Deep Co-Clustering

Dongkuan Xu*‡ Wei Cheng†‡ Bo Zong† Jingchao Ni† Dongjin Song†
Wenchao Yu§ Yuncong Chen† Haifeng Chen† Xiang Zhang*

1 The Neural Network Structure of DeepCC on Different Datasets

The details of the structures are shown in Table 1. The number represents the neuron number on each layer. The deep autoencoder structure corresponds to the encoding part. The decoding part is the symmetrical one, which is skipped in this table. The inference networks for instances and features share the same structure. All layers are fully connected layers. The reason why the structures of the deep autoencoder for features are different for different datasets is that the feature dimensions of different datasets are very different.

2 Co-Clustering Result on Another Synthetic Data with More Noise

DeepCC performs on another synthetic dataset that is with more noise. The standard deviation of the gaussian noise for this dataset is 50. The two synthetic data used in this paper are with the same instances and features. Both are with 5 clusters for both instances and features. The synthetic data with more noise is shown in Fig 1(a). The data is shuffled as shown in Fig 1(b) and then fed into DeepCC. After co-clustered, the instances and features are rearranged to show co-clusters as shown in Fig 1(c), according to which we observe that DeepCC detects the co-clusters successfully. It should be noted that the detected blocks (co-clusters) in Fig 1(c) might not be in the same sequence of in Fig 1(a). This is because the cluster ID for the same instance or feature cluster might change, though the instances or features contained are the same.

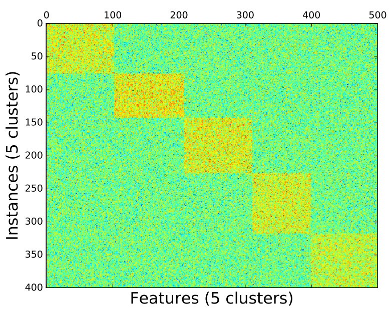
3 Hyperparameter Setting

The optimal hyperparameter settings are shown in Table 2. For the datasets belonging to the same category (images, texts or user ratings), their optimal hyperparameter setting is the same. This is because they share the similar data characteristic. For the image datasets (Coil20, Yale, Fashion-MNIST-test, Sign-MNIST-test), we search the optimal setting of λ_1 from $\{1\times10^{-2}$, 2×10^{-2}, 4×10^{-2}, 1×10^{-1}, 2×10^{-1}, $4\times10^{-1}\}$, the optimal setting of λ_2 from $\{1\times10^{-2}$, 2×10^{-2}, 4×10^{-2}, 1×10^{-1}, 2×10^{-1}, $4\times10^{-1}\}$, the optimal setting of λ_3 from $\{1\times10^{-2}$, 2×10^{-2}, 4×10^{-2}, 1×10^{-1}, 2×10^{-1}, $4\times10^{-1}\}$ and the optimal setting of λ_4 from $\{1\times10^{4}$, 2×10^{4}, 4×10^{4}, 1×10^{5}, 2×10^{5}, $4\times10^{5}\}$. For the text datasets (Citeseer, WebKB4 WebKB_cornell, WebKB_texas, WebKB_washington, WebKB_wisconsin), we search the optimal setting of λ_1 from $\{2^{-2}$, 2^{-1}, 2^{0}, 2^{1}, 2^{2}, $2^{3}\}$, the optimal setting of λ_2 from $\{2^{-2}$, 2^{-1}, 2^{0}, 2^{1}, 2^{2}, $2^{3}\}$, the optimal setting of λ_3 from $\{1\times10^{-5}$, 2×10^{-5}, 4×10^{-5}, 1×10^{-4}, 2×10^{-4}, $4\times10^{-4}\}$ and the optimal setting of λ_4 from $\{1\times10^{4}$, 2×10^{4}, 4×10^{4}, 1×10^{5}, 2×10^{5}, $4\times10^{5}\}$. For the user rating datasets, we search the optimal setting of λ_1 from $\{2^{-2}$, 2^{-1}, 2^{0}, 2^{1}, 2^{2}, $2^{3}\}$, the optimal setting of λ_2 from $\{1\times10^{-2}$, 2×10^{-2}, 4×10^{-2}, 1×10^{-1}, 2×10^{-1}, $4\times10^{-1}\}$, the optimal setting of λ_3 from $\{1\times10^{-5}$, 2×10^{-5}, 4×10^{-5}, 1×10^{-4}, 2×10^{-4}, $4\times10^{-4}\}$ and the optimal setting of λ_4 from $\{1\times10^{4}$, 2×10^{4}, 4×10^{4}, 1×10^{5}, 2×10^{5}, $4\times10^{5}\}$. The reason why the value of λ_1 for the text and the user rating datasets is much larger than the one of the image datasets is that the text and the user rating datasets are more sparse. So it needs a larger penalty on the reconstruction error of the deep autoencoder for the text and the user rating datsets, which corresponds to a larger value of λ_1.

