show the average matching accuracies of the supervised clustering for the resampled datasets with various parameters of σ_A (σ_M is fixed as 2.0), using the meteorite guidance of the reflectance spectra. It shows that the average accuracies are high and stable with $\sigma_A \geq 3.5$. The "Supervised clustering (compositions)" in Figure 2.11 shows the average matching accuracies by the chemical composition guidance for various parameters of σ_A (σ_M is fixed as 4.0). It shows that the average accuracies are high and stable with $\sigma_A \geq 2.5$. Based on the results, the links over the asteroid and meteorite are of high certainty.

Those analyses show that the taxonomical guidance of meteorite data improves the clustering of asteroids, implying that the common taxonomic structure exists over the meteorite and asteroid domains. Furthermore, the results of supervised clustering link the taxonomical clusters between the meteorite and asteroid.

2.5 Discussion

We have discussed common taxonomic structure over the asteroid and meteorite domains by the data-driven approach. For this purpose, we checked whether the taxonomy information of meteorite improves the classification of asteroid data. The

numerical experiments with supervised clustering showed that the taxonomical information of meteorites, in fact, improves the accuracy of clustering results of asteroids, when evaluated by the standard labeling by experts; this serves as a piece of evidence that there is a common taxonomic structure and links between these two domains. As a first step, this chapter considers linking between the three types of meteorites and the corresponding three types of asteroids: carbonaceous chondrite, ordinary chondrite, and HED in meteorites, while C-, S-, and V-type in asteroids. Note that, in addition to comparing the spectral reflectance of meteorites and asteroids, the chemical compositions of meteorites were also used for the supervised clustering, showing similar results with the reflectance spectra. This implies that, with different type of measurements, there is a common structure in an abstract sense among the cluster structure of asteroids and meteorites, providing stronger support for the common taxonomic structure.

In this chapter, we focus on well-known taxonomic systems having relatively high certain links; connecting C-, S-, and V-type asteroids to carbonaceous chondrite, ordinary chondrite, and Howardite-Eucrite-Diogenite meteorites, respectively. We consider our matching system is applicable to another case. In the future work, we are planning to chapter other systems, e.g., (Tholen, 1984; Bus and Binzel, 2002; Barucci et al., 1987; Chapman, Morrison, and Zellner, 1975), and other links for matching taxonomies.

2.6 More Details of Supervised Clustering

In this section, we give details on the proposed method of supervised clustering, including its derivation, for the general problem of taxonomy matching for two domains, X and Y. By abuse of notation, we use X and Y for datasets also. We assume that domain X is the target domain to be clustered, and domain Y is utilized as guidance for clustering of domain X.

To represent the features of data sufficiently, we use *kernel methods* (Schölkopf and Smola, 2002), which is a popular approach for nonlinear data analysis in the machine learning field. Suppose we have data $\mathbf{x} = \{x_1, x_2, \cdots, x_{N_X}\}$ in domain X, where N_X is the number of data. We use a feature vector $\phi(x_i)$ to extract the nonlinear feature of data x_i. Here, the feature mapping ϕ is defined with a positive definite kernel k, namely, $\phi(x) = k(\cdot, x)$. The vector $\phi(x_i)$ is included in the feature space, which

is in general infinite dimensional functional space. For the details, see a standard textbook, e.g., (Schölkopf and Smola, 2002).

A cluster is represented by the mean of the feature vectors;

$$\frac{1}{|N_X^i|}\sum_{x_i \in C_X^i} \phi(x_i). \tag{2.6}$$

Let W_X is a cluster assignment matrix, which is an $N_X \times C$ binary matrix with C cluster-size. In matrix notation, the clusters can be represented by

$$\phi\left(\mathbf{x}\right) W_X D_X, \qquad \text{s.t.} \qquad W_X 1_C = 1_{N_X}, \tag{2.7}$$

where $\phi(\mathbf{x}) = (\phi(x_1), \ldots, \phi(x_{N_X}))$ is the matrix of feature vectors, D_X is the reciprocal of cluster sizes as defined by Eq. (2.3), and 1_d is the d-dimensional vector with all elements 1. The constraints $W_X 1_C = 1$ means that only one of the elements is 1, indicating the selected cluster. The cluster assignment matrix W_Y for Y is defined similarly, and the clusters in the feature space are represented by the vectors

$$\varphi\left(\mathbf{y}\right) W_Y D_Y, \qquad \text{s.t.} \qquad W_Y 1_C = 1_{N_Y}, \tag{2.8}$$

where N_Y is data-size on domain Y, $\varphi\left(\cdot\right)$ is the feature map defined by a kernel ℓ on domain Y.

Let μ_X and μ_Y be the centroids of the C mean vectors corresponding to the clusters for X and Y, respectively. The mutual relation among the C vectors in respective feature space can be represented by the covariance matrix

$$S_X := (D_X W_X^T \phi(\mathbf{x})^T - 1_C \mu_X^T)(\phi(\mathbf{x}) W_X D_X - \mu_X 1_C)$$

and

$$S_Y := (D_Y W_Y^T \varphi(\mathbf{y})^T - 1_C \mu_Y^T)(\varphi(\mathbf{y}) W_Y D_Y - \mu_Y 1_C^T),$$

respectively. By using the famous kernel trick, the inner products can be computed the Gram matrices, i.e., $\phi(\mathbf{x})^T \phi(\mathbf{x}) = (k_X(x_i, x_j))_{ij}$ and $\varphi(\mathbf{y})^T \varphi(\mathbf{y}) = k_Y(y_i, y_j))_{ij}$.

For matching the cluster structures in the two domains, we use the similarity of the cluster structure matrices (covariance of clusters) S_X and S_Y; this can be done by

considering the standard inner product of the two matrices, i.e.,

$$\mathrm{Tr}[S_X S_Y].$$

By introducing the $C \times C$ centering matrix $H_{ij} = \delta_{ij} - 1/C$, the above trace can be explicitly written by

$$\mathrm{Tr}\left((W_X D_X)^{\mathrm{T}} K W_X D_X H \, (W_Y D_Y)^{\mathrm{T}} L W_Y D_Y H \right) \tag{2.9}$$

where K and L are the Gram matrices of X and Y, respectively. By normalizing Eq. (2.9), we finally derive objective function of the supervised clustering described in Eq. (2.5).

Remark 1: Eq. (2.6) is called *kernel mean* (Smola et al., 2007) in machine learning literatures, and popularly used for expressing the distribution of data.

Remark 2: Eq. (2.9) is the similarity of the two covariance matrices of clusters, and can be represented as

$$\mathrm{Tr}\left(H \bar{K} H \bar{L} \right) \tag{2.10}$$

where $\bar{K}$ is $(W_X D_X)^{\mathrm{T}} K W_X D_X$, and $\bar{L}$ is $(W_Y D_Y)^{\mathrm{T}} L W_Y D_Y$. This is equal to the empirical HSIC (Gretton et al., 2005), which is a popular dependence measure of two variables.

2.7 Additional Results

Analysis 1: Unsupervised Clustering of Asteroid Datasets

We show unsupervised clustering results with different deviation parameters in Table 2.1.

Analysis 2: Supervised Clustering Results with Meteorite Guidance

With meteorite guidance of reflectance spectral dataset

Table 2.2 shows supervised clustering results using guidance of meteorite reflectance spectra with various parameters, without random sampling. The best cluster purity was unexpectedly 100% with $\sigma_{\mathrm{A}} = 2.0$ and $\sigma_{\mathrm{M}} = 0.5$, so that random sampling is needed for fair analysis. Table 2.3 shows cluster purities of supervised clustering results with randomly sampled datasets. By comparing results of unsupervised clustering and supervised clustering in Table 2.2 and Table 2.3, we can see

that the unsupervised clustering results have been improved by the supervised clustering with most of parameters.

For more analysis, we show cluster matching accuracy without random sampling in Table 2.4, and with sampling in Table 2.5. Because unsupervised clustering in itself cannot match clusters, cluster matching accuracy is shown only for the evaluation of supervised clustering results.

With meteorite guidance of element composites dataset

Cluster purities using meteorite guidance of element composites dataset without and with random sampling are shown in Table 2.6 and Table 2.7, respectively. Supervised clustering have improved the unsupervised clustering results by using cluster information of meteorite element composites, under various parameter settings.

Cluster matching accuracies without and with random sampling are shown in Table 2.8 and Table 2.9, respectively.

Table 2.1. Cluster purities of spectral clustering results with different parameters on asteroid reflectance spectra dataset. 1st column shows different deviation parameter settings. 2nd column shows cluster purities of unsupervised clustering results with fixed dataset, and 3rd column shows average cluster purities and standard deviations of unsupervised clustering results with randomly sampled datasets.

σ_A	Cluster Purity (fixed)	Cluster Purity (sampled)
0.5	84.4%	84.0% ±0.6
1.0	83.6%	84.5% ±2.9
1.5	92.6%	92.8% ±3.8
2.0	91.8%	91.6% ±1.5
2.5	91.0%	90.4% ±1.9
3.0	89.3%	89.3% ±2.5
3.5	89.3%	88.7% ±2.6
4.0	89.3%	88.1% ±2.6
4.5	89.3%	87.6% ±2.6
5.0	88.5%	87.3% ±2.6
5.5	86.9%	87.2% ±2.6
6.0	86.9%	87.1% ±2.6
6.5	86.9%	86.9% ±2.5
7.0	86.9%	86.8% ±2.5

Table 2.2. Cluster purities of supervised clustering results on asteroid reflectance spectra dataset using meteorite reflectance spectra dataset with different parameters (without random sampling). Green color means improvements from unsupervised clustering result (2nd column in Table 2.1), red color means vice versa, black color means no change on cluster purity. For cluster initialization of supervised clustering we utilized unsupervised clustering results with the same deviation parameters.

		σ_M									
		0.5	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**3.5**	**4.0**	**4.5**	**5.0**
σ_A	**0.5**	82.0%	86.9%	93.4%	93.4%	93.4%	93.4%	93.4%	93.4%	93.4%	93.4%
	1.0	92.6%	92.6%	92.6%	90.2%	89.3%	89.3%	89.3%	89.3%	89.3%	89.3%
	1.5	99.2%	92.6%	92.6%	92.6%	92.6%	92.6%	92.6%	92.6%	92.6%	92.6%
	2.0	100.0%	99.2%	91.8%	83.6%	83.6%	83.6%	83.6%	83.6%	83.6%	83.6%
	2.5	99.2%	99.2%	99.2%	99.2%	91.8%	91.0%	83.6%	83.6%	83.6%	83.6%
	3.0	98.4%	98.4%	98.4%	98.4%	98.4%	95.9%	95.9%	95.9%	95.9%	95.9%
	3.5	98.4%	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	4.0	98.4%	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	4.5	98.4%	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	5.0	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	5.5	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	6.0	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	6.5	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	7.0	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%

Table 2.3. Cluster purities (average) and standard deviations of supervised clustering results on asteroid reflectance spectra dataset using meteorite reflectance spectra dataset with different parameters (with random sampling). Green color means improvements from unsupervised clustering result (3rd column in Table 2.1), red color means vice versa, black color means no change on cluster purity. For cluster initialization of supervised clustering we utilized unsupervised clustering results with the same deviation parameters.

		σ_M									
		0.5	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**3.5**	**4.0**	**4.5**	**5.0**
σ_A	**0.5**	86.3% ±4.4	89.8% ±4.7	91.5% ±4.0	92.2% ±3.5	92.5% ±2.9	92.7% ±2.4	92.8% ±2.1	92.8% ±2.1	92.7% ±2.2	92.6% ±2.4
	1.0	93.0% ±0.6	93.2% ±1.7	93.6% ±0.6	93.6% ±0.8	93.6% ±0.8	93.6% ±0.8	93.6% ±0.8	93.6% ±0.8	93.6% ±0.8	93.6% ±0.8
	1.5	94.8% ±3.0	93.6% ±2.8	93.9% ±3.5	92.9% ±4.7	92.0% ±5.3	91.4% ±5.2	91.0% ±5.2	90.7% ±5.3	90.5% ±5.1	90.6% ±4.9
	2.0	96.6% ±3.4	97.4% ±5.3	93.3% ±6.7	90.6% ±7.4	89.5% ±7.1	88.5% ±6.7	87.6% ±6.0	87.0% ±5.7	86.7% ±5.6	86.2% ±4.9
	2.5	98.8% ±1.4	98.9% ±1.0	97.2% ±3.7	94.8% ±5.4	92.9% ±6.5	92.0% ±6.6	91.6% ±6.8	91.2% ±6.9	91.0% ±6.8	90.6% ±6.8
	3.0	98.8% ±0.7	98.7% ±0.5	98.0% ±1.9	97.2% ±2.8	96.4% ±3.7	95.7% ±4.5	95.1% ±5.1	94.6% ±5.3	94.4% ±5.4	94.0% ±5.6
	3.5	98.6% ±0.7	98.6% ±0.5	98.3% ±1.1	97.9% ±1.5	97.6% ±1.6	97.3% ±1.8	97.2% ±1.9	97.0% ±2.1	96.8% ±2.5	96.7% ±2.6
	4.0	98.5% ±0.8	98.4% ±0.6	98.3% ±0.8	97.9% ±1.1	97.8% ±1.3	97.6% ±1.4	97.6% ±1.3	97.5% ±1.4	97.4% ±1.5	97.4% ±1.5
	4.5	98.5% ±0.9	98.4% ±0.6	98.1% ±0.8	98.0% ±0.8	97.9% ±0.8	97.8% ±0.9	97.7% ±1.0	97.6% ±1.1	97.6% ±1.1	97.6% ±1.2
	5.0	98.4% ±0.9	98.3% ±0.6	98.0% ±0.7	98.0% ±0.8	97.9% ±0.8	97.9% ±0.8	97.9% ±0.8	97.8% ±0.8	97.8% ±0.8	97.8% ±0.8
	5.5	98.4% ±0.9	98.3% ±0.7	98.0% ±0.7	97.9% ±0.7	97.8% ±0.7	97.8% ±0.8	97.8% ±0.8	97.8% ±0.8	97.8% ±0.8	97.8% ±0.8
	6.0	98.3% ±0.9	98.2% ±0.7	98.0% ±0.7	97.9% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7
	6.5	98.2% ±1.0	98.2% ±0.7	97.9% ±0.8	97.9% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7
	7.0	98.2% ±0.9	98.2% ±0.7	97.9% ±0.8	97.9% ±0.7	97.9% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.6	97.8% ±0.7

Table 2.4. Cluster matching accuracies of supervised clustering results on asteroid reflectance spectra dataset using meteorite reflectance spectra dataset with different parameters (without random sampling). For cluster initialization of supervised clustering we utilized unsupervised clustering results with the same deviation parameters.

		σ_M									
		0.5	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**3.5**	**4.0**	**4.5**	**5.0**
σ_A	**0.5**	11.5%	44.3%	48.4%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%	50.0%
	1.0	48.4%	58.2%	56.6%	55.7%	54.9%	54.9%	54.9%	54.9%	54.9%	54.9%
	1.5	0.8%	45.1%	45.1%	45.9%	45.9%	45.9%	45.9%	45.9%	45.9%	45.9%
	2.0	77.9%	99.2%	91.8%	51.6%	52.5%	52.5%	52.5%	51.6%	52.5%	52.5%
	2.5	77.0%	99.2%	99.2%	99.2%	91.8%	91.0%	83.6%	82.0%	82.0%	50.8%
	3.0	77.0%	98.4%	98.4%	98.4%	98.4%	95.9%	95.9%	95.9%	95.9%	95.9%
	3.5	77.0%	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	4.0	77.0%	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	4.5	77.0%	98.4%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	5.0	77.0%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	5.5	77.0%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	6.0	77.0%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	6.5	77.0%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%
	7.0	77.0%	98.4%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%	97.5%

Table 2.5. Cluster matching accuracies (average) and standard deviations of supervised clustering results on asteroid reflectance spectra dataset using meteorite reflectance spectra dataset with different parameters (with random sampling). For cluster initialization of supervised clustering we utilized unsupervised clustering results with the same deviation parameters.

