Uncertainty-Aware Multi-View Representation Learning

Yu Geng, [1] **Zongbo Han,** [1] **Changqing Zhang,** [1,2*] **Qinghua Hu** [1,2]
[1] College of Intelligence and Computing, Tianjin University, Tianjin, China
[2] Tianjin Key Lab of Machine Learning, Tianjin, China
{gengyu, zhangchangqing, huqinghua}@tju.edu.cn, hanzb1997@gmail.com

Abstract

Learning from different data views by exploring the underlying complementary information among them can endow the representation with stronger expressive ability. However, high-dimensional features tend to contain noise, and furthermore, the quality of data usually varies for different samples (even for different views), i.e., one view may be informative for one sample but not the case for another. Therefore, it is quite challenging to integrate multi-view noisy data under unsupervised setting. Traditional multi-view methods either simply treat each view with equal importance or tune the weights of different views to fixed values, which are insufficient to capture the dynamic noise in multi-view data. In this work, we devise a novel unsupervised multi-view learning approach, termed as Dynamic Uncertainty-Aware Networks (DUA-Nets). Guided by the uncertainty of data estimated from the generation perspective, intrinsic information from multiple views is integrated to obtain noise-free representations. Under the help of uncertainty, DUA-Nets weigh each view of individual sample according to data quality so that the high-quality samples (or views) can be fully exploited while the effects from the noisy samples (or views) will be alleviated. Our model achieves superior performance in extensive experiments and shows the robustness to noisy data.

Introduction

In recent years, there is a growing interest in multi-view learning. Information in the real world is usually in different forms simultaneously. When watching videos, the optic nerve receives visual signal while the auditory nerve receives speech signal. These two different types of signals complete each other and provide more comprehensive information. Accordingly, conducting representation learning from multi-view data has the potential to improve data analysis tasks (Yang and Wang 2018; Baltrušaitis, Ahuja, and Morency 2018; Li, Yang, and Zhang 2018).

However, the relationship among multiple views is usually very complex. There are two well-known principles in multi-view learning, i.e., *consistency* and *complementary* (Li, Yang, and Zhang 2018; Zhang et al. 2020). Most existing methods mainly focus on the consistency of multiple views which assume that the correlations among views should be maximized (Kumar, Rai, and Daume 2011; Wang et al. 2015). While there is also complementary information that is vital to comprehensive representations. Therefore, some methods are proposed to explore the complete information of multi-view data (Zhang, Liu, and Fu 2019; Hu and Chen 2019). More importantly, different sources of data may contain different amounts of information and possible noise. For example, due to the various sensor qualities or environmental factors, the information of different observations varies from each other. The quality of data usually varies for different samples (even for different views), i.e., one view may be informative for one sample but not the same case for another. Above challenges make multi-view learning rather difficult. In the context of unsupervised representation learning, it is even more challenging due to the lack of label guidance.

In this work, we propose a novel algorithm termed Dynamic Uncertainty-Aware Networks (DUA-Nets) to address these issues. As shown in Fig. 1, we employ Reversal Networks (R-Nets) to integrate intrinsic information from different views into a unified representation. R-Nets reconstruct each view from a latent representation, and thus the latent representation can encode complete information from multiple views. Furthermore, we are devoted to modeling the quality of each sample-specific view. This is quite different from the straightforward ways which ignore the differences between views and samples (Andrew et al. 2013; Wang et al. 2015). Another common approach is assigning each view a fixed weight (Huang et al. 2019; Peng et al. 2019). Although it considers view differences and is more effective than equal weighting, it cannot be adaptive to the noise variation inherent in different samples. In this paper, we employ *uncertainty* to estimate the quality of data. Specifically, under the assumption that each observation is sampled from a Gaussian distribution, R-Nets are applied to generate the mean and variance of the distribution, where the variance determines the sharpness of Gaussian distribution, and thus can be interpreted as uncertainty. Modeling data uncertainty can adaptively balance different views for different samples, which results in superior and robust performance. Comprehensive experiments demonstrate the effectiveness of the proposed DUA-Nets. We further provide insightful analyses about the estimated uncertainty.

For clarification, the main contributions of this work are

*Corresponding author

summarized as:

1. We propose an unsupervised multi-view representation learning (UMRL) algorithm which can adaptively address samples with noisy views, and thus, it guarantees the intrinsic information of multiple views are encoded into the learned unified representation.
2. We propose a novel online evaluation strategy for data quality by using uncertainty modeling, where the uncertainty can guide multi-view integration and alleviate the effect of unbalanced qualities of different views.
3. We devise a collaborative learning mechanism which seamlessly conducts representation learning and uncertainty estimation in a unified framework so that they can improve each other adaptively.
4. We conduct extensive experiments to validate the effectiveness of the proposed algorithm. In addition, insightful analyses are provided to further investigate the estimated uncertainty.

Related Works

Multi-View Representation Learning. The core problem of multi-view learning is how to effectively explore the consistency and complementary information of different views. Plenty of research works focus on multi-view learning and have achieved great progress. The most representative methods are canonical correlation analysis (CCA) (Hotelling 1936) and its variants (Bach and Jordan 2002; Hardoon and Shawe-Taylor 2011; Andrew et al. 2013; Wang et al. 2015). CCA searches a shared embedding of two views through maximizing the correlation between them. To reduce the influence of noisy data, sparse CCA (Hardoon and Shawe-Taylor 2011) is proposed to learn sparse representations. Kernel CCA (Bach and Jordan 2002) extends CCA to nonlinear conditions, which is the case of most real-world multi-view data. Based on deep learning framework, deep CCA (Andrew et al. 2013) is more powerful to capture nonlinear relationships. Deep canonically correlated autoencoder (DCCAE) (Wang et al. 2015) combines deep CCA and autoencoder structure to learn compact representation. Different from CCA, some approaches (Zhao, Ding, and Fu 2017; Zhang et al. 2018) employ matrix factorization to obtain hierarchical representation from multi-view data with specific constraints. Multi-view dimensionality co-reduction (MDcR) (Zhang et al. 2017) applies the kernel matching to regularize the dependence across views. Self-representation is also introduced to better incorporate multi-view information (Li et al. 2019; Cao et al. 2015). Moreover, generative adversarial network is applied to handle missing view problem (Wang et al. 2018) or impose prior information (Tao et al. 2019). There is a major difference between above approaches and our work - all of them treat each view equally or assign a fixed weight. In contrast, our method considers sample-specific view quality, while the corresponding uncertainty guides a robust multi-view integration.