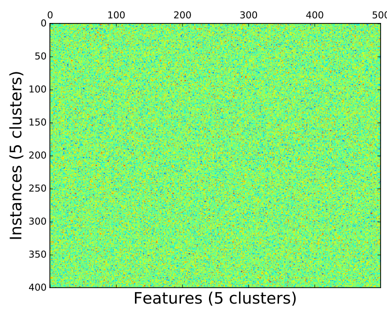
*The Pennsylvania State University. {dux19, xzz89}@psu.edu
†NEC Laboratories America, Inc. {weicheng, bzong, jni, dsong, yuncong, haifeng}@nec-labs.com
‡These authors contributed equally to this work.
§University of California Los Angeles. {yuwenchao}@ucla.edu

Table 1: The neural network structure of DeepCC on different datasets.

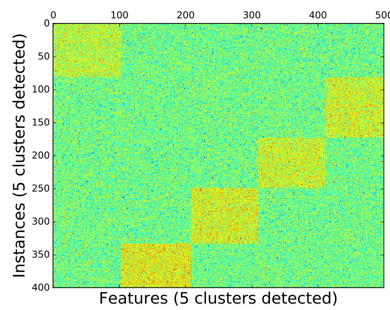
Dataset	Deep autoencoder for instances	Deep autoencoder for features	Inference network	Activation
Coil20	[1024×500×200×100×40]	[1440×500×200×100×40]	[40×160×80×40×20]	tanh
Yale	[1024×500×200×100×40]	[165×100×40]	[40×160×80×40×15]	tanh
Fashion-MNIST-test	[784×500×200×100×40]	[10000×1000×500×200×40]	[40×160×80×40×10]	tanh
Sign-MNIST-test	[784×500×200×100×40]	[7172×1000×500×200×40]	[40×160×80×40×25]	tanh
Citeseer	[3703×1000×500×200×40]	[3312×1000×500×200×40]	[40×160×80×40×6]	relu
WebKB4	[1000×500×200×100×40]	[4199×1000×500×200×40]	[40×160×80×40×4]	relu
WebKB_cornell	[1703×500×200×100×40]	[195×100×40]	[40×160×80×40×5]	relu
WebKB_texas	[1703×500×200×100×40]	[187×100×40]	[40×160×80×40×5]	relu
WebKB_washington	[1703×500×200×100×40]	[230×100×40]	[40×160×80×40×5]	relu
WebKB_wisconsin	[1703×500×200×100×40]	[265×100×40]	[40×160×80×40×5]	relu
IMDb_movies_keywords	[1878×500×200×100×40]	[617×200×100×40]	[40×160×80×40×17]	relu
IMDb_movies_actors	[1398×500×200×100×40]	[617×200×100×40]	[40×160×80×40×17]	relu



(a) Synthetic data.



(b) Data are shuffled.



(c) Co-clustering result.

Figure 1: DeepCC performs on the synthetic data with more noise.

Table 2: The optimal hyperparameter settings for different datasets.

Dataset	λ_1	λ_2	λ_3	λ_4
Coil20	2×10^{-2}	2×10^{-2}	1×10^{-1}	1×10^{5}
Yale	2×10^{-2}	2×10^{-2}	1×10^{-1}	1×10^{5}
Fashion-MNIST-test	2×10^{-2}	2×10^{-2}	1×10^{-1}	1×10^{5}
Sign-MNIST-test	2×10^{-2}	2×10^{-2}	1×10^{-1}	1×10^{5}
Citeseer	2^2	2^{-1}	2×10^{-4}	1×10^{5}
WebKB4	2^2	2^{-1}	2×10^{-4}	1×10^{5}
WebKB cornell	2^2	2^{-1}	2×10^{-4}	1×10^{5}
WebKB texas	2^2	2^{-1}	2×10^{-4}	1×10^{5}
WebKB washington	2^2	2^{-1}	2×10^{-4}	1×10^{5}
WebKB wisconsin	2^2	2^{-1}	2×10^{-4}	1×10^{5}
IMDb movies keywords	2^1	4×10^{-2}	1×10^{-4}	1×10^{5}
IMDb movies actors	2^1	4×10^{-2}	1×10^{-4}	1×10^{5}