		σ_M									
		0.5	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**3.5**	**4.0**	**4.5**	**5.0**
σ_A	**0.5**	45.2% ±9.0	47.3% ±7.4	51.1% ±8.9	51.0% ±6.4	51.1% ±5.3	50.7% ±3.6	50.5% ±2.0	50.6% ±2.0	50.7% ±1.7	50.6% ±2.0
	1.0	41.8% ±23.7	43.8% ±9.1	44.7% ±3.4	45.2% ±3.1	45.3% ±3.1	45.3% ±3.0	45.4% ±3.0	45.4% ±3.0	45.4% ±2.9	45.4% ±2.9
	1.5	21.2% ±24.0	54.6% ±19.2	60.3% ±21.0	60.4% ±20.8	59.8% ±19.9	58.5% ±18.6	57.1% ±17.2	56.8% ±16.6	55.2% ±14.9	54.5% ±14.2
	2.0	56.4% ±24.5	92.1% ±18.8	87.3% ±17.3	73.8% ±23.1	70.1% ±22.6	66.7% ±21.3	64.6% ±19.9	62.9% ±18.8	61.8% ±18.0	60.3% ±16.5
	2.5	76.5% ±6.0	98.9% ±1.0	97.1% ±4.2	93.5% ±8.7	87.3% ±16.8	83.6% ±19.5	80.6% ±21.2	79.1% ±21.6	78.3% ±21.6	77.9% ±21.4
	3.0	77.5% ±0.4	98.7% ±0.5	98.0% ±1.9	97.2% ±2.8	96.4% ±3.8	95.6% ±4.9	94.5% ±7.0	93.8% ±8.1	93.0% ±9.6	92.3% ±10.5
	3.5	77.5% ±0.4	98.6% ±0.5	98.3% ±1.1	97.9% ±1.5	97.6% ±1.6	97.3% ±1.8	97.2% ±1.9	97.0% ±2.1	96.8% ±2.5	96.7% ±2.6
	4.0	77.5% ±0.4	98.4% ±0.6	98.3% ±0.8	97.9% ±1.1	97.8% ±1.3	97.6% ±1.4	97.6% ±1.3	97.5% ±1.4	97.4% ±1.5	97.4% ±1.5
	4.5	77.5% ±0.4	98.4% ±0.6	98.1% ±0.8	98.0% ±0.8	97.9% ±0.8	97.8% ±0.9	97.7% ±1.0	97.6% ±1.1	97.6% ±1.1	97.6% ±1.2
	5.0	77.5% ±0.4	98.3% ±0.6	98.0% ±0.7	98.0% ±0.8	97.9% ±0.8	97.9% ±0.8	97.9% ±0.8	97.8% ±0.8	97.8% ±0.8	97.8% ±0.8
	5.5	77.5% ±0.4	98.3% ±0.7	98.0% ±0.7	97.9% ±0.7	97.8% ±0.7	97.8% ±0.8	97.8% ±0.8	97.8% ±0.8	97.8% ±0.8	97.8% ±0.8
	6.0	77.5% ±0.4	98.2% ±0.7	98.0% ±0.7	97.9% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7
	6.5	77.5% ±0.4	98.2% ±0.7	97.9% ±0.8	97.9% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7
	7.0	77.5% ±0.4	98.2% ±0.7	97.9% ±0.8	97.9% ±0.7	97.9% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.7	97.8% ±0.6	97.8% ±0.7

Table 2.6. Cluster purities of supervised clustering results on asteroid reflectance spectra dataset using meteorite element compositions dataset with different parameters (without random sampling). Green color means improvements from unsupervised clustering result (2nd column in Table 2.1), red color means vice versa, black color means no change on cluster purity. For cluster initialization of supervised clustering we utilized unsupervised clustering results with the same deviation parameters.

		σ_M									
		0.5	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**3.5**	**4.0**	**4.5**	**5.0**
σ_A	**0.5**	82.8%	82.8%	82.8%	82.8%	82.8%	82.8%	93.4%	93.4%	86.1%	93.4%
	1.0	92.6%	92.6%	92.6%	92.6%	92.6%	92.6%	92.6%	92.6%	92.6%	90.2%
	1.5	92.6%	92.6%	92.6%	92.6%	100.0%	100.0%	95.1%	92.6%	92.6%	92.6%
	2.0	99.2%	99.2%	99.2%	99.2%	99.2%	99.2%	98.4%	93.4%	91.8%	83.6%
	2.5	99.2%	99.2%	99.2%	99.2%	98.4%	98.4%	98.4%	98.4%	93.4%	91.8%
	3.0	99.2%	99.2%	99.2%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	96.7%
	3.5	99.2%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%
	4.0	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	4.5	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	5.0	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	5.5	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	6.0	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	6.5	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	7.0	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%	97.5%

Table 2.7. Cluster purities (average) and standard deviations of supervised clustering results on asteroid reflectance spectra dataset using meteorite element compositions dataset with different parameters (with random sampling). For cluster initialization of supervised clustering we utilized unsupervised clustering results with the same deviation parameters.

		σ_M									
		0.5	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**3.5**	**4.0**	**4.5**	**5.0**
σ_A	**0.5**	85.6% ±4.3	85.1% ±3.8	85.3% ±3.9	84.8% ±3.9	85.2% ±3.8	86.0% ±4.3	86.3% ±4.3	87.2% ±4.5	88.7% ±4.3	89.3% ±4.0
	1.0	92.2% ±2.6	92.1% ±2.7	92.4% ±2.3	92.7% ±1.6	92.9% ±1.0	93.0% ±0.6	93.1% ±0.6	93.2% ±0.7	93.1% ±1.0	93.1% ±1.0
	1.5	93.2% ±2.5	93.1% ±2.4	93.5% ±2.8	94.2% ±3.3	95.8% ±3.9	96.1% ±3.9	94.9% ±4.4	93.9% ±5.3	93.0% ±5.5	91.8% ±5.4
	2.0	96.8% ±3.1	98.0% ±2.4	98.5% ±2.0	98.7% ±1.7	98.7% ±1.5	98.2% ±2.5	96.8% ±4.3	94.2% ±5.6	91.1% ±6.1	88.4% ±5.6
	2.5	98.9% ±1.0	99.1% ±0.5	99.0% ±0.6	98.9% ±0.6	98.8% ±0.8	98.4% ±1.4	97.5% ±2.4	96.3% ±3.3	94.6% ±4.5	93.2% ±5.3
	3.0	98.9% ±0.6	98.9% ±0.6	98.9% ±0.6	98.8% ±0.6	98.7% ±0.7	98.5% ±1.0	98.2% ±1.3	97.4% ±2.1	96.7% ±2.8	95.7% ±3.5
	3.5	98.8% ±0.7	98.8% ±0.7	98.8% ±0.7	98.7% ±0.8	98.6% ±0.7	98.5% ±0.6	98.3% ±0.8	98.0% ±1.3	97.4% ±2.1	96.7% ±2.5
	4.0	98.7% ±0.8	98.7% ±0.7	98.6% ±0.8	98.6% ±0.8	98.5% ±0.7	98.4% ±0.6	98.3% ±0.6	98.2% ±0.8	97.7% ±1.3	97.3% ±1.6
	4.5	98.6% ±0.8	98.6% ±0.8	98.6% ±0.8	98.6% ±0.8	98.4% ±0.7	98.4% ±0.6	98.3% ±0.6	98.1% ±0.7	97.9% ±1.0	97.5% ±1.2
	5.0	98.5% ±0.9	98.6% ±0.8	98.5% ±0.8	98.5% ±0.8	98.4% ±0.7	98.4% ±0.6	98.3% ±0.7	98.1% ±0.7	97.9% ±0.9	97.6% ±1.0
	5.5	98.5% ±0.9	98.5% ±0.9	98.5% ±0.9	98.5% ±0.8	98.4% ±0.7	98.3% ±0.6	98.3% ±0.7	98.1% ±0.8	97.9% ±0.8	97.7% ±0.9
	6.0	98.4% ±1.0	98.5% ±0.9	98.4% ±0.9	98.4% ±0.8	98.3% ±0.7	98.3% ±0.7	98.3% ±0.7	98.1% ±0.7	97.9% ±0.8	97.7% ±0.8
	6.5	98.4% ±1.0	98.4% ±0.9	98.4% ±0.9	98.4% ±0.8	98.3% ±0.7	98.3% ±0.7	98.3% ±0.7	98.1% ±0.8	97.9% ±0.8	97.7% ±0.8
	7.0	98.4% ±1.0	98.4% ±0.9	98.4% ±1.0	98.4% ±0.8	98.3% ±0.7	98.3% ±0.7	98.2% ±0.7	98.1% ±0.8	97.9% ±0.8	97.7% ±0.8

Table 2.8. Cluster matching accuracies of supervised clustering results on asteroid reflectance spectra dataset using meteorite element compositions dataset with different parameters (without random sampling). For cluster initialization of supervised clustering we utilized unsupervised clustering results with the same deviation parameters.

		σ_M									
		0.5	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**3.5**	**4.0**	**4.5**	**5.0**
σ_A	**0.5**	10.7%	10.7%	10.7%	10.7%	10.7%	10.7%	32.0%	32.0%	43.4%	32.8%
	1.0	1.6%	0.8%	0.8%	0.8%	0.8%	58.2%	58.2%	56.6%	56.6%	55.7%
	1.5	6.6%	6.6%	0.8%	0.8%	6.6%	6.6%	11.5%	49.2%	49.2%	49.2%
	2.0	99.2%	99.2%	99.2%	99.2%	99.2%	99.2%	98.4%	93.4%	91.8%	76.2%
	2.5	99.2%	99.2%	99.2%	99.2%	98.4%	98.4%	98.4%	98.4%	93.4%	91.8%
	3.0	99.2%	99.2%	99.2%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	96.7%
	3.5	99.2%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%
	4.0	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	4.5	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	5.0	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	5.5	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	6.0	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	6.5	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%
	7.0	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	98.4%	97.5%	97.5%

Table 2.9. Cluster matching accuracies (average) and standard deviations of supervised clustering results on asteroid reflectance spectra dataset using meteorite element compositions dataset with different parameters (with random sampling). For cluster initialization of supervised clustering we utilized unsupervised clustering results with the same deviation parameters.

		σ_M									
		0.5	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**3.5**	**4.0**	**4.5**	**5.0**
σ_A	**0.5**	41.0% ±14.4	37.9% ±16.6	42.1% ±12.8	42.8% ±8.8	42.4% ±8.6	43.5% ±7.7	44.1% ±9.5	46.0% ±10.8	45.7% ±7.7	46.8% ±7.8
	1.0	22.6% ±22.5	20.9% ±23.0	22.1% ±21.2	25.8%2 ±1.8	30.3% ±19.7	37.0% ±13.7	37.8% ±11.4	38.2% ±9.6	39.3% ±8.4	38.9% ±8.4
	1.5	21.8% ±24.5	27.2% ±28.2	24.2% ±30.3	25.0%3 ±2.8	28.2% ±34.3	30.5% ±34.3	36.3% ±34.5	50.6% ±33.6	55.6% ±29.3	54.8% ±25.2
	2.0	67.5% ±30.2	70.6% ±39.0	83.1% ±31.8	85.5%3 ±0.6	88.3% ±29.5	88.4% ±28.6	89.1% ±24.7	88.4% ±19.2	83.8% ±18.0	78.0% ±17.2
	2.5	90.2% ±11.0	86.8% ±30.2	98.1% ±9.2	98.9% ±0.6	98.8% ±0.8	98.4% ±1.4	97.5% ±2.4	96.3% ±3.6	94.1% ±5.9	91.9% ±7.8
	3.0	90.8% ±10.5	86.7% ±30.1	97.9% ±9.2	98.8% ±0.6	98.7% ±0.7	98.5% ±1.0	98.2% ±1.3	97.4% ±2.1	96.7% ±2.8	95.6% ±3.7
	3.5	90.7% ±10.4	86.6% ±30.1	97.8% ±9.2	98.7% ±0.8	98.6% ±0.7	98.5% ±0.6	98.3% ±0.8	98.0% ±1.3	97.4% ±2.1	96.7% ±2.5
	4.0	90.6% ±10.3	86.5% ±30.1	97.7% ±9.2	98.6% ±0.8	98.5% ±0.7	98.4% ±0.6	98.3% ±0.6	98.2% ±0.8	97.7% ±1.3	97.3% ±1.6
	4.5	90.6% ±10.3	86.4% ±30.1	97.6% ±9.2	98.6% ±0.8	98.4% ±0.7	98.4% ±0.6	98.3% ±0.6	98.1% ±0.7	97.9% ±1.0	97.5% ±1.2
	5.0	90.5% ±10.3	86.3% ±30.1	97.6% ±9.2	98.5% ±0.8	98.4% ±0.7	98.4% ±0.6	98.3% ±0.7	98.1% ±0.7	97.9% ±0.9	97.6% ±1.0
	5.5	90.5% ±10.3	86.3% ±30.0	97.6% ±9.2	98.5% ±0.8	98.4% ±0.7	98.3% ±0.6	98.3% ±0.7	98.1% ±0.8	97.9% ±0.8	97.7% ±0.9
	6.0	90.5% ±10.2	86.3% ±30.0	97.5% ±9.2	98.4% ±0.8	98.3% ±0.7	98.3% ±0.7	98.3% ±0.7	98.1% ±0.7	97.9% ±0.8	97.7% ±0.8
	6.5	90.4% ±10.2	86.2% ±30.0	97.5% ±9.2	98.4% ±0.8	98.3% ±0.7	98.3% ±0.7	98.3% ±0.7	98.1% ±0.8	97.9% ±0.8	97.7% ±0.8
	7.0	90.4% ±10.2	86.2% ±30.0	97.5% ±9.2	98.4% ±0.8	98.3% ±0.7	98.3% ±0.7	98.2% ±0.7	98.1% ±0.8	97.9% ±0.8	97.7% ±0.8

Chapter 3

Exchangeable Deep Neural Networks for Set-to-Set Matching and Learning

3.1 Motivation

Matching pairs of data is a crucial part of many machine learning tasks, including recommendation (Sarwar et al., 2001; Rendle, Freudenthaler, and Schmidt-thieme, 2010; Le, Lauw, and Fang, 2019), person re-identification (re-id) (Zheng et al., 2015), image search (Wang et al., 2014), face recognition (Parkhi, Vedaldi, Zisserman, et al., 2015), as typical industrial applications.

Aside from these tasks, set-to-set matching, which is an extension of multiple instance matching, has recently been identified as an important element in various applications required by emerging web technologies or services. A representative example in e-commerce is fashion recommendation, where a group of fashion items deemed to match the collection of fashion items already owned by a user is recommended. Regarding the group as an unordered set, we can consider this task a set-to-set matching problem, as shown in Figure 3.1. Another example is group re-identification (group re-id) in surveillance systems (Lisanti et al., 2017; Xiao et al., 2018; Lin et al., 2019). Other examples include image-set retrieval (Gao et al., 2018; Feng, Karaman, and Chang, 2017), image-set classification (Lu et al., 2015), image-set reconstruction (Liu et al., 2019a), person re-id (Liu, Yan, and Ouyang, 2017), taxonomy matching (Saito et al., 2020), cross-lingual matching (Iwata et al., 2017), relational data matching (Iwata, Lloyd, and Ghahramani, 2015), and face verification (Liu et al., 2019b; Xie, Shen, and Zisserman, 2018). Earlier studies have also

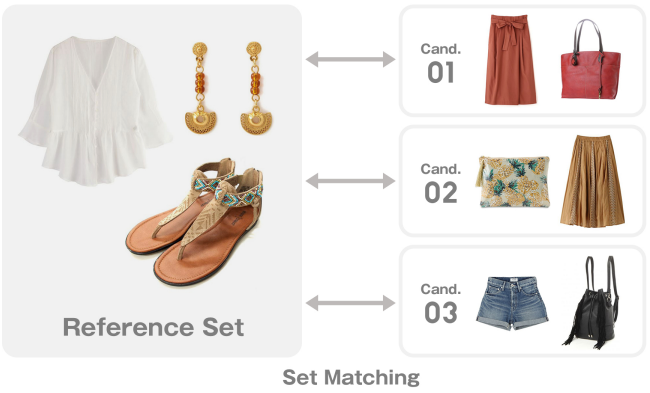


Figure 3.1. One of the main questions that set-to-set matching attempts to answer is as follows: which candidate is more compatible than others with the reference set? Here, we consider the matching of the reference set and the respective candidate set and then selecting the best pair.

explored face recognition as a set-to-set matching problem (Shakhnarovich, Fisher, and Darrell, 2002; Arandjelovic et al., 2005; Cevikalp and Triggs, 2010; Yamaguchi, Fukui, and Maeda, 1998) and next-basket recommendation (Rendle, Freudenthaler, and Schmidt-thieme, 2010).

Set-to-set matching scenarios can be grouped into two classes: homogeneous set matching and heterogeneous set matching as described in Chapter 1. To the best of our knowledge, there are very few studies on constructing deep learning frameworks for heterogeneous set matching. We consider that matching heterogeneous sets requires a strong learning architecture to match different sets.