Data Uncertainty Learning. Quantifying uncertainty and making reasonable decisions are critical in real-world applications (Paté-Cornell 1996; Faber 2005; Der Kiureghian and Ditlevsen 2009). There are mainly two categories of uncertainty, *data uncertainty* (a.k.a., aleatoric uncertainty) and *model uncertainty* (a.k.a., epistemic uncertainty). Data uncertainty can capture the noise inherent in the observations while model uncertainty (typically in supervised learning) can reflect the prediction confidence (Kendall and Gal 2017). Recently, many research works investigated how to estimate uncertainty in deep learning (Blundell et al. 2015; Gal and Ghahramani 2016). With these techniques, many computer vision models obtain great improvement on robustness and interpretability. For example, uncertainty modeling is introduced in face recognition (Shi and Jain 2019; Chang et al. 2020), and object detection (Choi et al. 2019; Kraus and Dietmayer 2019). Some methods (Kendall and Gal 2017; Kendall, Gal, and Cipolla 2018) utilize probability model to capture data uncertainty and reduce the effect of noisy samples. Our method introduces data uncertainty into multi-view learning. With the help of uncertainty, the proposed model can automatically estimate the importances of different views for different samples. Superior performance indicates that incorporating data uncertainty in information integration is more suitable to real-world applications.

Proposed Model

Multi-view representation learning (MRL) focuses on learning a unified representation encoding intrinsic information of multiple views. Formally, given a multi-view dataset $\mathcal{X} = \{\mathbf{x}_i^{(1)}; ...; \mathbf{x}_i^{(V)}\}_{i=1}^N$ which has V different views of observation, the goal of multi-view representation learning is inferring a latent representation $\mathbf{h}$ for each sample. Unfortunately, quality of views usually varies for different samples. For example, in multi-sensor system, there may be corrupted sensors providing inaccurate measurement (high-uncertainty-view), and furthermore there may be samples obtained in unpromising conditions (high-uncertainty-sample). A reliable multi-view representation learning model should take these conditions into consideration. In this section, we will show how to learn reliable representations from multi-view data by capturing the data uncertainty.

Uncertainty-Aware Multi-View Integration

For real-world applications, data usually contains inevitable noise, which is one of the main challenge in representation learning. In order to model the underlying noise, we assume different observations are sampled from different Gaussian distributions, i.e., $\mathbf{x}_i^{(v)} \sim \mathcal{N}(\boldsymbol{\mu}_i^{(v)}, (\sigma_i^{(v)})^2)$. Accordingly, the observations are modeled as

$$\mathbf{x}_i^{(v)} = \boldsymbol{\mu}_i^{(v)} + \epsilon\sigma_i^{(v)}, \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{1}$$

where the mean variable $\boldsymbol{\mu}_i^{(v)}$ refers to sample identity, and $\sigma_i^{(v)}$ reflects the uncertainty of the observation in the v^{th} view.

Based on above assumption, we target on encoding intrinsic information from multiple views into a unified representation. Considering the unified representation as latent

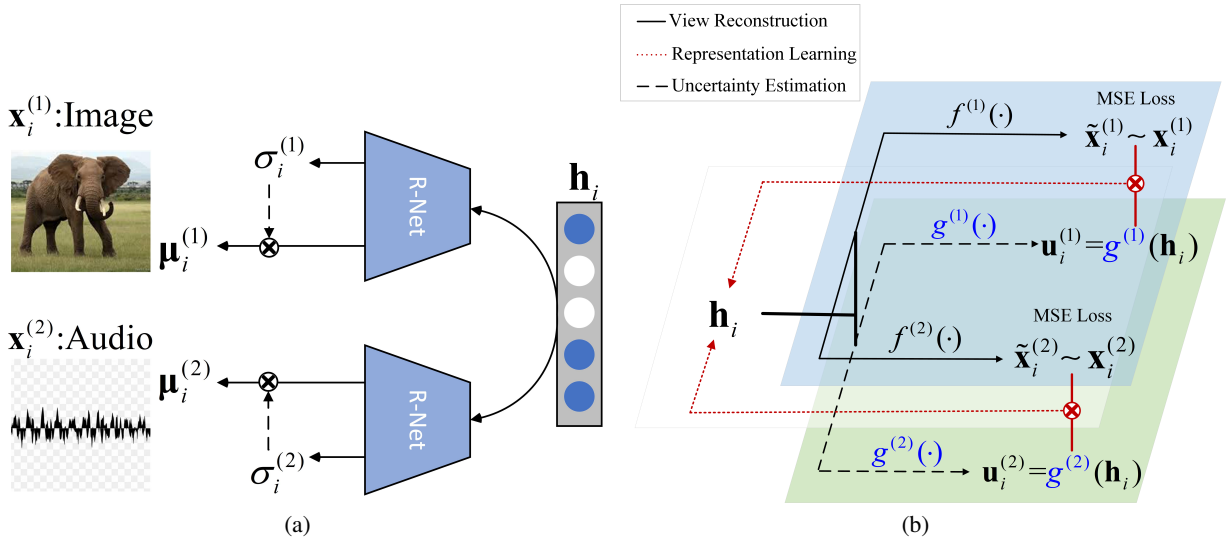


Figure 1: (a) Overview of the proposed DUA-Nets. (b) Learning process. We use two views for better elaboration. Latent variable $\mathbf{h}_i$ reconstructs each view through $f^{(v)}(\cdot)$. Simultaneously, $g^{(v)}(\cdot)$ estimates the uncertainty in the v^{th} view, which reflects the quality of view reconstruction. The learned uncertainty and reconstruction loss jointly guide the learning of $\mathbf{h}_i$.

variables, from the perspective of Bayesian, the joint distribution of latent variable $\mathbf{h}_i$ and multiple observations $\mathbf{x}_i^{(v)}$ for $v = 1, ..., V$ can be decomposed as prior on $\mathbf{h}_i$ ($p(\mathbf{h}_i)$) and likelihood as

$$p(\mathbf{x}_i^{(1)}, \cdots, \mathbf{x}_i^{(V)}, \mathbf{h}_i) = p(\mathbf{x}_i^{(1)}, \cdots, \mathbf{x}_i^{(V)}|\mathbf{h}_i)p(\mathbf{h}_i). \quad (2)$$

Since there is usually no prior knowledge on latent representation, we simply ignore the prior but focus on the likelihood. The likelihood aims to reconstruct observation of each view from the unified representation $\mathbf{h}_i$. The underlying assumption is that observation of each view $\mathbf{x}_i^{(v)}$ is conditionally independent given the latent variable $\mathbf{h}_i$, for which the likelihood can be factorized as

$$p(\mathbf{x}_i^{(1)}, \cdots, \mathbf{x}_i^{(V)}|\mathbf{h}_i) = p(\mathbf{x}_i^{(1)}|\mathbf{h}_i) \cdots p(\mathbf{x}_i^{(V)}|\mathbf{h}_i). \quad (3)$$

This implies that we can use multiple neural networks to decode the latent variable into different views. Taking one neural network $f^{(v)}(\cdot)$ for example, we use latent variable $\mathbf{h}_i$ to reconstruct the Gaussian distribution of observation $\mathbf{x}_i^{(v)}$, i.e.,

$$p(\mathbf{x}_i^{(v)}|\mathbf{h}_i) = \mathcal{N}(f^{(v)}(\mathbf{h}_i), (\sigma^{(v)})^2). \quad (4)$$

The parameters of neural networks are omitted for simplicity. To capture the uncertainty inherent in each observation instead of fixing it for each view, we model the variance to be variables which can vary with different samples. Then we have

$$p(\mathbf{x}_i^{(v)}|\mathbf{h}_i) = \mathcal{N}(f^{(v)}(\mathbf{h}_i), (g^{(v)}(\mathbf{h}_i))^2). \quad (5)$$

Rather than a deterministic output, now we model each observation containing different level of noise. Specifically, the learned variance is a metric that captures the uncertainty caused by noise.

Taking observation $\mathbf{x}_i^{(v)}$ as reconstruction target, it leads to the following likelihood

$$p(\mathbf{x}_i^{(v)}|\mathbf{h}_i) = \frac{1}{\sqrt{2\pi(\sigma_i^{(v)})^2}} \exp(-\frac{(\mathbf{x}_i^{(v)} - \boldsymbol{\mu}_i^{(v)})^2}{2(\sigma_i^{(v)})^2}) \quad (6)$$
$$s.t. \quad \boldsymbol{\mu}_i^{(v)} = f^{(v)}(\mathbf{h}_i), \ \sigma_i^{(v)} = g^{(v)}(\mathbf{h}_i).$$

In practice, the log likelihood as follows is maximized

$$\ln p(\mathbf{x}_i^{(v)}|\mathbf{h}_i) = -\frac{(\mathbf{x}_i^{(v)} - \boldsymbol{\mu}_i^{(v)})^2}{2(\sigma_i^{(v)})^2} - \ln(\sigma_i^{(v)}). \quad (7)$$

We omit the constant term because it will not affect the optimization. Then, we aim to search a positive scalar $\sigma_i^{(v)}$ for the v^{th} view of the i^{th} sample to weigh the reconstruction loss. The magnitude of variance determines the sharpness of Gaussian distribution. The larger the variance, the higher the uncertainty for the observation. Basically, large uncertainty can always reduce the reconstruction loss, but the second term introduced in the objective acts as a regularizer which constrains the uncertainty from increasing too much and avoids a trivial solution.

The reconstruction networks are utilized to enforce $\mathbf{h}_i$ to contain intrinsic information of multiple views (Fig. 1(a)), which makes it easier to infer $\mathbf{x}_i^{(v)}$. Through this reconstruction process, the latent variable $\mathbf{h}_i$ is optimized along with the parameters of networks. In this way, $\mathbf{h}_i$ is able to reconstruct each view, and thus information from different views can be well encoded into $\mathbf{h}_i$. Note that the flow of information in reconstruction networks is reverse to conventional neural network learning, where the input is observation and the output is latent representation. We term the decoder-like framework (i.e., reconstruction network) as Reversal Network (R-Net).

Accordingly, the final minimization objective of our multi-view model is

$$\mathcal{L} = \sum_{i=1}^{N}\sum_{v=1}^{V}\left(\frac{(\mathbf{x}_i^{(v)} - \boldsymbol{\mu}_i^{(v)})^2}{2(\sigma_i^{(v)})^2} + \ln(\sigma_i^{(v)})\right) \quad (8)$$
$$s.t. \quad \boldsymbol{\mu}_i^{(v)} = f^{(v)}(\mathbf{h}_i),\ \sigma_i^{(v)} = g^{(v)}(\mathbf{h}_i).$$

The overall proposed model is termed as Dynamic Uncertainty-Aware Networks (DUA-Nets). On the one hand, DUA-Nets estimate uncertainty in multi-view data. Instead of a fixed weight for each view, the model learns input-dependent uncertainty for different samples according to their quality. On the other hand, in DUA-Nets, the latent variable $\mathbf{h}_i$ acts as input and aims to reconstruct the original views in a reversal manner. The uncertainty of each view indicates the possible noise inherent in the observation, and thus it can guide the reconstruction process. With the help of uncertainty, the high-quality samples (and views) can be fully exploited while the effect from the noisy samples (and views) will be alleviated. In this way, we model the noise in multi-view data and reduce its impact to obtain robust representations. The learning process is shown in Fig. 1(b).

Why Can Our Model Capture Uncertainty without Supervision?