Furthermore, as described in Chapter 1, another fundamental difficulty in set-to-set matching, compared with ordinary data matching, lies in the two types of exchangeability required: exchangeability between the pair of sets and invariance across different permutations of the items in each set. A function that calculates a matching score should provide an invariant response, regardless of the order of the two sets, or the permutations of the items.

The main focus of this chapter is an architecture that preserves the aforementioned exchangeability properties, and at the same time, realizes a high performance in heterogeneous set matching tasks. In this chapter, we argue that allowing the feature extractor and matching layer to include interactions between the two sets is crucial to identify matching pairs among different items. We propose a deep learning model for (1) feature extraction, named *cross-set feature transformation* (CSeFT), which

iteratively provides the interactions between the pair of sets to each other in the intermediate layers. Our novel functions, *attention-* and *affinity-based functions*, organize the CSeFT spanning two different sets in the feature spaces, thereby improving the feature representations. The proposed architecture also includes (2) a matching layer, named *cross-similarity function* (CS function), that calculates the matching score between the features of the set members across the two sets accurately. Our model guarantees both types of exchangeability in the modules. Figure 3.2 shows the proposed architecture.

We examine the set-to-set matching problem in a supervised setting, where examples of correctly paired sets are deployed as training data. The objective is to train the feature extractor and matching layer in an end-to-end manner such that the appropriate sets of features to be matched can be extracted. To train the model efficiently, we also propose a novel training framework, *K-pair-set loss*. Following training, the model is then used to find correct pairs of sets among a group of candidates.

The effectiveness of our approach is demonstrated in two real-world applications. First, we consider fashion set recommendations, where provided examples of the outfits are used as correct combinations of items (clothes). Using a large number of examples of the outfits in the form of images, we aim to match the correct pair of defined sets through subset and superset matching tasks using the IQON dataset (Nakamura and Goto, 2018). Since two positive sets include images of different fashion items, we regard this case as heterogeneous set matching. In these tasks, taking into account combinations of items is required to fully consider fashion compatibility (Han et al., 2017; He, Packer, and McAuley, 2016). Next, we evaluate our methods through group re-id experiments using two datasets, a new extension of the Market-1501 dataset (Zheng et al., 2015) (Market-1501 Group) and the Road Group dataset (Xiao et al., 2018). The Market-1501 Group is composed of two categories of individual person images, taken under *noisy* and *non-noisy* conditions, which may change group membership in the paired sets. We also provide experimental results on a more practical problem using the Road Group dataset. Considering group membership change, we regard group re-id as a heterogeneous set matching problem. In the fashion set recommendations and group re-id experiments performed on the Market-1501 Group dataset, our methods show significant improvements and better results compared with state-of-the-art methods. We also

performed the group re-id experiment on the Road Group dataset using the data augmentation method that we developed for the pair set (set-data augmentation); our methods show competitive results without using any external datasets or spatial layout information in each group.

The main contributions of this chapter are as follows. (i) A novel deep learning architecture is proposed to provide the two types of exchangeability required for set-to-set matching. (ii) The proposed feature extractors using the interactions between two sets are shown to extract better features for heterogeneous set matching. (iii) A new loss function, *K*-pair-set loss, is proposed and provide better performances in our tasks. (iv) We introduce set-input methods into group re-id tasks (Road Group) using a new set-data augmentation, thereby showing competitive results without using external datasets or spatial relations. (v) The proposed models show state-of-the-art results for the fashion set recommendation and group re-id, supporting the claim that the interactions improve both the accuracy and robustness of the set-matching procedure.

3.2 Preliminaries: Set-to-Set Matching

We introduce the necessary notation as follows. Let $\boldsymbol{x}_n, \boldsymbol{y}_m \in \mathfrak{X} = \mathbb{R}^d$ be feature vectors representing the features of each individual item. Let $\mathcal{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ and $\mathcal{Y} = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_M\}$ be *sets* of these feature vectors, where $\mathcal{X}, \mathcal{Y} \in 2^{\mathfrak{X}}$.

The function $f : 2^{\mathfrak{X}} \times 2^{\mathfrak{X}} \to \mathbb{R}$ calculates a matching score between the two sets $\mathcal{X}$ and $\mathcal{Y}$. Guaranteeing the exchangeability of the set-to-set matching requires that the matching function $f(\mathcal{X}, \mathcal{Y})$ is *symmetric* and *invariant* under any permutation of items within each set.

We consider tasks where the matching function f is used to select a correct matching. Given candidate pairs of sets $(\mathcal{X}, \mathcal{Y}^{(k)})$, where $\mathcal{X}, \mathcal{Y}^{(k)} \in 2^{\mathfrak{X}}$ and $k \in \{1, \cdots, K\}$, we choose $\mathcal{Y}^{(k^*)}$ as a correct one so that $f(\mathcal{X}, \mathcal{Y}^{(k^*)})$ achieves the maximum score from amongst the K candidates. In this chapter, a supervised learning setting is considered, where the function f is trained to classify the correct pair and unmatched pairs.

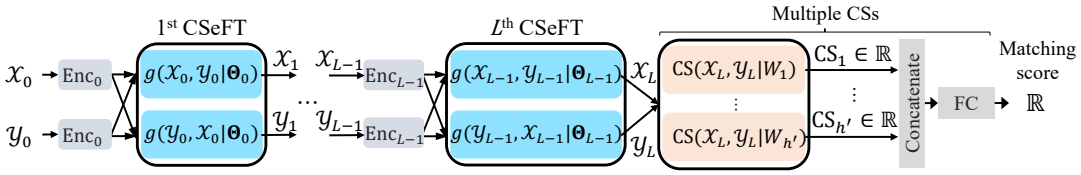


Figure 3.2. Our model calculates a matching score between the paired sets. Enc_i, CSeFT, CS, and FC indicate an $(i+1)$-th (one-layered) encoder sharing weights within the same layer, cross-set feature transformation, cross-similarity function, and fully connected layer, respectively. We exclude the multihead structure in g.

3.2.1 Mappings of Exchangeability

We present a brief review on several notions of exchangeability, which are used in building our models.

Permutation Invariance. A set-input function f is said to be *permutation invariant* if

$$f(\mathcal{X}, \mathcal{Y}) = f(\pi_x \mathcal{X}, \pi_y \mathcal{Y}) \tag{3.1}$$

for permutations π_x on $\{1, \ldots, N\}$ and π_y on $\{1, \ldots, M\}$.

Permutation Equivariance. A map $f : \mathfrak{X}^N \times \mathfrak{X}^M \to \mathfrak{X}^N$ is said to be *permutation equivariant* if

$$f(\pi_x \mathcal{X}, \pi_y \mathcal{Y}) = \pi_x f(\mathcal{X}, \mathcal{Y}) \tag{3.2}$$

for permutations π_x and π_y, where π_x and π_y are on $\{1, \ldots, N\}$ and $\{1, \ldots, M\}$, respectively. Note that f is permutation invariant for permutations within $\mathcal{Y}$.

Symmetric Function. A map $f : 2^{\mathfrak{X}} \times 2^{\mathfrak{X}} \to \mathbb{R}$ is said to be *symmetric* if

$$f(\mathcal{X}, \mathcal{Y}) = f(\mathcal{Y}, \mathcal{X}). \tag{3.3}$$

Two-Set-Permutation Equivariance. Given $\mathcal{Z}^{(1)} \in \mathfrak{X}^N$ and $\mathcal{Z}^{(2)} \in \mathfrak{X}^M$, a map $f : \mathfrak{X}^* \times \mathfrak{X}^* \to \mathfrak{X}^* \times \mathfrak{X}^*$ is said to be *two-set-permutation equivariant* if

$$pf(\mathcal{Z}^{(1)}, \mathcal{Z}^{(2)}) = f(\mathcal{Z}^{(p(1))}, \mathcal{Z}^{(p(2))}) \tag{3.4}$$

for any permutation operator p exchanging the two sets, where $\mathfrak{X}^* = \cup_{n=0}^{\infty} \mathfrak{X}^n$ indicates a sequence of arbitrary length such as $\mathfrak{X}^N$ or $\mathfrak{X}^M$.

3.3 Matching and Learning for Sets

In this section, based on the problem scenario as explained above, we describe our set-to-set matching methods. We describe in detail the architecture of the (1) feature extractor, *cross-set feature transformation* (CSeFT) in Section 3.3.1, and (2) matching layer, *cross-similarity* (CS) function in Section 3.3.2. Figure 3.3 shows the model of CSeFT. Finally, we discuss training procedures in Section 3.3.3. Figure 3.4 depicts the framework of our K-pair-set loss.

3.3.1 Cross-Set Feature Transformation

We construct the architecture of the feature extractor, which transforms sets of features using the interactions between the pair of sets, and extracts the desired features to be matched in the post-processing stages (Figure 3.3).

Here, consider the transformation of a pair of set-feature vectors $(\mathcal{X}, \mathcal{Y})$ into new feature representations on $\mathfrak{X}^N \times \mathfrak{X}^M$, using two-set-permutation equivariant functions. Let i be the iteration (layer) number of the CSeFT layers. Our feature extraction then can be described as a map of $(\mathcal{X}_i, \mathcal{Y}_i) \to (\mathcal{X}_{i+1}, \mathcal{Y}_{i+1})$, where $\mathcal{X}_{i+1}, \mathcal{X}_i \in \mathfrak{X}^N$, $\mathcal{Y}_{i+1}, \mathcal{Y}_i \in \mathfrak{X}^M$, $\mathcal{X}_{i+1} = (\boldsymbol{x}_{(n,i+1)})_{n=1}^N$, $\mathcal{X}_i = (\boldsymbol{x}_{(n,i)})_{n=1}^N$, $\mathcal{Y}_{i+1} = (\boldsymbol{y}_{(m,i+1)})_{m=1}^M$, and $\mathcal{Y}_i = (\boldsymbol{y}_{(m,i)})_{m=1}^M$. For example, $\boldsymbol{x}_{(n,i)} \in \mathfrak{X}$ denotes the feature vector extracted by the i-th layer representing the n-th item, $\boldsymbol{x}_n$, and $\boldsymbol{y}_{(m,i)}$ is defined similarly. Note that the initial feature vectors with $i = 0$ are found with a typical feature extractor, i.e., a deep convolutional neural network (CNN) for the image of each item. Then, we construct a parallel architecture of CSeFT, with an asymmetric transformation g, as follows:

$$\text{cross-set feature transformation (CSeFT)}: \begin{cases} \mathcal{X}_{i+1} &= g(\mathcal{X}_i, \mathcal{Y}_i | \boldsymbol{\Theta}_i) \\ \mathcal{Y}_{i+1} &= g(\mathcal{Y}_i, \mathcal{X}_i | \boldsymbol{\Theta}_i), \end{cases} \tag{3.5}$$

where $g : \mathfrak{X}^* \times \mathfrak{X}^* \to \mathfrak{X}^*$ is a permutation equivariant function, which transforms the set features in the first argument into new feature representations regardless of the order of the set features in the second argument, $\mathfrak{X}^* = \cup_{n=0}^{\infty} \mathfrak{X}^n$, which indicates a sequence of arbitrary length such as $\mathfrak{X}^N$ or $\mathfrak{X}^M$, and $\boldsymbol{\Theta}_i$ is learnable weights shared in the same layer. Also, residual paths (He et al., 2016) may be used in Eq. (3.5) if required. Figure 3.3 shows the model of our CSeFT.

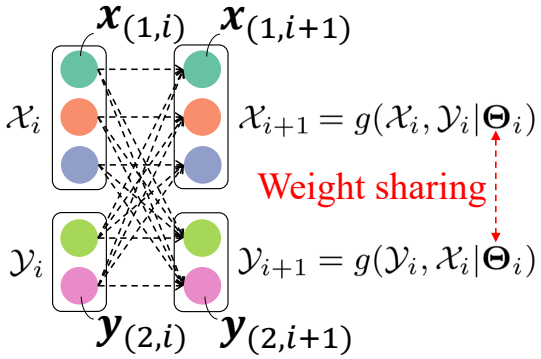


Figure 3.3. A diagram of CSeFT. Here, we assume $|\mathcal{X}| = 3$ and $|\mathcal{Y}| = 2$. The colors indicate the respective set members.

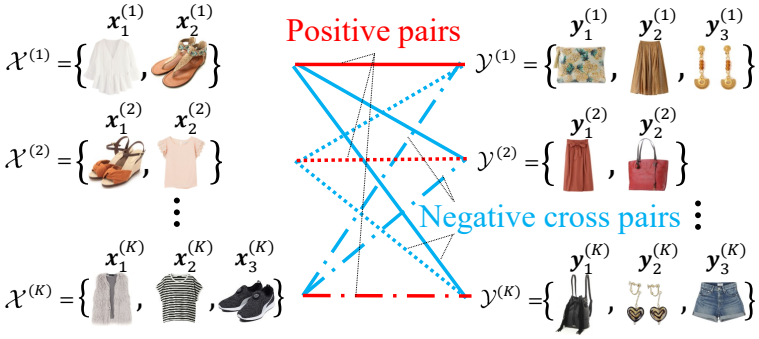


Figure 3.4. K-pair-set-based matching candidates. Red and blue lines indicate correct pairs $(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)})$ and negative cross pairs $(\mathcal{X}^{(k)}, \mathcal{Y}^{(k')}) : \forall k' \neq k$, where $k, k' \in \{1, \cdots, K\}$, respectively.

We propose two possible feature extractors for g: an *attention-based function*, and an *affinity-based function*. Both are constructed to assign the *matched* feature vectors to the *reference* feature vector, taking account of interactions between the two sets. For simplicity, we provide an explanation via the case of extracting the features for $\mathcal{X}$ as follows (we can easily exchange $\mathcal{X}$ and $\mathcal{Y}$ for $\mathcal{Y}$).

The attention-based function of $g(\mathcal{X}_i, \mathcal{Y}_i|\boldsymbol{\Theta}_i)$ maps $\boldsymbol{x}_{(n,i)} \to \boldsymbol{x}_{(n,i+1)}$ as follows:

$$\boldsymbol{x}_{(n,i+1)} = \frac{1}{|\mathcal{Y}_i|} \sum_{\boldsymbol{y} \in \mathcal{Y}_i} \left(\frac{l_i^{(1)}(\boldsymbol{x}_{(n,i)})^{\mathrm{T}} l_i^{(2)}(\boldsymbol{y})}{\sqrt{d_g}} \right)_+ l_i^{(3)}(\boldsymbol{y}), \tag{3.6}$$

where $n \in \{1, \cdots, N\}$, $\boldsymbol{\Theta}_i = \{\Theta_i^{(1)}, \Theta_i^{(2)}, \Theta_i^{(3)}\}$, $\Theta_i^{(j)} \in \mathbb{R}^{d_g \times d}$, $|\mathcal{Y}_i| = M$, $l_i^{(j)}$ denote a linear transformation, i.e., $l_i^{(j)}(\boldsymbol{x}) := \Theta_i^{(j)} \boldsymbol{x}$, and $()_+$ is a non-negative mapping, i.e., ReLU (Glorot, Bordes, and Bengio, 2011), which introduces nonlinear interactions between the two elements. Here, $d_g = d$ if a multihead structure is not utilized. Our attention-based function transforms the respective feature vectors based on attention calculated via the inner product between set members of $\mathcal{X}$ and $\mathcal{Y}$.

Note that our attention-based function has a strong relation to dot-product attention (Vaswani et al., 2017; Lee et al., 2019), which has in the past been introduced to calculate the weighted average on $\mathcal{Y}$ using softmax as the coefficients. However, the softmax operation would be inconsistent with our matching objective, as through normalization it increases the coefficients even in unmatched cases of $\mathcal{X}$ and $\mathcal{Y}$. To preserve non-linearity, we use instead the non-negative weighted sum and then average it using Eq. (3.6).