There may be a natural question: since most existing models estimate uncertainty with the help of class labels, how can we learn the uncertainty inherent in data without supervision? First, if given noise-free data, our model is able to promisingly reconstruct each observation. In this case, the estimated uncertainty is close to zero. While with noise in the data, the reconstruction loss will increase accordingly. The principle behind this is that the real-world data distributions have natural patterns that neural networks can easily capture. The noisy signals are usually high-frequency components that are difficult to model. Thus, it is difficult for neural networks to reconstruct noisy data, which causes large reconstruction loss on these samples. The assumption is consistent with prior study (Ulyanov, Vedaldi, and Lempitsky 2018). Therefore, when the low-quality data is as input, our model tends to output a larger reconstruction loss, and the corresponding uncertainty will be larger to prevent the reconstruction loss from increasing too much. Each R-Net is able to capture the data noise in each view, and further assigns different views with corresponding weights to produce a unified representation. We will show this effect in the experiments section (Fig. 3).

Experiments

In the experiments, we evaluate the proposed algorithm on real-world multi-view datasets and compare it with existing multi-view representation learning methods. The learned representation is evaluated by conducting clustering and classification tasks. Furthermore, we also provide the analysis of uncertainty estimation and robustness evaluation on noisy data.

Datasets

We conduct experiments on six real-world multi-view datasets as follows: **UCI-MF** (UCI Multiple Features)[1]: This dataset consists of handwritten numerals ('0'–'9') from a collection of Dutch utility maps. These digits are represented with six types of features. **ORL**[2]: ORL face dataset contains 10 different images of each of 40 distinct subjects under different conditions. Three types of features: intensity, LBP and Gabor are used as different views. **COIL20MV**[3]: There are 1440 images from 20 object categories. Three types of features that are same to ORL are used. **MSRCV1** (Xu, Han, and Nie 2016): This dataset contains 30 different images for each class out of 7 classes in total. Six types of features: CENT, CMT, GIST, HOG, LBP, SIFT are extracted. **CUB**[4]: Caltech-UCSD Birds dataset contains 200 different bird categories with 11788 images and text descriptions. Features of 10 categories are extracted by GoogLeNet and Doc2Vec in Gensim[5]. **Caltech101**[6]: This dataset contains images of 101 object categories. About 40 to 800 images per category. We use a subset of 1,474 images with 6 views.

Compared Methods

We compare the proposed DUA-Nets with multi-view representation learning methods as follows:

- **DCCA**: Deep Canonically Correlated Analysis (Andrew et al. 2013) extends CCA (Hotelling 1936) by applying deep neural networks to learn nonlinear projection. DCCA maximizes the correlation between learned representations of two views.
- **DCCAE**: Deep Canonically Correlated AutoEncoders (Wang et al. 2015) employs autoencoders structure to obtain better embedding.
- **MDcR**: Multi-view Dimensionality co-Reduction (Zhang et al. 2017) applies kernel matching constraint to enhance correlations among multiple views and combine these projected low-dimensional features together.
- **DMF-MVC**: Deep Semi-Non-negative Matrix Factorization for Multi-View Clustering (Zhao, Ding, and Fu 2017) uses deep neural networks to conduct semi-NMF on multi-view data and seek the consistent representation.
- **RMSL**: Reciprocal Multi-layer Subspace Learning (Li et al. 2019) uses self representation and reciprocal encoding to explore the consistency and complementary information among multiple views.

Implementation Details

There are two parts in the proposed DUA-Nets, view-specific reconstruction network and uncertainty estimation network. We employ similar network architecture for both

[1]https://archive.ics.uci.edu/ml/datasets/Multiple+Features
[2]http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html
[3]http://www.cs.columbia.edu/CAVE/software/softlib/
[4]http://www.vision.caltech.edu/visipedia/CUB-200.html
[5]https://radimrehurek.com/gensim/models/doc2vec.html
[6]http://www.vision.caltech.edu/Image_Datasets/Caltech101

dataset	metric	DCCA	DCCAE	MDcR	DMF-MVC	RMSL	R-Nets	DUA-Nets
UCI-MF	ACC	66.26 ± 0.16	69.17 ± 1.02	76.72 ± 2.77	71.86 ± 4.25	77.07 ± 5.36	94.64 ± 4.38	**96.50 ± 0.81**
	NMI	66.01 ± 0.45	66.96 ± 0.91	76.68 ± 0.93	73.09 ± 3.23	75.54 ± 3.09	91.70 ± 2.47	**93.04 ± 0.65**
	F	59.05 ± 0.39	60.50 ± 1.10	71.93 ± 2.22	66.66 ± 4.69	66.43 ± 6.77	92.01 ± 3.55	**93.75 ± 0.75**
	RI	91.39 ± 0.06	91.77 ± 0.21	94.11 ± 0.48	92.85 ± 1.13	92.08 ± 2.57	98.30 ± 0.89	**98.72 ± 0.17**
ORL	ACC	59.68 ± 2.04	59.40 ± 2.20	61.70 ± 2.19	65.38 ± 2.86	**76.95 ± 1.95**	68.85 ± 2.16	70.38 ± 1.25
	NMI	77.84 ± 0.83	77.52 ± 0.86	79.45 ± 1.20	82.87 ± 1.26	**91.29 ± 1.30**	84.05 ± 0.77	85.49 ± 0.76
	F	47.72 ± 2.05	46.71 ± 2.22	48.48 ± 2.59	52.01 ± 3.43	65.30 ± 2.17	66.72 ± 1.51	**69.32 ± 1.39**
	RI	97.42 ± 0.13	97.39 ± 0.14	97.28 ± 0.22	97.29 ± 0.30	97.96 ± 0.63	97.90 ± 0.14	**98.06 ± 0.09**
COIL20	ACC	63.73 ± 0.78	62.72 ± 1.40	64.25 ± 2.98	53.92 ± 5.89	63.04 ± 2.20	66.47 ± 6.73	**72.28 ± 4.79**
	NMI	76.02 ± 0.50	76.32 ± 0.66	79.44 ± 1.37	72.36 ± 2.11	79.24 ± 1.91	79.56 ± 2.23	**82.72 ± 1.81**
	F	58.76 ± 0.53	57.56 ± 1.15	63.60 ± 2.57	46.39 ± 4.97	61.05 ± 2.79	66.48 ± 4.22	**71.47 ± 3.35**
	RI	95.60 ± 0.06	95.27 ± 0.30	96.11 ± 0.29	92.56 ± 1.46	95.67 ± 0.61	96.03 ± 0.55	**96.77 ± 0.49**
MSRCV1	ACC	76.09 ± 0.08	65.43 ± 3.94	80.81 ± 2.13	32.19 ± 1.64	69.71 ± 1.25	81.52 ± 3.65	**84.67 ± 3.03**
	NMI	66.04 ± 0.09	60.75 ± 2.83	72.58 ± 1.94	17.32 ± 2.22	59.78 ± 0.91	76.77 ± 3.19	**77.26 ± 2.80**
	F	62.38 ± 0.12	55.27 ± 3.46	**80.83 ± 2.34**	23.45 ± 2.73	53.61 ± 1.22	75.10 ± 3.30	76.85 ± 3.17
	RI	89.62 ± 0.07	85.33 ± 1.63	91.05 ± 0.60	70.39 ± 2.19	86.08 ± 0.89	92.52 ± 0.99	**93.18 ± 0.99**
CUB	ACC	54.50 ± 0.29	66.70 ± 1.52	**73.68 ± 3.32**	37.50 ± 2.45	66.47 ± 4.58	70.53 ± 2.03	72.98 ± 2.97
	NMI	52.53 ± 0.19	65.76 ± 1.36	**74.49 ± 0.75**	37.82 ± 2.04	68.95 ± 4.28	71.63 ± 1.65	72.89 ± 1.59
	F	45.84 ± 0.31	58.22 ± 1.18	65.72 ± 1.37	28.95 ± 1.54	57.58 ± 6.93	64.79 ± 2.25	**66.33 ± 2.01**
	RI	88.61 ± 0.06	91.27 ± 0.24	92.75 ± 0.44	85.52 ± 0.26	89.76 ± 2.88	92.25 ± 0.50	**92.79 ± 0.42**
Caltech101	ACC	49.85 ± 6.93	53.93 ± 5.78	46.51 ± 0.67	52.75 ± 5.67	53.13 ± 9.63	50.86 ± 2.75	**54.55 ± 0.68**
	NMI	49.51 ± 5.18	53.94 ± 3.73	**56.43 ± 0.56**	45.52 ± 2.28	23.96 ± 9.03	52.31 ± 0.87	49.38 ± 0.91
	F	56.31 ± 8.44	57.57 ± 7.08	51.55 ± 0.56	55.67 ± 5.50	48.03 ± 3.04	57.62 ± 2.36	**58.37 ± 0.36**
	RI	72.79 ± 3.78	74.12 ± 3.27	73.27 ± 0.30	73.43 ± 2.33	57.77 ± 8.02	74.57 ± 0.63	**74.68 ± 0.16**

Table 1: Results of representation clustering performance.

components. For all datasets, 3-layer fully connected network followed by ReLU activation function is used in the experiments. The latent representation $\mathbf{h}_i$ is randomly initialized with Gaussian distribution. Adam optimizer (Kingma and Ba 2014) is employed for optimization of all parameters. The model is implemented by PyTorch on one NVIDIA Geforce GTX TITAN Xp with GPU of 12GB memory.

Performance Evaluation on Clustering

We apply DUA-Nets and compared methods to learn multi-view representations, then we conduct clustering task to evaluate these learned representations. We employ k-means algorithm because it is simple and intuitive that can well reflect the structure of representations. For quantitative comparison of these methods, we use four common evaluation metrics including accuracy (ACC), normalized mutual information (NMI), rand index (RI), F-score to comprehensively evaluate different properties of clustering results. For each of these metrics, a higher value indicates a better performance. In order to reduce the impact of randomness, we run each method for 30 times. Clustering results are shown in Table 1. We report the performance of our method without uncertainty (R-Nets) as an ablation comparison. It is observed that DUA-Nets achieve better performance on most datasets. Take UCI-MF for example, R-Nets improve 17% over RMSL. DUA-Nets further outperform R-Nets, which validates that our modeling of uncertainty can capture the noise inherent in data and further promote representation learning.

Performance Evaluation on Classification

We also conduct experiments on classification task based on the learned representations. KNN algorithm is used for its simplicity. We divide the learned representations into training and testing sets with different proportions, denoted as G_{ratio}/P_{ratio}, where G is "gallery set" and P is "probe set". Accuracy (ACC) is used as evaluation metric. We run 30 times for each partition and report the mean and standard deviation value as well. Classification results are shown in Table 2. DUA-Nets achieve more promising performance than compared methods. RMSL and MDcR perform better on some cases but DUA-Nets show more stable performance on all cases.

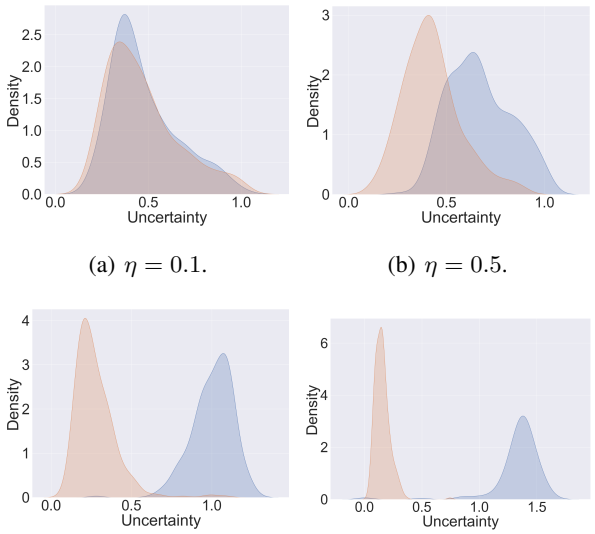
Uncertainty Estimation Analysis

According to the comparison between R-Nets and DUA-Nets, it seems the uncertainty is critical for the improvement. Therefore, in this part, we conduct qualitative experiments to provide some insights for the estimated uncertainty.