The affinity-based function of $g(\mathcal{X}_i, \mathcal{Y}_i|\mathbf{\Theta}_i)$ maps $\boldsymbol{x}_{(n,i)} \to \boldsymbol{x}_{(n,i+1)}$ as follows:

$$\boldsymbol{x}_{(n,i+1)} = \frac{1}{2}\left(\bar{\boldsymbol{x}}_{(n,i)} + \frac{1}{|\bar{\mathcal{Y}}_i|}\sum_{\bar{\boldsymbol{y}}\in\bar{\mathcal{Y}}_i}\left(\frac{\bar{\boldsymbol{x}}_{(n,i)}^{\mathrm{T}}\bar{\boldsymbol{y}}}{\sqrt{d_g}}\right)_+ \bar{\boldsymbol{y}}\right), \tag{3.7}$$

where $\mathbf{\Theta}_i = \{\Theta_i^{(1)}, \Theta_i^{(2)}\}$, $\bar{\boldsymbol{x}}_{(n,i)} = l_i^{(1)}(\boldsymbol{x}_{(n,i)})$, and $\bar{\mathcal{Y}}_i = \{l_i^{(2)}(\boldsymbol{y}_{(m,i)})\}_{m=1}^{M}$. Using the two linear transformations $l_i^{(1)}$ and $l_i^{(2)}$, the affinity-based function combines the resembling feature vectors within different sets so that the feature vectors for $\mathcal{X}$ have similar representations to the linearly transformed vectors in $\mathcal{Y}$.

Other simple permutation equivariant functions of g, e.g., $\boldsymbol{x}_{(n,i+1)} = \boldsymbol{x}_{(n,i)} + \frac{1}{|\mathcal{Y}_i|}\sum_{\boldsymbol{y}\in\mathcal{Y}_i}\boldsymbol{y}$, may be utilized. However, we consider it a function incapable of extracting appropriate enough features without any rich interactions between the two sets to yield accurate matching for two sets.

Instead of performing g singly, we introduce a multihead structure (Vaswani et al., 2017) to our feature extractor g, which is also a permutation equivariant function. Denoting the output of $g(\mathcal{X}_i, \mathcal{Y}_i|\mathbf{\Theta}_i^{(j)})$ as $g_{\mathcal{X}_i}^{(j)}$, the multihead version of g is defined as $\Theta_h \mathrm{Concat}\left(g_{\mathcal{X}_i}^{(1)}, \cdots, g_{\mathcal{X}_i}^{(h)}\right)$, where Concat indicates a concatenation for each corresponding set member in $g_{\mathcal{X}_i}^{(j)}$, $\Theta_h \in \mathbb{R}^{d\times hd_g}$, and $hd_g = d$. Note that the multihead structure is related to recent models such as MobileNet (Howard et al., 2017), which isolates and places the convolutional operations in parallel to reduce the calculation costs whilst preserving the accuracy of the recognition. We assume that the multihead structure provides various interactions between the set members, reducing the calculation costs as well.

3.3.2 Calculating Matching Score for Sets

We introduce a matching layer to calculate the matching score between two given sets, mapping $2^{\mathfrak{X}} \times 2^{\mathfrak{X}} \to [0, \infty]$. It is designed to calculate the inner product for

every combination of set members across sets, so we call this *cross-similarity* (CS), defined as follows:

$$\mathrm{CS}(\mathcal{X}, \mathcal{Y}|W) := \frac{1}{|\mathcal{X}||\mathcal{Y}|} \sum_{\boldsymbol{x}\in\mathcal{X}} \sum_{\boldsymbol{y}\in\mathcal{Y}} \left(\frac{l(\boldsymbol{x}|W)^{\mathrm{T}} l(\boldsymbol{y}|W)}{\sqrt{d_w}} \right)_+, \tag{3.8}$$

where x and y are feature vectors in $\mathcal{X}$ and $\mathcal{Y}$, respectively, l is a linear function allowing conversions into a lower-dimensional space using learnable weights $W \in \mathbb{R}^{d_w \times d}$, i.e., $l(\boldsymbol{x}|W) := W\boldsymbol{x}$, and d_w is the number of dimensions of the lower-dimensional space. CS can be seen as a calculation of the average similarity in the linear subspaces created by the dimensionality reduction l, or the normalized and non-negative inner product if both sets contain only one set member.

Instead of calculating CS singly, we utilize multiple CSs (mCS) to combine the CSs calculated with different linear mappings. The procedure runs as follows:

$$\mathrm{mCS}(\mathcal{X}, \mathcal{Y}|\boldsymbol{W}) = l(\mathrm{Concat}\,(\mathrm{CS}_1, \cdots, \mathrm{CS}_{h'})\,|W_o), \tag{3.9}$$

where $\boldsymbol{W} = \{W_1, \cdots, W_{h'}, W_o\}$, $\mathrm{CS}_j = \mathrm{CS}(\mathcal{X}, \mathcal{Y}|W_j) \in \mathbb{R}$, and the linear function l with learnable weights W_o maps $\mathbb{R}^{h'} \to \mathbb{R}$.

Because CS is permutation invariant, mCS is also permutation invariant:

Property 1. *Both CS and mCS are permutation invariant.*

Additionally, because CS is symmetric (definition in Eq. (3.3)), mCS is symmetric as well:

Property 2. *Both CS and mCS are symmetric.*

These symmetric and permutation invariance properties entails that CS and mCS satisfy the exchangeability criterion for the pair of sets, i.e., $\mathrm{CS}(\mathcal{X}, \mathcal{Y}|W) = \mathrm{CS}(\mathcal{Y}, \mathcal{X}|W)$ and $\mathrm{mCS}(\mathcal{X}, \mathcal{Y}|\boldsymbol{W}) = \mathrm{mCS}(\mathcal{Y}, \mathcal{X}|\boldsymbol{W})$, and permutations within each set.

Since CS is a symmetric and permutation invariant function, mCS is also symmetric and permutation invariant. Combined with the fact that CSeFT is a two-set-permutation equivariant function, we have the following:

Proposition 1. *A composition of the function* CS *or* mCS *with the cross-set transformation* CSeFT, *i.e.*, CS $\circ$ CSeFT *or* mCS $\circ$ CSeFT, *is symmetric and permutation invariant.*

Proof. See Section 3.7.

Note that we can stack CSeFTs in a way that preserves the symmetric architecture, by combining with other networks that operate upon the sets or items independently. We discuss the overall architecture in Section 3.5.2.

3.3.3 Training for Pairs of Sets

Next, to allow for comparison against the scores for other matching candidates, the output of mCS or CS is fed into a loss function. That is, the task of maximizing the matching score is translated into a minimization of the loss function.

We attempt to train our model efficiently using multiple correct pairs taken together. As described in the problem formulation, K candidates are provided to find the correct one-to-one matching. Here, K candidates per reference set of $\mathcal{X}$ are fed into the matching process in each training iteration. However, if we prepare K different candidates for each reference set of $\mathcal{X}$, the calculations for data processing would be inefficient.

To train our model efficiently, we create matching candidates from the correct pairs (Figure 3.4). Let $(\mathcal{X}^{(k)}, \mathcal{Y}^{(k)})$ be a correct pair of sets, where $k \in \{1, \cdots, K\}$. From those K-pair, by extracting all $\mathcal{Y}^{(k)}$, we create the set of $\mathcal{Y}^{(k)}$ as $\boldsymbol{Y} = \{\mathcal{Y}^{(1)}, \cdots, \mathcal{Y}^{(K)}\}$. That is, $\boldsymbol{Y}$ is composed of sets exhibiting correct relations to the respective $\mathcal{X}^{(k)}$, and $\boldsymbol{Y}$ can be used as a set of candidates for each $\mathcal{X}^{(k)}$ in the training stage. We construct positive pairs and negative cross pairs from these candidates by assuming that one correct pair exists for the respective sets, as described in Section 3.2. Then, we train our models using these pairs with a conventional softmax cross-entropy loss. Compared with a typical mini-batch training, suppose that the set size is n on average, the K-pair training method utilizes $2nK$ data (images) per training iteration; this can be regarded as the size of the mini-batch. We consider the above training method as a set version of N-pair loss (Sohn, 2016), so we call this K-pair-set loss.

If the quantity of set data is not large, we can use other training frameworks, e.g., a triplet loss with the softplus function (Hermans, Beyer, and Leibe, 2017); we can use the triplet loss for the relations among the reference set of $\mathcal{X}$, the positive candidate $\mathcal{Y}^{(p)}$, and the negative candidate $\mathcal{Y}^{(n)}$, where $p, n \in \{1, \cdots, K\}$.

3.4 Related Works

Set-Input Methods. Deep learning architecture for set data is developing and has been well studied (Li et al., 2016; Vinyals, Bengio, and Kudlur, 2015; Lee et al., 2019; Zaheer et al., 2017; Murphy et al., 2018; Hayat, Bennamoun, and An, 2014; Pang et al., 2019; Bello, Yu, and Wang, 2020; Zhou et al., 2017a; Zhou et al., 2017b; Zhong, Arandjelović, and Zisserman, 2018; Vollgraf, 2019; Wagstaff et al., 2019; Yarotsky, 2018; Sannai, Takai, and Cordonnier, 2019; Zhang, Hare, and Prügel-bennett, 2019). In the work of Lee et al. (Lee et al., 2019), the state-of-the-art set-feature representation was introduced by applying a self-attention based Transformer (Vaswani et al., 2017) to a set data. An encoder–decoder model, called Set Transformer, is trained through supervised/unsupervised learning; it transforms the set data into a vector/matrix representation in the feature space, and recognizes the set feature. Zaheer et al. (Zaheer et al., 2017) derived a condition for the property of permutation invariance/equivariance in functions, and introduced an operator referred to as deep sets. These models can manage set data that serve multiple objectives, such as set classification, calculation from images, text retrieval, etc. However, constructing a deep learning model that can manage multiple sets has not been well studied.

Furthermore, various methods have been proposed for representing a set as described in Chapter 1. However, these methods were mainly proposed to model homogeneous sets; they do not include feature learning schemes for paired sets.

Methods for Measuring Distributions. In the literature on statistical machine learning, matching multiple data is related to measuring the distance between two distributions (Gretton et al., 2005; Muandet et al., 2017; Póczos et al., 2012; Muandet et al., 2012; Zhang et al., 2017; Li et al., 2017). However, to the best of our knowledge, deep learning for measuring distribution has not been well studied.

Methods for Heterogeneous data. Many studies have investigated heterogeneous data. For example, topic models for entity matching (Yang et al., 2015), graph matching for malware detection (Wang et al., 2019), multiple instance learning for anomaly detection (Sultani, Chen, and Shah, 2018), and various methods for multi-domain data (Liu et al., 2012; Li, Lei, and Ao, 2009; Tang and Wang, 2004; Klare and Jain, 2010; Klare, Li, and Jain, 2010; Yi et al., 2007; Lei and Li, 2009; Eitz et al., 2010; Torfi et al., 2017; Jiang and Li, 2017; Venugopalan et al., 2015) have been proposed.

Attention models. Recently, several studies have investigated attention functions (Jain

and Wallace, 2019; Ilse, Tomczak, and Welling, 2018; Yang et al., 2016; Hu, Shen, and Sun, 2018; Lee et al., 2019; Vaswani et al., 2017). Compared with the above studies, our works focus on investigating and developing the application of attention functions for set-to-set matching task.

Applications. Many fashion item recommendation studies have investigated natural combinations of fashion items, the so-called visual fashion compatibility, to recommend fashion items or outfits (Han et al., 2017; He, Packer, and McAuley, 2016; Hsiao and Grauman, 2018; Vasileva et al., 2018). In this chapter, the main difficulties of the subset/superset matching procedures lie in satisfying the fashion compatibility requirements of the matched sets.

In the applications of group re-id (Lisanti et al., 2017; Xiao et al., 2018; Lin et al., 2019; Zheng, Gong, and Xiang, 2009; Cai, Takala, and Pietikainen, 2010; Huang et al., 2019; Zhu, Chu, and Yu, 2016), multi-shot person re-id (Wang and Zhao, 2014; Zhu et al., 2017), and tracking (Solera et al., 2016), problems of multiple instance matching arise. One group re-id scenario has been proposed that the detection of known groups from videos (Lin et al., 2019) is required. Also, two group re-id datasets, the Road Group dataset, and the DukeMTMC Group dataset[1] have been constructed (Lin et al., 2019), which include bounding box annotations for each person. Our experiments focus on set-to-set matching using these given cropped images.

Methods for Non-Exchangeable Data. Many powerful data-processing methods have been proposed based on specific data structures (Bai et al., 2019; Guo et al., 2018; Li et al., 2019; Bai et al., 2018; Zanfir and Sminchisescu, 2018; Fey et al., 2020; Guo et al., 2018; Yoshida, Takeuchi, and Karasuyama, 2019; Mudgal et al., 2018; Si et al., 2018; Caspi, Simakov, and Irani, 2006). In natural language processing, Devlin et al. achieved state-of-the-art results in various tasks using the bidirectional encoder representations from transformers (BERT) (Devlin et al., 2018). Furthermore, Cucurull et al. applied graph neural networks (GNNs) to predict fashion compatibility between related fashion items using graph structures (Cucurull, Taslakian, and Vazquez, 2019). Although the data in those tasks are known to be non-exchangeable, we still consider that comparing these promising models with our model is possible and necessary.

[1] Note that the DukeMTMC (Ristani et al., 2016) is no longer available.

3.5 Experiments

We conduct set-to-set matching in the three scenarios: subset/superset matching and group re-id. We present an ablation study to show the validity of our models.

3.5.1 Baselines for Comparisons

We validate our architecture through comparison with other set-matching models. However, to the best of our knowledge, studies using deep neural networks for matching two heterogeneous sets are non-existent. Instead, we use extensions from the state-of-the-art set-input method and the promising models in other related domains to a set-to-set matching procedure, and consider this acceptable for the comparison.

We briefly explain straightforward extensions of Set Transformer (Lee et al., 2019), BERT (Devlin et al., 2018), and GNN (Cucurull, Taslakian, and Vazquez, 2019) below.

Set Transformer. We straightforwardly extend the Set Transformer towards the two sets matching method. The Set Transformer transforms a set of feature vectors into a vector on $\mathbb{R}^d$. Denoting the Set Transformer model ST, we perform the extension by calculating the matching score between the two sets $\mathcal{X}$ and $\mathcal{Y}$ via the inner product $\mathrm{ST}(\mathcal{X})^{\mathrm{T}}\mathrm{ST}(\mathcal{Y})$, sharing the weights between the two ST. This extension satisfies the exchangeability criteria for the set-to-set matching, however, no interactions between the pair of sets are provided.

BERT. We consider a union of two sets as a set-input for the extension of BERT and omit the individual token embedding, i.e., the position embedding. We use the segment embedding to designate items of $\mathcal{X}$ and $\mathcal{Y}$. We use three variants; $\mathrm{BERT}_{\mathrm{BASE}}$ is the same model described in (Devlin et al., 2018) using the first-token feature; $\mathrm{BERT}_{\mathrm{SMALL}}$ is a smaller version of $\mathrm{BERT}_{\mathrm{BASE}}$, which has two encoding layers and 512 channels; $\mathrm{BERT}_{\mathrm{BASE-AP}}$ is the average pooling version of $\mathrm{BERT}_{\mathrm{BASE}}$ in the last layer. These promising models provide interactions between the items, but no exchangeability for sets. Also, we omit the pre-training stage of BERT to investigate the effects of differences in the architecture. For classification, in the last layer, we use the first-token features (Devlin et al., 2018), which is selected randomly from the tokens in this chapter, or average pooled features of all the tokens.

GNN. We combine two sets as one input for the extension of GNN, as we did for BERT. We use the same settings described in (Cucurull, Taslakian, and Vazquez,

2019); three-layered graph convolutions and one-step neighbor on the adjacency matrix are used. The training objective is to reconstruct the correct adjacency matrix, in which all the connections are of 1 when the two sets are correct pair, and otherwise of 1 within each set and 0 between different sets. Because this model is not presented to train in an end-to-end manner with the feature extractor (Cucurull, Taslakian, and Vazquez, 2019), we do not finetune the CNN with the GNN in subset/superset matching, where pre-trained CNNs are used, and we simultaneously train the CNN and GNN in group re-id, where we do not use the pre-trained CNN. We use the function of compatibility prediction (Cucurull, Taslakian, and Vazquez, 2019) to score the input is correct or not. Note that we omit the context provided from the external graphs in the evaluation stage to apply this model in the same scenarios of our tasks. The extension of GNN provide interactions between the items, but do not facilitate the exchangeability of the sets.

The other processes and scenarios are the same as those in our methodology.

Also, in our first experiments, we introduce a conventional CNN, trained by Hard-Aware Point-to-Set deep metric learning (HAP2S) (Yu et al., 2018) as a minimum configuration, based on the triplet loss between the anchor point (item), which is randomly selected from positive set $\mathcal{X}$, and other sets $\mathcal{Y}^{(p)}$ and $\mathcal{Y}^{(n)}$, which are a positive and negative one, respectively. Our parameter settings are the same as (Yu et al., 2018) and we use the exponential weighting. To calculate scores, we use a mean squared distance between the feature vectors of each item within two sets extracted via the CNN. Here, the extension of HAP2S does not provide set-based feature extractions.