Ability of capturing uncertainty. We estimate data uncertainty on the modified CUB dataset. There are two views in the dataset, we add noise to half of the data in one view. Specifically, we generate $\frac{N}{2}$ noise vectors that are sampled from Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Then, we add these noise vectors (denoted as $\boldsymbol{\epsilon}$) multiplied with intensity η to pollute half of the original data, i.e., $\tilde{\mathbf{x}}_i^{(1)} = \mathbf{x}_i^{(1)} + \eta\boldsymbol{\epsilon}_i$, for $i = 1, ..., \frac{N}{2}$. The Gaussian kernel density estimation (Scott 2015) of learned uncertainty is shown in Fig. 2. It can be found that when the noise intensity is small ($\eta = 0.1$), the distribution curves of noisy samples and clean samples are

dataset	proportion	DCCA	DCCAE	MDcR	DMF-MVC	RMSL	R-Nets	DUA-Nets
UCI-MF	G_8/P_2	95.18 ± 0.55	95.78 ± 0.46	92.33 ± 0.73	94.68 ± 0.71	93.05 ± 1.17	96.78 ± 0.38	**98.10 ± 0.32**
	G_7/P_3	94.62 ± 0.64	95.10 ± 0.64	91.55 ± 0.39	93.72 ± 0.60	91.67 ± 1.14	96.55 ± 0.41	**97.98 ± 0.47**
	G_5/P_5	94.35 ± 0.46	94.79 ± 0.58	91.41 ± 0.68	93.33 ± 0.46	90.74 ± 1.32	95.95 ± 0.59	**97.72 ± 0.15**
	G_2/P_8	92.79 ± 0.51	92.63 ± 0.54	88.11 ± 0.61	88.23 ± 0.57	88.60 ± 0.78	93.34 ± 0.62	**96.44 ± 0.46**
ORL	G_8/P_2	83.25 ± 2.71	81.62 ± 2.95	92.00 ± 1.58	93.13 ± 1.21	**96.37 ± 1.38**	93.88 ± 2.05	94.14 ± 0.57
	G_7/P_3	78.92 ± 1.93	80.00 ± 1.47	90.83 ± 2.08	91.75 ± 1.64	**95.83 ± 1.47**	91.75 ± 1.51	92.94 ± 1.03
	G_5/P_5	71.15 ± 1.86	72.80 ± 2.04	83.35 ± 1.08	85.45 ± 1.85	**94.30 ± 1.64**	85.22 ± 1.62	86.36 ± 0.79
	G_2/P_8	51.69 ± 1.75	51.25 ± 1.90	57.38 ± 2.08	56.44 ± 2.50	**84.63 ± 1.14**	57.23 ± 1.79	59.56 ± 1.05
COIL20	G_8/P_2	90.96 ± 1.24	92.54 ± 0.70	91.11 ± 0.80	95.25 ± 1.06	93.14 ± 1.60	99.44 ± 0.09	**99.65 ± 0.27**
	G_7/P_3	90.48 ± 1.56	91.88 ± 1.44	90.29 ± 1.05	94.76 ± 0.77	91.79 ± 1.43	99.28 ± 0.41	**99.42 ± 0.43**
	G_5/P_5	88.65 ± 0.84	90.35 ± 0.58	87.63 ± 1.12	92.07 ± 0.61	90.32 ± 1.24	97.31 ± 0.52	**98.67 ± 0.23**
	G_2/P_8	83.35 ± 0.60	84.11 ± 1.10	79.46 ± 1.39	82.96 ± 1.03	85.65 ± 1.01	87.49 ± 0.99	**92.51 ± 0.44**
MSRCV1	G_8/P_2	79.52 ± 4.52	72.62 ± 4.38	**85.25 ± 2.21**	41.67 ± 4.52	79.52 ± 3.58	79.00 ± 2.24	82.40 ± 2.08
	G_7/P_3	76.90 ± 2.26	70.55 ± 5.67	**84.10 ± 3.17**	36.67 ± 4.16	78.25 ± 2.90	78.10 ± 3.16	81.75 ± 3.42
	G_5/P_5	65.05 ± 0.90	68.89 ± 2.77	79.86 ± 2.97	35.05 ± 2.27	77.90 ± 1.89	77.33 ± 2.65	**80.86 ± 2.43**
	G_2/P_8	43.69 ± 1.13	59.85 ± 3.40	72.55 ± 2.54	28.81 ± 1.55	71.69 ± 2.51	70.67 ± 1.14	**73.51 ± 3.33**
CUB	G_8/P_2	65.67 ± 2.85	77.00 ± 2.94	79.08 ± 3.43	60.08 ± 2.79	78.70 ± 2.50	75.73 ± 0.91	**80.25 ± 2.98**
	G_7/P_3	64.83 ± 1.83	74.56 ± 2.74	78.44 ± 3.08	58.56 ± 2.84	77.61 ± 1.38	74.11 ± 1.51	**79.67 ± 0.65**
	G_5/P_5	62.37 ± 1.58	72.60 ± 2.52	77.53 ± 1.67	55.30 ± 1.90	75.48 ± 1.57	72.48 ± 0.87	**77.87 ± 2.14**
	G_2/P_8	58.44 ± 2.92	67.35 ± 3.84	**74.58 ± 1.65**	49.60 ± 1.38	70.35 ± 1.95	62.77 ± 1.88	68.17 ± 1.44
Caltech101	G_8/P_2	92.12 ± 0.58	91.58 ± 1.02	90.14 ± 0.74	85.51 ± 1.05	40.71 ± 3.08	92.81 ± 0.66	**93.63 ± 0.58**
	G_7/P_3	91.46 ± 0.70	90.91 ± 0.75	89.45 ± 0.76	84.67 ± 0.82	39.76 ± 1.74	92.23 ± 0.42	**93.16 ± 0.45**
	G_5/P_5	91.30 ± 0.48	90.54 ± 0.44	88.95 ± 0.41	81.88 ± 0.73	37.14 ± 1.22	91.42 ± 0.21	**92.18 ± 0.52**
	G_2/P_8	88.73 ± 0.38	89.44 ± 0.43	88.46 ± 0.35	74.19 ± 0.99	33.82 ± 1.36	88.51 ± 0.48	**89.72 ± 0.77**

Table 2: Performance comparison on classification task.



(a) $\eta = 0.1$. (b) $\eta = 0.5$.

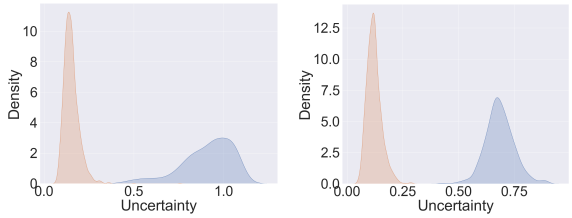
(c) $\eta = 1$. (d) $\eta = 2$.

Figure 2: Investigation of our model in capturing data noise. The curves in blue and orange correspond to distributions of noisy and clean data, respectively. The uncertainty basically becomes larger with the increasing of noise intensity.

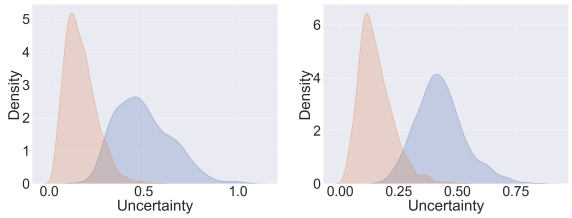
highly overlapped. With the increasing of noise intensity, the uncertainty of noisy samples grows correspondingly. This demonstrates that the estimated uncertainty is closely related to sample quality, which validates that the proposed DUA-Nets are aware of the quality of each observation, and guide the integration of views into promising representations.

Capture uncertainty without supervision. In order to investigate the principle behind uncertainty estimation in unsupervised manner, we synthetically add Gaussian noise to different ratio of samples. Typically, we conduct view-specific uncertainty estimation on first two views of UCI-MF dataset individually. Fig. 3 shows the Gaussian kernel density estimation of estimated uncertainty. DUA-Nets are able to capture uncertainty of observations. As the number of noisy samples become larger, the uncertainty distribution of data changes very slightly. This demonstrates that although the noise may significantly pollute data, the neural network is still able to identify the underlying pattern even under large ratio of noisy data. Specifically, the noise inherent in data increases the difficulty to reconstruction thus produces corresponding larger uncertainty. Accordingly, the uncertainty estimated by each R-Net (for each view) reasonably guides the integration of multi-view data.

Uncertainty in improving model performance. In Table 1 and Table 2, we show that the DUA-Nets are superior to R-Nets without uncertainty, which means that the learned uncertainty is beneficial for the representation learning. Here we conduct experiments to further verify the effect of uncertainty. We use UCI-MF dataset (with the first two views) and CUB dataset to conduct clustering task. In the experiment, view 1 of each dataset is selected to add Gaussian noise to



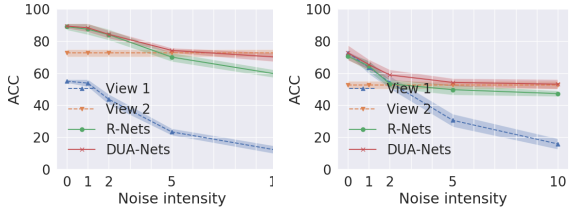
(a) View 1: Noise ratio 10%. (b) View 1: Noise ratio 50%.



(c) View 2: Noise ratio 10%. (d) View 2: Noise ratio 50%.

Figure 3: Investigation of "Why can our model capture uncertainty without supervision?" The curves in blue and orange correspond to distributions of noisy and clean data, respectively. The model is able to capture uncertainty even with large ratio of noisy data.

half of the samples. As shown in Fig. 4, with the increasing of the noise intensity, clustering performance on noisy view data decreases rapidly. However, the performance of DUA-Nets is quite stable. With the help of uncertainty, DUA-Nets are more robust to noise compared to R-Nets without uncertainty, which demonstrates that uncertainty can alleviate the influence of noisy observations.

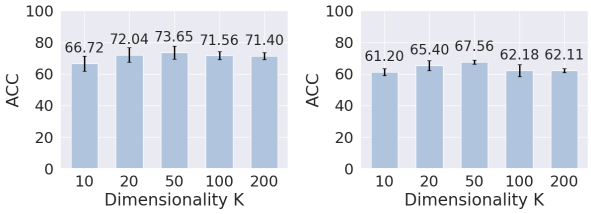


(a) UCI-MF. (b) CUB.

Figure 4: Uncertainty in improving model performance. With the increasing of noise intensity, DUA-Nets can achieve a robust performance.

Parameter Selection and Convergence

There is no explicit hyperparameter in our model, however, the dimension of latent representation needs to be specified in advance. In the experiments, we choose different dimensions of latent representation $\mathbf{h}_i$ to investigate its effect. We conduct clustering task on original CUB dataset as well as noisy CUB dataset (half of view 1 polluted by Gaussian noise). The dimensions are selected from [10, 20, 50, 100, 200]. As shown in Fig. 5, the best performance is obtained when the dimension is set to 50. As the dimension decreases too much, the latent representation may not have enough capacity to encode information from all views, which leads to a clear performance decline. Too larger dimensions also produce lower performance, where high-dimensional representation tends to overfit, and may contain possible noise in the final representation. Fig. 6 demonstrates the convergence of proposed method. Typically, the optimization process is basically stable, where the loss decreases quickly and converges within a number of iterations.



(a) CUB (original). (b) CUB (noise intensity=1) .

Figure 5: Parameter tuning. The performance of DUA-Nets with different dimensions for latent representation.

Figure 6: Convergence curve on CUB dataset (where loss values are normalized to range [0, 1]). The proposed method converges quickly within a small number of iterations.