Comparing the experimental results of our models with the results of other models serves as a performance comparison and also an evaluation for our models, providing insight into whether our architecture is valid or not.

3.5.2 Overall Architecture

In applications of our model, we use an encoder–decoder structure, inspired by the Transformer models (Vaswani et al., 2017; Lee et al., 2019). As an example, Vaswani et al. regarded the Transformer as an encoder–decoder model for text translation (Vaswani et al., 2017); the encoder transforms a set of features within the input domain, and the decoder transforms the resultant set of features onto the output domain. Because the translation is unidirectional, the one-encoder–decoder structure is

included in the Transformer. Meanwhile, an iterative model of the encoder–decoder, e.g., the Stacked Hourglass model, has been proposed and demonstrates a high accuracy in the task of human pose estimation (Newell, Yang, and Deng, 2016). Borrowing from the above architectures, we construct our overall architecture by combining the encoder (Lee et al., 2019), which is a permutation equivariant function called a self-attention block, with the decoder, which is a function of our CSeFT. We then repeat the encoder–decoder structure L times in succession. Here, the encoder is the preprocessing layer of our decoder, serving better feature representations within a set. Note that a function of our CSeFT does not entail interactions within a set, and combining with the encoder or stacking CSeFTs takes account of the full interactions.

We construct our models as follows. We set both the number of CSeFT layers and encoder layers to 2. That is, we iteratively perform the one-layered encoder and the one-layered CSeFT two times in succession. To extract the individual feature vector from one of the images within the set, we use the CNN.

We use two CNNs. For the task of the fashion set recommendation, we use Inception-v3 (Szegedy et al., 2016), which is pre-trained using the ILSVRC-2012 ImageNet dataset (Russakovsky et al., 2015). Using this model, we extract the feature vectors on $\mathbb{R}^{2048}$ extracted by the global average pooling layer. We linearly transform each feature vector into $\mathbb{R}^{512}$ to provide one of the set members, and then the two sets of collected feature vectors are fed into the set-input functions. For the group re-id tasks, we utilize a simple four-layered CNN without any pre-training. This CNN transforms an RGB image into the feature vector mapping $3 \to 64 \to 128 \to 256 \to 512$ channels using 3×3 kernels, and we then apply global average pooling so that the resultant feature vectors are on $\mathbb{R}^{512}$ as well.

The resultant set of feature vectors extracted from each encoder model is fed into the next cross-set feature transformation (CSeFT) and also the respective residual paths. Alongside this, we apply a feed-forward network, which comprises two-layered linear transformations with a leaky ReLU (Maas, Hannun, and Ng, 2013) to the first argument of each function g.

We set the numbers of multihead functions of cross-set transformation function h and multiple cross-similarity functions (mCS) h' to eight. The numbers of dimension sizes d_g and d_w are 64. We set the number of dimension size d of the feature space to 512 except the BERT. The dimension sizes of the feature space are on $\mathbb{R}^{768}$ for $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{BASE}-\text{AP}}$.

In ablation study, we replace our matching layer. We replace the mCS with max pooling, average pooling, projection metric (Huang, Wu, and Van Gool, 2018), covariance matrix (Wang et al., 2012; Cai, Takala, and Pietikainen, 2010), set kernel (Kim et al., 2019), and cosine similarity metric (Nguyen and Bai, 2010). For projection metric, we use the inner product as described in (Huang, Wu, and Van Gool, 2018). For covariance matrix, we calculate two covariance matrices and the inner product between the two matrices (Zhu et al., 2013). Note that we also normalize the calculated similarities as described in (Nguyen and Bai, 2010). For set kernel, we use Gaussian kernel and multiple kernel learning as described in (Li et al., 2017).

3.5.3 Set-Data Augmentation

In this section, we describe our set-data augmentation (set-aug) method. Algorithm 1 shows the set-aug algorithm. As we described in this chapter, given positive person image pairs X and several negative person images Z, we create set pairs randomly on each training iteration. Here, index i is the iteration number in each epoch.

Algorithm 1: Set-Data Augmentation.

Data: paired-image dataset X, noise-image dataset Z, index i
Result: paired sets $(\mathcal{X}, \mathcal{Y})$
begin
  //select an image-pair and create initial paired sets from i-th paired-image in X, where $|\mathcal{X}| = |\mathcal{Y}| = 1$
  $(\mathcal{X}, \mathcal{Y}) \longleftarrow selectPairedImage(X, i)$
  //randomly select multiple paired images
  $(\mathcal{X}', \mathcal{Y}') \longleftarrow randomSelectPairedImage(X)$
  $\mathcal{X} \longleftarrow \mathcal{X} \cup \mathcal{X}'$
  $\mathcal{Y} \longleftarrow \mathcal{Y} \cup \mathcal{Y}'$
  //randomly drop the image(s) and use the remained set
  $\mathcal{X} \longleftarrow randomDrop(\mathcal{X})$
  $\mathcal{Y} \longleftarrow randomDrop(\mathcal{Y})$
  //randomly select the noise image(s) (if possible, select the images captured on the same camera of each target set)
  $\mathcal{X}'' \longleftarrow randomSelectImage(Z)$
  $\mathcal{Y}'' \longleftarrow randomSelectImage(Z)$
  $\mathcal{X} \longleftarrow \mathcal{X} \cup \mathcal{X}''$
  $\mathcal{Y} \longleftarrow \mathcal{Y} \cup \mathcal{Y}''$

3.5.4 Training Settings

In this section, we briefly describe the training settings. The learning settings are as follows. We use a stochastic gradient descent method with a learning rate of 0.005, a momentum of 0.5, and a weight decay of 0.00004. The learning rate is set to degrade every 16 epochs by multiplying by 0.7. For fashion set recommendations, we set the maximum number of epochs to 32, which requires a week for training on Amazon SageMaker ml.p3.8xlarge. For group re-id, we set the maximum number of epochs to 256, which takes a few hours. We set the numbers of matching candidates to 16, 4, and 16 for subset matching, superset matching, and group re-id, respectively. We train both the CNN and set-matching model simultaneously (except for the GNN). In each iteration, we randomly swap pairs of sets and items in each set, to learn all the methods stably.

In the selection of the loss function, to develop an item category constraint between the reference and candidate sets, described in Section 3.5.5, we use the triplet loss with softplus function (Hermans, Beyer, and Leibe, 2017) in the subset matching problem. We use our K-pair-set loss in other tasks.

For group re-identification, we trained the models under the same *noisy* or *non-noisy* settings for each test, to investigate the robustness of the models for the set-to-set matching under noisy situations.

For data augmentation, we randomly flip images horizontally in our tasks and use the set-aug on Road Group dataset using Algorithm 1. In each training iteration, we choose the number of base-set-size $s \in \{3, 4\}$ randomly, and select $s - 1$ paired images using $randomSelectPairedImage$. Furthermore, we add one noise image to each set randomly with a probability of 85%, and drop an image from each set randomly with a probability of 50%.

3.5.5 Fashion Set Recommendation

Dataset. We examine the set-to-set matching for the fashion set recommendation using the *IQON dataset* (Nakamura and Goto, 2018). IQON (www.iqon.jp) is a user-participating fashion web service sharing outfits for women. The IQON dataset consists of recently created, high-quality outfits, including 199,792 items grouped into 88,674 outfits. We split these outfits into groups, using 70,997 for training, 8,842 for

validation, and 8,835 for testing. IQON Dataset (Nakamura and Goto, 2018) contains images with 480×480 size.

To create our training dataset from IQON dataset, we set the maximum and minimum numbers of items for each outfit as eight and four, respectively; if the outfit contains more than eight items, then we randomly select eight items from it. The outfits contain roughly 5.5 items on average. After this operation, we created our training datasets for subset/superset matching.

Preparing Set Pairs. To construct the correct pair of sets to be matched, we randomly halve the given outfit $\mathcal{O}$ into two non-empty proper subsets $\mathcal{X}$ and $\mathcal{Y}$ as follows: $\mathcal{O} \rightarrow \{\mathcal{X}, \mathcal{Y}\}$, where $\mathcal{X} \cap \mathcal{Y} = \emptyset$. We perform our experiments using these subsets, to try to find the correct pairs in *subset matching*. Also, we extend the problem of subset matching to *superset matching*, which presents more complex situations. We consider the superset as a multimodal/mixture set comprising the multiple subsets, which consists of multiple fashion styles.

We expect our model to reconstruct the original outfits $\mathcal{O}$ by combining two subsets/supersets, provided such an inverse mapping exists. In the reconstruction, we assume that the desired features either remain within both the input sets or are extracted during matching. For example, we regard the desired features as the discriminative features, which serve to recognize the fashion compatibility (Han et al., 2017; He, Packer, and McAuley, 2016) or infer the visual styles of the outfits. That is, in the matching of the two subsets/supersets, such desired features to be matched must be obtained.

We perform our experiments using these subsets, to try to find the correct pairs. Here, we consider matching two subsets $\mathcal{X}$ and $\mathcal{Y}$; we call this problem *subset matching*. In the subset matching, K subsets $\{\mathcal{Y}^{(1)}, \cdots, \mathcal{Y}^{(K)}\}$ are provided as a set of matching candidates, whilst maintaining the category restrictions for each fashion item. That is, these K candidates only contain the same-category fashion items and are fed into the training or testing stages. Note that without any category restrictions, the models tend to be trained to select the candidate $\mathcal{Y}^{(k)}$ that contains non-overlapped fashion category items, e.g., shoes, with the gallery subset $\mathcal{X}$. To avoid this situation, we introduce category restrictions to the K candidates in each training/testing iteration.

Additionally, we extend the problem of subset matching to *superset matching*

which presents more complex situations. We choose K outfits $\{\mathcal{O}^{(1)}, \cdots, \mathcal{O}^{(K)}\}$ randomly and split the respective outfits randomly in half $\mathcal{O}^{(i)} \rightarrow \{\mathcal{X}^{(i)}, \mathcal{Y}^{(i)}\}$, where $i \in \{1, \cdots, K\}$. Then we create two supersets $\{\mathcal{X}^{(1)}, \cdots, \mathcal{X}^{(K)}\}$ and $\{\mathcal{Y}^{(1)}, \cdots, \mathcal{Y}^{(K)}\}$. These two supersets serve as a correct pair for the superset matching problem. We consider the superset as a multimodal/mixture set, which consists of multiple fashion styles, such that the matching problem is one of finding similar supersets in terms of these mixed fashion styles. Because each superset has a category overlap of fashion items themselves, providing category restrictions to the candidates is not necessarily required in the superset matching, so we do not give the restrictions. Note that we used $K = 4$ in the training stage and selected $K \in \{2, 4\}$ in the test stage.

Subset/Superset Matching. We discuss the experimental results of the matching subsets/supersets. Table 3.1 shows significantly different results between our models and the baselines. Here, Cross Attention and Cross Affinity denote our models with the attention-based and affinity-based functions, respectively. Comparing the performance of Cross Affinity and $\text{BERT}_{\text{SMALL}}$, which is the most accurate among the baselines, the differences in their accuracies were 9.6% and 6.1%, on average, in subset and superset matching, respectively. Furthermore, a comparison of the results obtained using the attention- and affinity-based function is shown in Table 3.1. It can be seen that the affinity-based function performed better in both the subset and superset matching.

In this experiment, we consider that the components on which the comparative effectiveness of the proposed models depended were potentially three-fold. Compared with the extensions of BERT, (a) our model preserves the exchangeability in two sets, which may ensure that the set features to be matched are accurately represented. Furthermore, (b) our model preserves two set features explicitly, whereas BERT provides a set of features with segment embedding that may have a limitation. Compared with the results of the Set Transformer, our models and BERT yielded accurate results is made possible by (c) providing the strength of interactions between two sets. Therefore, we conclude that these results justify the fine aspects of our architecture.

Figure 3.5 shows the results of the subset matching in a "Top-K" ranking view, using limited candidates up to eight. Using our models, Figure 3.5 shows that top-3

Table 3.1. Accuracy of subset/superset matching (%). Cand and Mix indicate the number of candidates to be matched and number of outfits mixed in the supersets, respectively.

	Subset Matching		Superset Matching			
	Cand:4	Cand:8	Cand:4		Cand:8	
Method			Mix:2	Mix:4	Mix:2	Mix:4
Set Transformer	39.2	22.7	73.5	65.3	57.5	49.6
$\text{BERT}_{\text{SMALL}}$	50.5	33.8	87.3	69.7	77.0	53.0
$\text{BERT}_{\text{BASE}}$	50.5	33.5	86.6	66.1	76.3	50.8
$\text{BERT}_{\text{BASE}-\text{AP}}$	50.0	33.5	86.4	65.4	75.7	49.5
GNN	30.3	17.3	32.4	25.5	17.5	13.4
HAP2S	29.4	16.8	36.6	32.0	20.8	17.8
Cross Attention (ours)	58.1	41.9	88.8	74.3	80.6	58.9
Cross Affinity (ours)	**60.2**	**43.3**	**90.6**	**75.9**	**82.8**	**61.9**

candidates out of eight candidates contained the correct answers in an 80% probability, which might be an acceptable level in real-world applications.

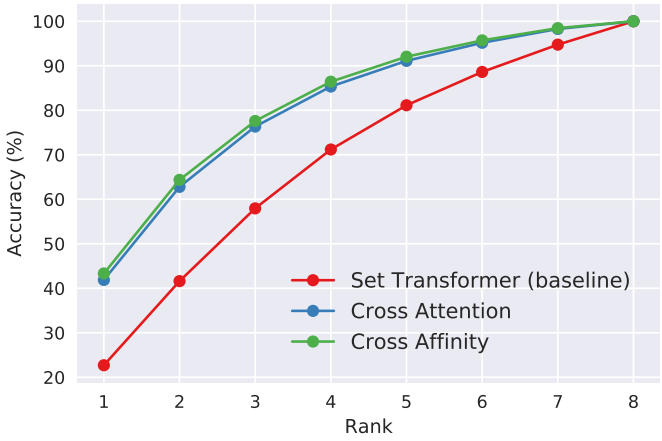


Figure 3.5. Matching accuracy for subset matching using eight candidates. The rank indicates the accuracy with the "Top-K" acceptance setting for evaluation.

3.5.6 Group Re-Identification

We present the results of a group re-id on the Market-1501 Group dataset, a new extension of a well-known person re-id dataset, Market-1501 (Zheng et al., 2015), and the Road Group dataset (Xiao et al., 2018). The task is to identify the pairs of sets that consist of individual images of the (mostly) same multiple persons.

Evaluation on Market-1501 Group dataset. We evaluated the accuracy using the training/validation data, including the query/gallery splits. We regard sets of gallery and query data as $\mathcal{X}$ and $\mathcal{Y}$, respectively.

Figure 3.6. An example of a correct pair for group re-identification. $\mathcal{Y}$ contains four persons, including a "non-target" person who is not included in $\mathcal{X}$. This example is corresponding to the case of $(\frac{0}{3}, \frac{1}{4})$ in Table 3.2.

Market-1501 (Zheng et al., 2015) contains 32,668 annotated bounding boxes (images) of 1,501 identities. The image size is 64×128. The identities were divided into training and testing sets, containing 750 and 751 identities, respectively.

To create our dataset from Market-1501, we ignored camera information of images, owing to less number of query images. Because query data contain 2–5 images per person in which one or zero image provided for each camera position, it is difficult to take into account camera intersections to create image sets of the query and gallery systematically. For the experiment, we construct image sets composed of multiple persons. Each set consists of 3–8 persons and contains three different images of each.

We investigated noise robustness through the experiments to show that our models do not over-fit on the data; here, the *noise* means that random persons that accidentally contained into the group additionally or that the label noise (Jiang et al., 2017) for paired sets generated based on the given noise fraction. Note that the noise persons and label noise have some relations, e.g., a candidate set composed of only noise persons corresponds to a set mislabelled by label noise.

To evaluate experimental results on Market-1501 Group dataset, we set the number of candidates to 5.

Table 3.2 presents the comparison results. In the non-noisy case, many models showed almost perfect accuracies; we consider that *averaging feature vectors in sets* achieves high accuracy in this homogeneous case. In the case the noise person included, the noise ratio was inversely proportional to the accuracy across all the models; however, our models yielded more accurate results, e.g., the average accuracy of Cross Affinity, Set Transformer, and $\text{BERT}_{\text{BASE}-\text{AP}}$ was 87.0, 80.6, and 72.4%, respectively. Because the main differences between the architectures exist in the interactions for paired sets or the exchangeability, the results support the claim that

Table 3.2. Accuracy (%) for Market-1501 Group dataset.

Method	Non-noisy	Ratio of *noise* persons in $\mathcal{X} \times \mathcal{Y}$						Label noise frac.		
		$(\frac{0}{3}, \frac{1}{4})$	$(\frac{1}{4}, \frac{1}{4})$	$(\frac{0}{3}, \frac{3}{6})$	$(\frac{0}{3}, \frac{5}{8})$	$(\frac{3}{6}, \frac{3}{6})$	$(\frac{5}{8}, \frac{5}{8})$	0.2	0.4	0.8
Set Transformer	99.5	95.1	89.9	85.7	80.4	65.7	48.1	**99.3**	98.8	95.6
$\text{BERT}_{\text{SMALL}}$	94.3	77.6	69.2	83.7	64.9	49.5	24.7	99.2	98.7	79.5
$\text{BERT}_{\text{BASE}}$	96.8	80.5	77.6	68.8	69.9	61.9	49.2	98.9	98.1	76.0
$\text{BERT}_{\text{BASE}-\text{AP}}$	97.3	84.4	74.7	70.7	69.3	62.8	47.7	**99.3**	97.5	77.9
GNN	82.0	29.3	46.0	23.7	22.1	29.3	21.1	81.7	73.0	76.7
Cross Attention (ours)	99.6	**96.9**	**94.8**	91.9	90.7	**72.9**	56.1	**99.3**	99.6	95.5
Cross Affinity (ours)	**99.7**	96.5	92.5	**94.4**	**92.4**	72.0	**61.7**	**99.3**	**99.9**	**98.4**

considering these properties improves both the accuracy and robustness. Furthermore, in the case of label noise fraction is 0.8, the permutation invariance would be essential to preserve high accuracy.

Evaluation on Road Group dataset. We conduct experiments on the Road Group dataset (Xiao et al., 2018; Lin et al., 2019), which consists of 162 group pairs taken from a 2-camera-view of a crowded road scene. One image per group for each camera is provided, where most groups do not have the same person's image in common with the different group pairs. The image dataset comprises a total of 1099 pedestrians and the bounding boxes annotated by a person detector or human hands (GT), showing large variations in spatial group layout, group membership change in crowds, and pose transformation. Following the experimental protocol described in (Xiao et al., 2018; Lin et al., 2019), we construct training/validation datasets, splitting the 162 group pairs randomly in half into two different 81 group pairs, and reporting the accuracies calculated by the cumulative matching characteristic (CMC) metric (Moon and Phillips, 2001).

Because group re-id is a newly emerging task, most datasets, including the Road Group dataset, contain a small number of groups and images, and training on such datasets is difficult (Huang et al., 2019). Specifically, our set-to-set matching method extracts features that rely on input set pairs, thus, the variations in the set pairs are crucial. Considering the difference in appearances or camera parameters, however, importing external data (Huang et al., 2019; Huang et al., 2019b) is also a challenging task itself.

To relax the data limitation, we introduce our novel set-data augmentation (set-aug) method that significantly enhances the learning results of the proposed set-to-set matching modules by increasing the training data. Given positive person image pairs and several negative person images, creating set pairs randomly on each training iteration, our set-aug effectively increases the group member variations.

Table 3.3. Evaluation results (%) for Road Group dataset.

Method (detector-based)	CMC-1	CMC-5	CMC-10	CMC-15	CMC-20
Data augmentation ablation					
Cross Affinity (our baseline)	45.2 ± 3.5	77.5 ± 2.9	87.9 ± 3.8	91.9 ± 2.4	94.1 ± 2.1
Baseline + img-aug	47.7 ± 4.2	78.3 ± 3.2	87.7 ± 2.6	91.1 ± 2.4	93.3 ± 1.8
Baseline + set-aug	**84.0** ± 3.6	93.8 ± 0.8	**96.8** ± 0.6	**97.0** ± 1.0	97.5 ± 1.1
Baseline + set-aug + img-aug	81.7 ± 1.9	**94.1** ± 1.3	96.5 ± 1.1	**97.0** ± 0.9	**97.8** ± 0.8
Baseline + set-aug (ours)	**84.0** ± 3.6	93.8 ± 0.8	**96.8** ± 0.6	97.0 ± 1.0	97.5 ± 1.1
MGM w/ spatial layout (Lin et al., 2019)	80.2	93.8	96.3	**97.5**	97.5
MGM w/o spatial layout (Lin et al., 2019)	70.4	90.1	91.3	92.6	96.3
TSCN w/ external data (Huang et al., 2019)	**84.0**	**95.1**	96.3	-	**98.8**
GNN w/ external data (Huang et al., 2019b)	74.1	90.1	92.6	-	**98.8**
Method (GT-based)	CMC-1	CMC-5	CMC-10	CMC-15	CMC-20
Baseline + set-aug (ours)	**85.7** ± 3.7	**96.3** ± 0.8	**97.8** ± 0.5	**98.3** ± 0.6	**98.3** ± 0.6
MGM w/ spatial layout (Lin et al., 2019)	82.4	95.1	96.3	97.5	98.0

Table 3.3 shows the experimental results. The top block in Table 3.3 indicates the results of our methods and three types of data augmentation: (a) the horizontal flipping (Krizhevsky, Sutskever, and Hinton, 2012), which is used to train the baseline model; (b) image-based data augmentation (img-aug), which includes both scale augmentation (Simonyan and Zisserman, 2014; He et al., 2016) and random erasing (Zhong et al., 2017) on images; and (c) our set-aug. Using the 81 pre-defined groups, the baseline model was not very effective, even with img-aug. However, using the set-aug, our method exhibited significant improvements without applying img-aug. These results imply that generating combinations on sets is very beneficial to our models. The other parts in Table 3.3 show that our methods yield very competitive results, compared with the state-of-the-art methods that utilize a large transferred external dataset or auxiliary features such as spatial layout information within each group. Furthermore, compared with MGM w/o spatial layout (Lin et al., 2019), which also does not use the spatial layout information, our methods significantly improved the accuracy of CMC-1 by 13.6%.

3.5.7 Ablation Study

In this section, we report the results of an ablation study performed to highlight the importance of each proposed component. The top part in Table 3.4 shows the two results obtained when our models are trained using triplet loss and the proposed *K*-pair-set loss. Triplet loss triggered slight accuracy degradation, even though the training losses converged to zero in the training stages. We believe that this might be attributable to the nonexistence of a function for mining hard samples, such as

Table 3.4. Ablation study. Average accuracies (%) of group re-id (Market-1501 Group) are shown, where the seven noise patterns, presented in Table 3.2, are included.

Method	Accuracy
Training method ablation	
Cross Affinity (baseline)	**87.0**
Baseline with triplet loss	45.5
Feature extractor ablation	
Baseline with L=1	86.0
Baseline with h=1	85.9
w/o Enc	85.7
w/o CSeFT	82.8
Matching layer ablation	
Single CS	86.0
w/o ReLU in mCS	85.0
Max pooling	86.1
Average pooling	85.8
Projection metric	67.0
Covariance matrix	61.1
Set kernel	53.3
Cosine similarity metric	53.1
Feature & matching layer ablation	
Set Transformer	80.6
Set Transformer$_{UNION}$	66.7

triplet selection (Hermans, Beyer, and Leibe, 2017). On the other hand, the proposed K-pair-set loss can manage to train the models accurately, without selecting hard set pairs. The second topmost part in Table 3.4 shows the results of ablations in the feature extractor. Reducing the number of layers and number of multiheads in the CSeFT, and excluding the encoder and CSeFT, the accuracies are degraded by 1.0, 1.1, 1.3, and 4.2%. In the results, our model performed well without the encoder (1.3% degradation); however, excluding the CSeFT module significantly reduced the accuracy (4.2% degradation). These results imply that the proposed CSeFT module is a crucial part of the set-to-set matching model architecture. The second lower part of Table 3.4 shows the results of ablation study performed on the matching layer. Reducing mCS to a single CS and excluding ReLU from the CS functions reduced the accuracies of both models by 1.0 and 2.0%, respectively. It is interesting to observe that the ReLU was more important than the number of CS functions; this demonstrated the importance of nonlinearity in the matching layer. Furthermore, replacing our mCS with max pooling, average pooling, projection metric (Huang, Wu, and Van Gool, 2018), covariance matrix (Wang et al., 2012; Cai, Takala, and Pietikainen, 2010), set kernel (Kim et al., 2019), and cosine similarity metric (Nguyen

and Bai, 2010) all resulted in significant accuracy degradation implying the effectiveness of our mCS functions. The lowermost part of Table 3.4 shows the results of ablation study performed on the feature extractor and matching layer. The extension of Set Transformer, which does not include the proposed CSeFT module and CS function, yielded significant accuracy degradation. Furthermore, Set Transformer$_{UNION}$, which combines two sets as one input and calculates a matching score by applying a linear layer to the resultant feature vector, also degrades accuracy. These results show the validity of our architecture for heterogeneous set-to-set matching.

3.5.8 Weak Point Analysis

We consider that our models are promising to match a reference and candidate sets in high accuracy, but impose more substantial calculations. For example, a one-set-input function, i.e., the extension of Set Transformer, can transform a set of features individually for two sets to match. Also, after the feature extractions, it does not require calculations except the inner product in matching two vectors. Comparing with the Set Transformer, our models and the extensions of BERT and GNN models need additional calculation costs in matching two sets; they need paired sets for the feature extraction. Figure 3.7 shows examples of the calculation time in a testing stage, where $|\mathbf{Y}|$ indicates the number of candidate sets. The calculation time of these models except for the Set Transformer significantly increased when the number of candidate sets increased.

Reducing the calculation costs preserving the interactions is challenging but interesting, and we leave it as future work.

3.6 Discussion

In this chapter, we investigated the heterogeneous set-to-set matching problem. We proposed a novel architecture comprising the (1) cross-set feature transformation (CSeFT) module and (2) cross-similarity (CS) function, in addition to a loss function and set-data augmentation for performing set-to-set matching.

We showed that our architecture preserves the two types of exchangeability for a pair of sets and also the items within them (Proposition 1), thereby satisfying the requirements of set-to-set matching procedure.

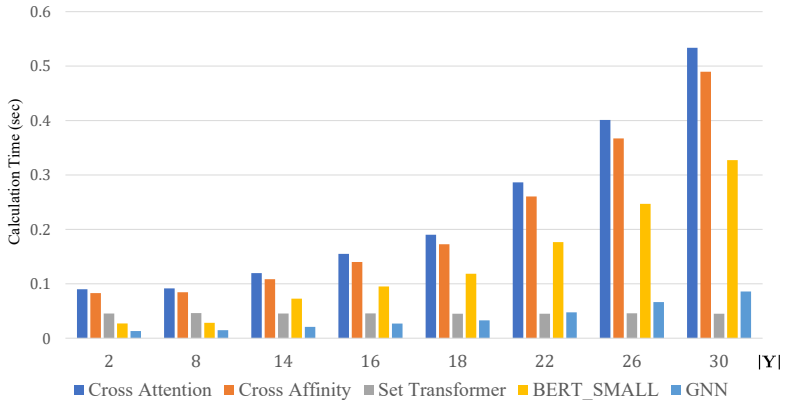


Figure 3.7. Inference time for set-to-set matching. Here, we test each model 110 times successively and plot the median in the last 100 records. We randomly generated pseudo data for the calculation, which are sets vectors on $\mathbb{R}^{512}$. Each set contains eight data. The number of candidates is two. We used GeForce GTX 970 for the calculation.

We demonstrated that our models performed well compared with the state-of-the-art methods and baselines, which were considered as extensions of promising models, in fashion set recommendation and group re-id experiments. Furthermore, we validated our proposed architecture through the ablation study. These results support the claim that the feature representations extracted with interactions between the set members of the two sets improve the accuracy and robustness of the heterogeneous set-to-set matching.

3.7 Proof and Discussion of Proposition 1

In this section, we aim to present the proof of Proposition 1, discussing our building blocks in detail, showing that our architecture is symmetric and permutation invariant function.

Symmetric Function. We show the proposed architecture is *symmetric* as follows. Assuming that $f : 2^{\mathfrak{X}} \times 2^{\mathfrak{X}} \to \mathbb{R}$ and $\hat{G} : \mathfrak{X}^* \times \mathfrak{X}^* \to \mathfrak{X}^* \times \mathfrak{X}^*$ are a symmetric function and feature extractor, respectively, in general, we can say function $\hat{G}$ does not preserve the symmetric property as follows:

$$f(\hat{G}(\mathcal{X}, \mathcal{Y})) \neq f(\hat{G}(\mathcal{Y}, \mathcal{X})). \tag{3.10}$$

Here, Eq. 3.10 shows an unsatisfied condition for the exchangeability. On the other hand, assuming that $G : \mathfrak{X}^* \times \mathfrak{X}^* \to \mathfrak{X}^* \times \mathfrak{X}^*$ is a *two-set-permutation equivariant*

function, we can construct a symmetric architecture as follows:

$$f(G(\mathcal{X},\mathcal{Y})) = f(G(\mathcal{Y},\mathcal{X})), \tag{3.11}$$

where Eq. 3.11 satisfies the property of exchangeability for two sets. Next, we consider splitting G into two *permutation equivariant functions* of $g : \mathfrak{X}^* \times \mathfrak{X}^* \rightarrow \mathfrak{X}^*$, where the range of g is the same shape of the first argument of g and g is also a feature extractor, which preserves the interactions between the two sets. The function G is then described as follows:

$$G(\mathcal{X},\mathcal{Y}) = (g(\mathcal{X},\mathcal{Y}|\mathbf{\Theta}_\mathrm{A}), g(\mathcal{Y},\mathcal{X}|\mathbf{\Theta}_\mathrm{B})), \tag{3.12}$$

where $\mathbf{\Theta}_\mathrm{A}$ and $\mathbf{\Theta}_\mathrm{B}$ are learnable weights. We can exchange $\mathcal{X}$ and $\mathcal{Y}$ as follows:

$$G(\mathcal{Y},\mathcal{X}) = (g(\mathcal{Y},\mathcal{X}|\mathbf{\Theta}_\mathrm{A}), g(\mathcal{X},\mathcal{Y}|\mathbf{\Theta}_\mathrm{B})). \tag{3.13}$$

Furthermore, using Eq. 3.11 and considering the fact that f is a symmetric function, we derive the following equation:

$$f(g(\mathcal{X},\mathcal{Y}|\mathbf{\Theta}_\mathrm{A}), g(\mathcal{Y},\mathcal{X}|\mathbf{\Theta}_\mathrm{B})) = f(g(\mathcal{X},\mathcal{Y}|\mathbf{\Theta}_\mathrm{B}), g(\mathcal{Y},\mathcal{X}|\mathbf{\Theta}_\mathrm{A})). \tag{3.14}$$

Here, Eq. 3.14 shows that $g(\mathcal{X},\mathcal{Y}|\mathbf{\Theta}_\mathrm{A}) = g(\mathcal{X},\mathcal{Y}|\mathbf{\Theta}_\mathrm{B})$ and $g(\mathcal{Y},\mathcal{X}|\mathbf{\Theta}_\mathrm{A}) = g(\mathcal{Y},\mathcal{X}|\mathbf{\Theta}_\mathrm{B})$ must be held, subject to $\mathbf{\Theta}_\mathrm{A} = \mathbf{\Theta}_\mathrm{B}$. Note that this is the weight sharing structure that our cross-set feature transformation must satisfy. Using the aforementioned weight sharing structure, we can say our architecture is symmetric in property.

Permutation Invariant Function. For the permutation invariance, we consider a composite function of a permutation equivariant function and permutation invariant function, which is a permutation invariant function in property. Because the feature extractor and matching layer in our architecture are permutation equivariant and permutation invariant function, respectively, we can say our architecture is permutation invariant.

3.8 More Details of Models

In the ablation study, we replace our mCS with max pooling, average pooling, projection metric (Huang, Wu, and Van Gool, 2018), covariance matrix (Wang et al.,

2012; Cai, Takala, and Pietikainen, 2010), set kernel (Kim et al., 2019), and cosine similarity metric (Nguyen and Bai, 2010). For projection metric, we use the inner product as described in (Huang, Wu, and Van Gool, 2018). For covariance matrix, we calculate two covariance matrices and the inner product between the two matrices (Zhu et al., 2013). Note that we also normalize the calculated similarities as described in (Nguyen and Bai, 2010). For set kernel, we use Gaussian kernel and multiple kernel learning as described in (Li et al., 2017).