Conclusions

In this work, we propose a novel multi-view representation learning method incorporating data uncertainty. Our model considers the noise inherent in observations and weighs different views of different samples dynamically. The estimated uncertainty provides guidance for multi-view integration, which leads to more robust and interpretable representations. Extensive experiments show the effectiveness and superiority of our model compared to deterministic methods. We further provide insights of the estimated uncertainty by qualitative analysis. In the future, we will focus on more theoretical results to explain and support the improvement.

Acknowledgments

This work was supported in part by National Natural Science Foundation of China (No. 61976151, No. 61732011 and No. 61876127), the Natural Science Foundation of Tianjin of China (No. 19JCYBJC15200).

References

Andrew, G.; Arora, R.; Bilmes, J.; and Livescu, K. 2013. Deep canonical correlation analysis. In *ICML*, 1247–1255.

Bach, F. R.; and Jordan, M. I. 2002. Kernel independent component analysis. *Journal of machine learning research* 3(Jul): 1–48.

Baltrušaitis, T.; Ahuja, C.; and Morency, L.-P. 2018. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence* 41(2): 423–443.

Blundell, C.; Cornebise, J.; Kavukcuoglu, K.; and Wierstra, D. 2015. Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424* .

Cao, X.; Zhang, C.; Fu, H.; Liu, S.; and Zhang, H. 2015. Diversity-induced multi-view subspace clustering. In *CVPR*, 586–594.

Chang, J.; Lan, Z.; Cheng, C.; and Wei, Y. 2020. Data Uncertainty Learning in Face Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5710–5719.

Choi, J.; Chun, D.; Kim, H.; and Lee, H.-J. 2019. Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, 502–511.

Der Kiureghian, A.; and Ditlevsen, O. 2009. Aleatory or epistemic? Does it matter? *Structural safety* 31(2): 105–112.

Faber, M. H. 2005. On the treatment of uncertainties and probabilities in engineering decision analysis. *Journal of Offshore Mechanics and Arctic Engineering* 127(3): 243–248.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, 1050–1059.

Hardoon, D. R.; and Shawe-Taylor, J. 2011. Sparse canonical correlation analysis. *Machine Learning* 83(3): 331–353.

Hotelling, H. 1936. Relations between two sets of variates. *Biometrika* 28(3/4): 321–377.

Hu, M.; and Chen, S. 2019. Doubly aligned incomplete multi-view clustering. *arXiv preprint arXiv:1903.02785* .

Huang, S.; Kang, Z.; Tsang, I. W.; and Xu, Z. 2019. Auto-weighted multi-view clustering via kernelized graph learning. *Pattern Recognition* 88: 174–184.

Kendall, A.; and Gal, Y. 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, 5574–5584.

Kendall, A.; Gal, Y.; and Cipolla, R. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7482–7491.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Kraus, F.; and Dietmayer, K. 2019. Uncertainty estimation in one-stage object detection. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 53–60. IEEE.

Kumar, A.; Rai, P.; and Daume, H. 2011. Co-regularized multi-view spectral clustering. In *NIPS*, 1413–1421.

Li, R.; Zhang, C.; Fu, H.; Peng, X.; Zhou, T.; and Hu, Q. 2019. Reciprocal Multi-Layer Subspace Learning for Multi-View Clustering. In *Proceedings of the IEEE International Conference on Computer Vision*, 8172–8180.

Li, Y.; Yang, M.; and Zhang, Z. 2018. A survey of multi-view representation learning. *IEEE transactions on knowledge and data engineering* 31(10): 1863–1883.

Paté-Cornell, M. E. 1996. Uncertainties in risk analysis: Six levels of treatment. *Reliability Engineering & System Safety* 54(2-3): 95–111.

Peng, X.; Huang, Z.; Lv, J.; Zhu, H.; and Zhou, J. T. 2019. COMIC: Multi-view clustering without parameter selection. In *International Conference on Machine Learning*, 5092–5101.

Scott, D. W. 2015. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons.

Shi, Y.; and Jain, A. K. 2019. Probabilistic face embeddings. In *Proceedings of the IEEE International Conference on Computer Vision*, 6902–6911.

Tao, Z.; Liu, H.; Li, J.; Wang, Z.; and Fu, Y. 2019. Adversarial Graph Embedding for Ensemble Clustering. In *IJCAI*, 3562–3568.

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V. 2018. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9446–9454.

Wang, Q.; Ding, Z.; Tao, Z.; Gao, Q.; and Fu, Y. 2018. Partial multi-view clustering via consistent GAN. In *2018 IEEE International Conference on Data Mining (ICDM)*, 1290–1295. IEEE.

Wang, W.; Arora, R.; Livescu, K.; and Bilmes, J. 2015. On Deep Multi-View Representation Learning 1083–1092.

Xu, J.; Han, J.; and Nie, F. 2016. Discriminatively embedded k-means for multi-view clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5356–5364.

Yang, Y.; and Wang, H. 2018. Multi-view clustering: A survey. *Big Data Mining and Analytics* 1(2): 83–107.

Zhang, C.; Cui, Y.; Han, Z.; Zhou, J. T.; Fu, H.; and Hu, Q. 2020. Deep Partial Multi-View Learning. *IEEE transactions on pattern analysis and machine intelligence* .

Zhang, C.; Fu, H.; Hu, Q.; Zhu, P.; and Cao, X. 2017. Flexible multi-view dimensionality co-reduction. *IEEE Transactions on Image Processing* 26(2): 648–659.

Zhang, C.; Liu, Y.; and Fu, H. 2019. Ae2-nets: Autoencoder in autoencoder networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2577–2585.

Zhang, Z.; Liu, L.; Shen, F.; Shen, H. T.; and Shao, L. 2018. Binary multi-view clustering. *IEEE transactions on pattern analysis and machine intelligence* 41(7): 1774–1782.

Zhao, H.; Ding, Z.; and Fu, Y. 2017. Multi-view clustering via deep matrix factorization. In *AAAI*, 2921–2927.