Chapter 4

Conclusion

In this thesis, we advance multiple data matching via introducing two types of problem scenarios and propose novel matching models, including the recent promising frameworks: (i) matching clusters that commonly lie in heterogeneous data groups using kernel mean embeddings and (ii) matching sets via deep neural network models.

Through this thesis, we extend matching problems via modeling the structure of data using the powerful functions as follows:

- For the (i) cluster matching case, we have proposed the clustering method to maximize the similarity between the cluster structures within two domains based on the supervised information on the one-side domain. Here, the similarity is calculated via the similarity of kernel means, which represent the probability distributions of each cluster uniquely and nonparametrically.

- For the (ii) set matching case, we introduced the new deep neural network that learns matching up heterogeneous sets in feature spaces, which transform set-feature constrained on the two types of exchangeability required: exchangeability between the pair of sets and invariance across different permutations of the items in each set.

Furthermore, we have proposed the new applications as follows:

- For the (i) cluster matching case, we have proposed taxonomy matching via multiple data matching on asteroid and meteorite datasets. In the experiments, we investigated the links between the clusters of meteorites and asteroids.

- For the (ii) set matching case, we introduced set matching on fashion set recommendation and group re-identification. In the experiments, we have shown

that the proposed method provides significant improvements and results compared with the state-of-the-art methods in heterogeneous set matching applications.

4.1 Open Problems

We discuss some important problems for future research below.

Unsupervised Learning. Although we have proposed the models that use supervised information in this thesis, the unsupervised approach is also essential, and optimizing our method in an unsupervised manner is non-trivial. For example, unsupervised learning for our deep neural network may lead to unsupervised set-to-set matching that would be meaningful in some use-cases.

Rejection Scheme. Our methods must respond to select one of the answers from candidates in the matching process. In a real-world application, however, sometimes rejections are necessary, and unmatched cases are important (Iwata and Ishiguro, 2017).

Elimination of Spurious Correlation. Our models do not include an explicit scheme for eliminating spurious correlation; however, it is essential to discover matching that does not rely on such a spurious correlation. Specifically, for the (i) cluster matching case, the matching scenario requires scientific experiments towards investigating the long-standing hypothesis on meteorite and asteroid. Although our methodology includes feature pre-processing that reduces noises on spectra data, which decreases the spurious correlation, we further need to investigate the spurious relationship on the matching process.

Bibliography

Airoldi, Edoardo M. et al. (2008). "Mixed membership stochastic blockmodels". In: *Journal of machine learning research* 9.Sep, pp. 1981–2014.

Ando, Rie K. and Tong Zhang (2005). "A framework for learning predictive structures from multiple tasks and unlabeled data". In: *Journal of machine learning research* 6.Nov, pp. 1817–1853.

Arandjelovic, Ognjen et al. (2005). "Face recognition with image sets using manifold density divergence". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol. 1. IEEE, pp. 581–588.

Argyriou, Andreas et al. (2008). "A spectral regularization framework for multi-task structure learning". In: *Advances in neural information processing systems*, pp. 25–32.

Bai, Yunsheng et al. (2018). "Convolutional set matching for graph similarity". In: *arXiv preprint arXiv:1810.10866*.

Bai, Yunsheng et al. (2019). "Simgnn: A neural network approach to fast graph similarity computation". In: *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 384–392.

Barucci, Maria A. et al. (1987). "Classification of asteroids using G-mode analysis". In: *Icarus* 72.2, pp. 304–324.

Bello, Saifullahi A., Shangshu Yu, and Cheng Wang (2020). "Review: deep learning on 3D point clouds". In: *arXiv preprint arXiv:2001.06280*.

Blaschko, Matthew B. and Arthur Gretton (2008). *Taxonomy Inference Using Kernel Dependence Measures*. Tech. rep. 181. Max-Planck Institute for Biological Cybernetics, Tübingen, Germany.

Blaschko, Matthew B. and Christoph H. Lampert (2008). "Correlational spectral clustering". In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1–8.

Britt, Daniel T. et al. (1992). "Comparison of asteroid and meteorite spectra: Classification by principal component analysis". In: *Icarus* 99.1, pp. 153–166. ISSN: 0019-1035.

Buratti, Bonnie J. et al. (2013). "Vesta, vestoids, and the HED meteorites: Interconnections and differences based on Dawn Framing Camera observations". In: *Journal of Geophysical Research: Planets* 118.10, pp. 1991–2003. ISSN: 2169-9100.

Bus, Schelte J. and Richard P. Binzel (2002). "Phase II of the small main-belt asteroid spectroscopic survey: A feature-based taxonomy". In: *Icarus* 158.1, pp. 146–177.

Busarev, Vladimir V. (2012). "A Hypothesis on the Origin of C-Type Asteroids and Carbonaceous Chondrites". In: *Asteroids, Comets, Meteors 2012*. Vol. 1667. LPI Contributions, p. 6017.

Cai, Yinghao, Valtteri Takala, and Matti Pietikainen (2010). "Matching groups of people by covariance descriptor". In: *2010 20th International Conference on Pattern Recognition*. IEEE, pp. 2744–2747.

Caruana, Rich (1997). "Multitask learning". In: *Machine learning* 28.1, pp. 41–75.

Caspi, Yaron, Denis Simakov, and Michal Irani (2006). "Feature-based sequence-to-sequence matching". In: *International Journal of Computer Vision* 68.1, pp. 53–64.

Cevikalp, Hakan and Bill Triggs (2010). "Face recognition based on image sets". In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2567–2573.

Chang, Jonathan and David Blei (2009). "Relational topic models for document networks". In: *Artificial Intelligence and Statistics*, pp. 81–88.

Chapman, Clark R. (1996). "S-type asteroids, ordinary chondrites, and space weathering: The evidence from Galileo's fly-bys of Gaspra and Ida". In: *Meteoritics & Planetary Science* 31.6, pp. 699–725. ISSN: 1945-5100.

Chapman, Clark R., David Morrison, and Ben Zellner (1975). "Surface properties of asteroids: A synthesis of polarimetry, radiometry, and spectrophotometry". In: *Icarus* 25.1, pp. 104–130.

Chopra, Sumit, Raia Hadsell, and Yann LeCun (2005). "Learning a similarity metric discriminatively, with application to face verification". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 539–546.

Cloutis, Edward A., Richard P. Binzel, and Michael J. Gaffey (2014). "Establishing asteroid–meteorite links". In: *Elements* 10.1, pp. 25–30. ISSN: 1811-5209.

Cucurull, Guillem, Perouz Taslakian, and David Vazquez (2019). "Context-aware visual compatibility prediction". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12617–12626.

Dai, Wenyuan et al. (2007a). "Boosting for transfer learning". In: *Proceedings of the 24th international conference on Machine learning*, pp. 193–200.

Dai, Wenyuan et al. (2007b). "Transferring naive bayes classifiers for text classification". In: *Twenty-Second AAAI Conference on Artificial Intelligence*. Vol. 7, pp. 540–545.

Dai, Wenyuan et al. (2008). "Self-taught clustering". In: *Proceedings of the 25th international conference on Machine learning*, pp. 200–207.

DeMeo, Francesca E. et al. (2009). "An extension of the Bus asteroid taxonomy into the near-infrared". In: *Icarus* 202.1, pp. 160–180. ISSN: 0019-1035.

Devlin, Jacob et al. (2018). "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805*.

Eitz, Mathias et al. (2010). "Sketch-based image retrieval: Benchmark and bag-of-features descriptors". In: *IEEE transactions on visualization and computer graphics* 17.11, pp. 1624–1636.

Faivishevsky, Lev and Jacob Goldberger (2010). "Nonparametric information theoretic clustering algorithm". In: *Proceedings of the 27th International Conference on Machine Learning*, pp. 351–358.

Feng, Jie, Svebor Karaman, and Shih-fu Chang (2017). "Deep image set hashing". In: *2017 IEEE Winter Conference on Applications of Computer Vision*. IEEE, pp. 1241–1250.

Fey, Matthias et al. (2020). "Deep graph matching consensus". In: *arXiv preprint arXiv:2001.09621*.

Gaffey, Michael J. (1976). "Spectral reflectance characteristics of the meteorite classes". In: *Journal of Geophysical Research* 81.5, pp. 905–920.

Gaffey, Michael J., Thomas H. Burbine, and Richard P. Binzel (1993). "Asteroid spectroscopy: Progress and perspectives". In: *Meteoritics & Planetary Science* 28.2, pp. 161–187.

Gao, Sheng et al. (2013). "Cross-domain recommendation via cluster-level latent factor model". In: *Joint European conference on machine learning and knowledge discovery in databases*. Springer, pp. 161–176.

Gao, Zan et al. (2018). "Group-pair convolutional neural networks for multi-view based 3d object retrieval". In: *Thirty-Second AAAI Conference on Artificial Intelligence*.

Gionis, Aristides, Piotr Indyk, Rajeev Motwani, et al. (1999). "Similarity search in high dimensions via hashing". In: *Vldb*. Vol. 99. 6, pp. 518–529.

Glorot, Xavier, Antoine Bordes, and Yoshua Bengio (2011). "Deep sparse rectifier neural networks". In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 315–323.

Gretton, Arthur et al. (2005). "Measuring statistical dependence with Hilbert-Schmidt norms". In: *International conference on algorithmic learning theory*. Springer, pp. 63–77.

Gruen, Armin (1985). "Adaptive least squares correlation: a powerful image matching technique". In: *South African Journal of Photogrammetry, Remote Sensing and Cartography* 14.3, pp. 175–187.

Guo, Michelle et al. (2018). "Neural graph matching networks for fewshot 3d action recognition". In: *Proceedings of the European Conference on Computer Vision*, pp. 653–669.

Hadid, Abdenour and Matti Pietikainen (2004). "From still image to video-based face recognition: an experimental analysis". In: *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.* IEEE, pp. 813–818.

Hamm, Jihun and Daniel D. Lee (2008). "Grassmann discriminant analysis: a unifying view on subspace-based learning". In: *Proceedings of the 25th international conference on Machine learning*, pp. 376–383.

Han, Xintong et al. (2017). "Learning fashion compatibility with bidirectional lstms". In: *Proceedings of the 25th ACM international conference on Multimedia*. ACM, pp. 1078–1086.

Hardersen, Paul S., Vishnu Reddy, and Rachel Roberts (2015). "Vestoids, Part II: The basaltic nature and HED meteorite analogs for eight Vp-type asteroids and their associations with (4) Vesta". In: *The Astrophysical Journal Supplement Series* 221.1, p. 19.

Hardersen, Paul S. et al. (2006). "Near-infrared spectral observations and interpretations for S-asteroids 138 Tolosa, 306 Unitas, 346 Hermentaria, and 480 Hansa". In: *Icarus* 181.1, pp. 94–106.

Hardersen, Paul S. et al. (2011). "The M-/X-asteroid menagerie: Results of an NIR spectral survey of 45 main-belt asteroids". In: *Meteoritics & Planetary Science* 46.12, pp. 1910–1938.

Hardersen, Paul S. et al. (2014). "More chips off of Asteroid (4) Vesta: Characterization of eight Vestoids and their HED meteorite analogs". In: *Icarus* 242, pp. 269–282.

Hardersen, Paul S. et al. (2018). "Basalt or not? Near-infrared spectra, surface mineralogical estimates, and meteorite analogs for 33 Vp-type asteroids". In: *The Astronomical Journal* 156.1, p. 11.

Hayat, Munawar, Mohammed Bennamoun, and Senjian An (2014). "Learning non-linear reconstruction models for image set classification". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1907–1914.

He, Kaiming et al. (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

He, Ruining, Charles Packer, and Julian McAuley (2016). "Learning compatibility across categories for heterogeneous item recommendation". In: *2016 IEEE 16th International Conference on Data Mining*. IEEE, pp. 937–942.

Hermans, Alexander, Lucas Beyer, and Bastian Leibe (2017). "In defense of the triplet loss for person re-identification". In: *arXiv preprint arXiv:1703.07737*.

Hiroi, Takahiro et al. (1993). "Modeling of S-type asteroid spectra using primitive achondrites and iron meteorites". In: *Icarus* 102.1, pp. 107–116.

Howard, Andrew G. et al. (2017). "MobileNets: Efficient convolutional neural networks for mobile vision applications". In: *arXiv preprint arXiv:1704.04861*.

Hsiao, Wei-lin and Kristen Grauman (2018). "Creating capsule wardrobes from fashion images". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7161–7170.

Hu, Jie, Li Shen, and Gang Sun (2018). "Squeeze-and-excitation networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141.

Hu, Yiqun, Ajmal S. Mian, and Robyn Owens (2011). "Sparse approximated nearest points for image set classification". In: *2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 121–128.

Huang, Zhenyu et al. (2019a). "Multi-view spectral clustering network". In: *Proc. 28th Int. Joint Conf. Artif. Intell.* Pp. 2563–2569.

Huang, Zhiwu, Jiqing Wu, and Luc Van Gool (2018). "Building deep networks on Grassmann manifolds". In: *Thirty-Second AAAI Conference on Artificial Intelligence*.

Huang, Ziling et al. (2019b). "DoT-GNN: Domain-transferred graph neural network for group re-identification". In: *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 1888–1896.

Huang, Ziling et al. (2019). "Group re-identification via transferred representation and adaptive fusion". In: *2019 IEEE Fifth International Conference on Multimedia Big Data*, pp. 128–132.

Huang, Ziling et al. (2019). "Group re-identification via transferred single and couple representation learning". In: *arXiv preprint arXiv:1905.04854*.

Ilse, Maximilian, Jakub M. Tomczak, and Max Welling (2018). "Attention-based deep multiple instance learning". In: *arXiv preprint arXiv:1802.04712*.

Iwata, Tomoharu, Tsutomu Hirao, and Naonori Ueda (2017). "Topic models for unsupervised cluster matching". In: *IEEE Transactions on Knowledge and Data Engineering* 30.4, pp. 786–795.

Iwata, Tomoharu and Katsuhiko Ishiguro (2017). "Robust unsupervised cluster matching for network data". In: *Data Mining and Knowledge Discovery* 31.4, pp. 1132–1154.

Iwata, Tomoharu, James Robert Lloyd, and Zoubin Ghahramani (2015). "Unsupervised many-to-many object matching for relational data". In: *IEEE transactions on pattern analysis and machine intelligence* 38.3, pp. 607–617.

Iwata, Tomoharu et al. (2017). "Unsupervised group matching with application to cross-lingual topic matching without alignment information". In: *Data mining and knowledge discovery* 31.2, pp. 350–370.

Jain, Sarthak and Byron C. Wallace (2019). "Attention is not explanation". In: *arXiv preprint arXiv:1902.10186*.

Jiang, Lu et al. (2017). "MentorNet: Regularizing Very Deep Neural Networks on Corrupted Labels". In: *arXiv preprint arXiv:1712.05055*.

Jiang, Qing-yuan and Wu-jun Li (2017). "Deep cross-modal hashing". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3232–3240.

Jolliffe, Ian T. (1986). *Principal component analysis*. Springer Verlag.

Kim, Jungtaek et al. (2019). "Practical Bayesian optimization over sets". In: *arXiv preprint arXiv:1905.09780*.

Kim, Tae-kyun, Josef Kittler, and Roberto Cipolla (2007). "Discriminative learning and recognition of image set classes using canonical correlations". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.6, pp. 1005–1018.

Kimura, Manabu and Masashi Sugiyama (2011). "Dependence-maximization clustering with least-squares mutual information." In: *JACIII* 15.7, pp. 800–805.

Klare, Brendan and Anil K. Jain (2010). "Heterogeneous face recognition: Matching nir to visible light images". In: *2010 20th International Conference on Pattern Recognition*. IEEE, pp. 1513–1516.

Klare, Brendan, Zhifeng Li, and Anil K. Jain (2010). "Matching forensic sketches to mug shot photos". In: *IEEE transactions on pattern analysis and machine intelligence* 33.3, pp. 639–646.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton (2012). "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*, pp. 1097–1105.

Le, Duc-trong, Hady W. Lauw, and Yuan Fang (2019). "Correlation-sensitive next-basket recommendation". In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, pp. 2808–2814.

Le, Quoc and Tomas Mikolov (2014). "Distributed representations of sentences and documents". In: *International conference on machine learning*, pp. 1188–1196.

Lee, Juho et al. (2019). "Set transformer: A Framework for attention-based permutation-invariant neural networks". In: *International Conference on Machine Learning*, pp. 3744–3753.

Lei, Zhen and Stan Z. Li (2009). "Coupled spectral regression for matching heterogeneous faces". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1123–1128.

Li, Chun liang et al. (2017). "MMD GAN: Towards deeper understanding of moment matching network". In: *arXiv preprint arXiv:1705.08584*.

Li, Stan Z., Zhen Lei, and Meng Ao (2009). "The HFB face database for heterogeneous face biometrics research". In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, pp. 1–8.

Li, Xinchao, Martha Larson, and Alan Hanjalic (2015). "Pairwise geometric matching for large-scale object retrieval". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5153–5161.

Li, Yujia et al. (2019). "Graph matching networks for learning the similarity of graph structured objects". In: *arXiv preprint arXiv:1904.12787*.

Li, Yuncheng et al. (2016). "Mining fashion outfit composition using an end-to-end deep learning approach on set data". In: *arXiv preprint arXiv:1608.03016*.

Lin, Weiyao et al. (2019). "Group Reidentification with Multigrained Matching and Integration". In: *IEEE transactions on cybernetics*.

Lisanti, Giuseppe et al. (2017). "Group re-identification via unsupervised transfer of sparse features encoding". In: *arXiv preprint arXiv:1707.09173*.

Liu, Deyin et al. (2019a). "Exploring inter-instance relationships within the query set for robust image set matching". In: *Sensors* 19.22, p. 5051.

Liu, Sifei et al. (2012). "Heterogeneous face image matching using multi-scale features". In: *2012 5th IAPR International Conference on Biometrics*. IEEE, pp. 79–84.

Liu, Xiaofeng et al. (2019b). "Permutation-invariant Feature Restructuring for Correlation-aware Image Set-based Recognition". In: *arXiv preprint arXiv:1908.01174*.

Liu, Yu, Junjie Yan, and Wanli Ouyang (2017). "Quality aware network for set to set recognition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5790–5799.

Lu, Jiwen et al. (2015). "Multi-manifold deep metric learning for image set classification". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1137–1145.

Maas, Andrew L., Awni Y. Hannun, and Andrew Y. Ng (2013). "Rectifier nonlinearities improve neural network acoustic models". In: *Proc. icml*. Vol. 30, p. 3.

MacQueen, James (1967). "Some methods for classification and analysis of multivariate observations". In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA., pp. 281–297.

McCord, Thomas B., John B. Adams, and Torrence V. Johnson (1970). "Asteroid Vesta: Spectral reflectivity and compositional implications". In: *Science* 168.3938, pp. 1445–1447.

Moon, Hyeonjoon and P. Jonathon Phillips (2001). "Computational and performance aspects of PCA-based face-recognition algorithms". In: *Perception* 30.3, pp. 303–321.

Muandet, Krikamol et al. (2012). "Learning from distributions via support measure machines". In: *Advances in neural information processing systems*, pp. 10–18.

Muandet, Krikamol et al. (2017). "Kernel mean embedding of distributions: A review and beyond". In: *Foundations and Trends® in Machine Learning* 10.1-2, pp. 1–141.

Mudgal, Sidharth et al. (2018). "Deep learning for entity matching: A design space exploration". In: *Proceedings of the 2018 International Conference on Management of Data*, pp. 19–34.

Murphy, Ryan L. et al. (2018). "Janossy pooling: Learning deep permutation-invariant functions for variable-size inputs". In: *arXiv preprint arXiv:1811.01900*.

Nakamura, Takuma and Ryosuke Goto (2018). "Outfit generation and style extraction via bidirectional LSTM and autoencoder". In: *arXiv preprint arXiv:1807.03133*.

Nakamura, Tomoki et al. (2011). "Itokawa dust particles: A direct link between S-type asteroids and ordinary chondrites". In: *Science* 333.6046, pp. 1113–1116.

Nallapati, Ramesh M. et al. (2008). "Joint latent topic models for text and citations". In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 542–550.

Newell, Alejandro, Kaiyu Yang, and Jia Deng (2016). "Stacked hourglass networks for human pose estimation". In: *arXiv preprint arXiv:1603.06937*.

Ng, Andrew Y., Michael I. Jordan, and Yair Weiss (2002). "On spectral clustering: Analysis and an algorithm". In: *Advances in neural information processing systems*, pp. 849–856.

Nguyen, Hieu V. and Li Bai (2010). "Cosine similarity metric learning for face verification". In: *Asian conference on computer vision*. Springer, pp. 709–720.

Noguchi, Takaaki et al. (2011). "Incipient space weathering observed on the surface of Itokawa dust particles". In: *Science* 333.6046, pp. 1121–1125. ISSN: 0036-8075.

Pang, Liang et al. (2019). "SetRank: Learning a permutation-invariant ranking model for information retrieval". In: *arXiv preprint arXiv:1912.05891*.

Parkhi, Omkar M., Andrea Vedaldi, Andrew Zisserman, et al. (2015). "Deep face recognition." In: *BMVC*. Vol. 1, p. 6.

Peng, Jiangtao, Lefei Zhang, and Luoqing Li (2016). "Regularized set-to-set distance metric learning for hyperspectral image classification". In: *Pattern Recognition Letters* 83, pp. 143–151.

Pieters, Carlé M. and Takahiro Hiroi (2004). "RELAB (Reflectance Experiment Laboratory): A NASA multiuser spectroscopy facility, lunar planet". In: *Sci. Conf. XXXV*.

Pieters, Carlé M. et al. (2005). "Asteroid-meteorite links: The Vesta conundrum (s)". In: *Proceedings of the International Astronomical Union* 1.S229, pp. 273–288.

Póczos, Barnabás et al. (2012). "Support distribution machines". In:

Quadrianto, Novi, Le Song, and Alexander J. Smola (2009). "Kernelized sorting". In: *Advances in neural information processing systems*, pp. 1289–1296.

Raina, Rajat, Andrew Y. Ng, and Daphne Koller (2006). "Constructing informative priors using transfer learning". In: *Proceedings of the 23rd international conference on Machine learning*, pp. 713–720.

Rayner, John T. et al. (2003). "SpeX: A medium-resolution 0.8–5.5 micron spectrograph and imager for the NASA Infrared Telescope Facility". In: *Publications of the Astronomical Society of the Pacific* 115.805, p. 362.

Reinsch, Christian H. (1967). "Smoothing by spline functions". In: *Numerische mathematik* 10.3, pp. 177–183.

Rendle, Steffen, Christoph Freudenthaler, and Lars Schmidt-thieme (2010). "Factorizing personalized markov chains for next-basket recommendation". In: *Proceedings of the 19th international conference on World wide web*. ACM, pp. 811–820.

Ristani, Ergys et al. (2016). "Performance measures and a data set for multi-target, multi-camera tracking". In: *arXiv preprint arXiv:1609.01775*.

Russakovsky, Olga et al. (2015). "Imagenet large scale visual recognition challenge". In: *International journal of computer vision* 115.3, pp. 211–252.

Saito, Yuki et al. (2020). "Data-driven taxonomy matching of asteroid and meteorite". In: *Meteoritics & Planetary Science* 55.1, pp. 193–206.

Sannai, Akiyoshi, Yuuki Takai, and Matthieu Cordonnier (2019). "Universal approximations of permutation invariant/equivariant functions by deep neural networks". In: *arXiv preprint arXiv:1903.01939*.

Sarwar, Badrul Munir et al. (2001). "Item-based collaborative filtering recommendation algorithms." In: *Www* 1, pp. 285–295.

Schölkopf, Bernhard and Alexander J. Smola (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press.

Shakhnarovich, Gregory, John W. Fisher, and Trevor Darrell (2002). "Face recognition from long-term observations". In: *European Conference on Computer Vision*. Springer, pp. 851–865.

Si, Jianlou et al. (2018). "Dual attention matching network for context-aware feature sequence based person re-identification". In: *arXiv preprint arXiv:1803.09937*.

Simonyan, Karen and Andrew Zisserman (2014). "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556*.

Smola, Alexander et al. (2007). "A Hilbert space embedding for distributions". In: *International Conference on Algorithmic Learning Theory*. Springer, pp. 13–31.

Sogi, Naoya, Taku Nakayama, and Kazuhiro Fukui (2018). "A method based on convex cone model for image-set classification with cnn features". In: *2018 International Joint Conference on Neural Networks*. IEEE, pp. 1–8.

Sohn, Kihyuk (2016). "Improved deep metric learning with multi-class N-pair loss objective". In: *Advances in neural information processing systems*, pp. 1857–1865.

Solera, Francesco et al. (2016). "Tracking social groups within and across cameras". In: *IEEE Transactions on Circuits and Systems for Video Technology* 27.3, pp. 441–453.

Song, Le et al. (2007). "A dependence maximization view of clustering". In: *Proceedings of the 24th international conference on Machine learning*. ACM, pp. 815–822.

Sultani, Waqas, Chen Chen, and Mubarak Shah (2018). "Real-world anomaly detection in surveillance videos". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6479–6488.

Szegedy, Christian et al. (2016). "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826.

Tang, Xiaoou and Xiaogang Wang (2004). "Face sketch recognition". In: *IEEE Transactions on Circuits and Systems for video Technology* 14.1, pp. 50–57.

Thirion, Jean-philippe (1998). "Image matching as a diffusion process: an analogy with Maxwell's demons." In: *Medical Image Anal.* 2.3, pp. 243–260.

Tholen, David J. (1984). "Asteroid taxonomy from cluster analysis of photometry". PhD thesis.

Tolias, Giorgos, Yannis Avrithis, and Hervé Jégou (2016). "Image search with selective match kernels: aggregation across single and multiple images". In: *International Journal of Computer Vision* 116.3, pp. 247–261.

Torfi, Amirsina et al. (2017). "3d convolutional neural networks for cross audio-visual matching recognition". In: *IEEE Access* 5, pp. 22081–22091.

Van den Elsen, Petra A., Evert-Jan D. Pol, and Max A. Viergever (1993). "Medical image matching-a review with classification". In: *IEEE Engineering in Medicine and Biology Magazine* 12.1, pp. 26–39.

Vasileva, Mariya I. et al. (2018). "Learning type-aware embeddings for fashion compatibility". In: *arXiv preprint arXiv:1803.09196*.

Vaswani, Ashish et al. (2017). "Attention is all you need". In: *Advances in neural information processing systems*, pp. 5998–6008.

Venugopalan, Subhashini et al. (2015). "Sequence to sequence-video to text". In: *Proceedings of the IEEE international conference on computer vision*, pp. 4534–4542.

Vincent, Pascal and Yoshua Bengio (2002). "K-local hyperplane and convex distance nearest neighbor algorithms". In: *Advances in neural information processing systems*, pp. 985–992.

Vinyals, Oriol, Samy Bengio, and Manjunath Kudlur (2015). "Order matters: Sequence to sequence for sets". In: *arXiv preprint arXiv:1511.06391*.

Vollgraf, Roland (2019). "Learning set-equivariant functions with SWARM mappings". In: *arXiv preprint arXiv:1906.09400*.

Wagstaff, Edward et al. (2019). "On the limitations of representing functions on sets". In: *arXiv preprint arXiv:1901.09006*.

Wang, Jiang et al. (2014). "Learning fine-grained image similarity with deep ranking". In: *arXiv preprint arXiv:1404.4661*.

Wang, Pu, Carlotta Domeniconi, and Jian Hu (2008). "Using wikipedia for co-clustering based cross-domain text classification". In: *2008 Eighth IEEE international conference on Data Mining*. IEEE, pp. 1085–1090.

Wang, Ruiping et al. (2008). "Manifold-manifold distance with application to face recognition based on image set". In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1–8.

Wang, Ruiping et al. (2012). "Covariance discriminative learning: A natural and efficient approach to image set classification". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2496–2503.

Wang, Shen et al. (2019). "Heterogeneous graph matching networks for unknown malware detection." In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 3762–3770.

Wang, Xiaogang and Rui Zhao (2014). "Person re-identification: System design and evaluation overview". In: *Person Re-Identification*. Springer, pp. 351–370.

Weisberg, Michael K., Timothy J. McCoy, and Alexander N. Krot (2006). "Systematics and evaluation of meteorite classification". In: *Meteorites and the early solar system II* 19.

Xiao, Hao et al. (2018). "Group re-identification: Leveraging and integrating multi-grain information". In: *Proceedings of the 26th ACM International Conference on Multimedia*. MM '18. Seoul, Republic of Korea: ACM, pp. 192–200. ISBN: 978-1-4503-5665-7.

Xie, Weidi, Li Shen, and Andrew Zisserman (2018). "Comparator networks". In: *Proceedings of the European Conference on Computer Vision*, pp. 782–797.

Yamaguchi, Osamu, Kazuhiro Fukui, and Ken-ichi Maeda (1998). "Face recognition using temporal image sequence". In: *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*. IEEE, pp. 318–323.

Yanai, Keizo and Hideyasu Kojima (1995). "Catalog of the Antarctic meteorites : collected from December 1969 to December 1994, with special reference to those represented in the collections of the National Institute of Polar Research". In: *National government publication*.

Yang, Jun et al. (2007). "Evaluating bag-of-visual-words representations in scene classification". In: *Proceedings of the international workshop on Workshop on multimedia information retrieval*, pp. 197–206.

Yang, Meng et al. (2013). "Face recognition based on regularized nearest points between image sets". In: *2013 10th IEEE international conference and workshops on automatic face and gesture recognition*. IEEE, pp. 1–7.

Yang, Qiang et al. (2009). "Heterogeneous transfer learning for image clustering via the social web". In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th international joint Conference on Natural Language Processing of the AFNLP*. Association for Computational Linguistics, pp. 1–9.

Yang, Yang et al. (2015). "Entity matching across heterogeneous sources". In: *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1395–1404.

Yang, Zichao et al. (2016). "Hierarchical attention networks for document classification". In: *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 1480–1489.

Yarotsky, Dmitry (2018). "Universal approximations of invariant maps by neural networks". In: *arXiv preprint arXiv:1804.10306*.

Yi, Dong et al. (2007). "Face matching between near infrared and visible light images". In: *International Conference on Biometrics*. Springer, pp. 523–530.

Yoshida, Tomoki, Ichiro Takeuchi, and Masayuki Karasuyama (2019). "Learning Interpretable Metric between Graphs: Convex Formulation and Computation with Graph Mining". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '19. Anchorage, AK, USA: Association for Computing Machinery, 1026–1036. ISBN: 9781450362016.

Yu, Rui et al. (2018). "Hard-aware point-to-set deep metric for person re-identification". In: *Proceedings of the European Conference on Computer Vision*, pp. 188–204.

Zaheer, Manzil et al. (2017). "Deep sets". In: *arXiv preprint arXiv:1703.06114*.

Zanfir, Andrei and Cristian Sminchisescu (2018). "Deep learning of graph matching". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2684–2693.

Zhang, Yan, Jonathon Hare, and Adam Prügel-bennett (2019). "FSPool: Learning set representations with featurewise sort pooling". In: *arXiv preprint arXiv:1906.02795*.

Zhang, Ying et al. (2017). "Deep Mutual Learning". In: *arXiv preprint arXiv:1706.00384*.

Zheng, Liang et al. (2015). "Scalable person re-identification: A benchmark". In: *Proceedings of the IEEE International Conference on Computer Vision*.

Zheng, Wei-shi, Shaogang Gong, and Tao Xiang (2009). "Associating groups of people." In: *BMVC*. Vol. 2. 6.

Zhong, Yujie, Relja Arandjelović, and Andrew Zisserman (2018). "GhostVLAD for set-based face recognition". In: *Asian Conference on Computer Vision*. Springer, pp. 35–50.

Zhong, Zhun et al. (2017). "Random erasing data augmentation". In: *arXiv preprint arXiv:1708.04896*.

Zhou, Sanping et al. (2017a). "Large margin learning in set-to-set similarity comparison for person reidentification". In: *IEEE Transactions on Multimedia* 20.3, pp. 593–604.

Zhou, Sanping et al. (2017b). "Point to set similarity based deep feature learning for person re-identification". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3741–3750.

Zhu, Feng, Qi Chu, and Nenghai Yu (2016). "Consistent matching based on boosted salience channels for group re-identification". In: *2016 IEEE International Conference on Image Processing*. IEEE, pp. 4279–4283.

Zhu, Pengfei et al. (2013). "From point to set: Extend the learning of distance metrics". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2664–2671.

Zhu, Xiaoke et al. (2017). "Learning heterogeneous dictionary pair with feature projection matrix for pedestrian video retrieval via single query image". In: *Thirty-First AAAI Conference on Artificial Intelligence*.