User-Centered Feature Fusion

Marleny Hilasaca M.
Dalhousie University
Department of Community Health and Epidemiology
gm.hilasaca@dal.ca

1 Introduction

Multi-view data are common in real world applications. They are usually collected from diverse feature extractors, each providing complementary information. Leveraging them properly can substantially improve the performance compared with using only a single type of feature. The integration of multiple feature sets is known as feature fusion or Multi-view learning [6]. A naive solution for feature fusion considers concatenating all multiple features into one single feature vector or combining linearly a set of similarity metrics from individual features [5]. Another commonly used approach is weighting features automatically in order to improve an objective function associated with an evaluation metric (e.g. accuracy in classification tasks). However, when such function is not available, or there is a degree of subjectivity in the process, the definition of weights without proper user support hampers its applicability in practice. In that case, a feature fusion guided by the user's perception could be necessary. Some examples of subjective setting are music collections or photo albums, where user tastes differs greatly and different ways to capture users semantics and tastes are required. We propose the design of an efficient and effective method to generate an interpretable feature fusion defined by a user in real time. First, an initial sample of each feature is defined, and then they are mapped to a common space preserving the distance relationships of the individual feature. In this common space, we perform an alignment of all features to ensure consistency among views through a gradient descent approach. This sample is used to configure the initial feature fusion process. Then, this configuration is the input for a local affine transformation, which propagates the user semantic understanding to the whole data. Hence, we ensure that its user-defined knowledge and interpretability is preserved.

The main contributions of this work is: A novel feature fusion technique that allows users to explore and understand different combinations of features in real-time.

2 Proposed Methodology

Our approach for feature fusion employs a two phase strategy to support users on defining combinations that reflect a particular point-of-view regarding similarity relationships. On the first phase, samples $S_1, S_2, \ldots, S_h$ are extract from each different set of features $F_1, F_2, \ldots, F_h$ and merged so that each set S_i presents the same objects but represented using the different types of feature. Each sample S_i is then mapped to a vectorial representation $R_i \in \mathbb{R}^m$ preserving as much as possible the distance relationships between the instances. These vectorial representations are then combined to generate a single representation $\overline{R} = \alpha_1 R_1 + \alpha_2 R2 + \ldots + \alpha_h R_h$, which is visualized.

The user can then change the features weights and observe the outcome. Once the sample visualization reflects the user expectations, that is, once the proper weights $\alpha_1, \alpha_2, \ldots, \alpha_h$ are found, the second step takes place and the defined weights are used to combine the complete sets of features. In this process, the vectorial sample representations $R_1, R_2, \ldots, R_h$ and the samples $S_1, S_2, \ldots, S_h$ are used to construct models to map each set of feature F_i to a vectorial representation $V_i \in \mathbb{R}^m$. Since these vectorial representations are embedded in the same space, they can be combined using the weights $\alpha_1, \alpha_2, \ldots, \alpha_h$, obtaining the final vectorial representation $\overline{V}$ that matches the users

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

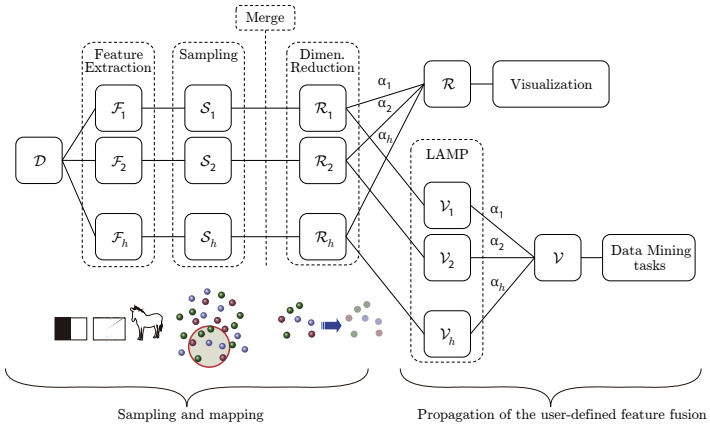

Figure 1: Overview of our process for feature fusion. Initially a sample is extracted, combined and visualized. Based on that, the user can test different weights to fuse the features and observe the outcome. Once sample combination reflects the user expectation, the same weights are used to combine the complete sets of features that can them be used on subsequent tasks, such as clustering.

expectations defined by the sample visualization. Figure 1 outlines our approach showing the involved steps. Next we detail these steps.

2.1 Sampling and Mapping

The first step of our process is sampling. Since users employ the sample visualization to guide the feature fusion process, it is important to have all possible data structures of the different features represented. In this process, we can extract samples from each set $F_1, F_2, \ldots, F_h$ separately using a cluster-based or random strategy. After extracting the sample sets $S_1, S_2, \ldots, S_h$, we merge their indexes defining a unified set of indexes. Then we recreate the sets $S_1, S_2, \ldots, S_h$ to have the instances with the indexes contained in the unified set of indexes. Therefore, all sample sets have the same instances, which is mandatory for the sample visualization given that we visualize the combination of all features $\overline{R}$. After recovering the samples, we map them to a common m-dimensional space, obtaining their vectorial representation $R_1, R_2, \ldots, R_h \in \mathbb{R}^m$ so that we can combine them to obtain $\overline{R} \in \mathbb{R}^m$ (for the sample visualization). In this process, each set of samples F_i is mapped to $\mathbb{R}^m$ preserving as much as possible the distance relationships in F_i. We do this by minimizing

$$E_{st}(F_i) = \frac{1}{|F_i|^2} \sum_{i}^{|F_i|} \sum_{j}^{|F_i|} \left(\delta(f_i^i, f_j^i) - ||r_i^i - r_j^i||\right)^2 \quad (1)$$

where f_i^i and f_j^i are instances in F_i, $\delta(f_i^i, f_j^i)$ is the distance between them, and r_i^i and r_j^i are the vectorial representations in the m-dimensional space of f_i^i and f_j^i, respectively.

2.2 Weighted Feature Combination

Given the samples vectorial representations $R_1, R_2, \ldots, R_h$, we build a set of functions using the process defined in [2] to map each feature set F_i into its vectorial representation $V_i \in \mathbb{R}^m$ preserving as much as possible the distance relationships while obeying the geometry define in R_i. In this process, each instance $f_j^i \in F_i$ is mapped to the m-dimensional space trough a orthogonal local affine transformation $T_j^i : \mathbb{R}^{q^i} \rightarrow \mathbb{R}^m$, where q^i is the dimensionality of F_i.

2.3 Feature Combination Widget

To visually support the feature sample combination, we create a widget. The idea is to position anchors (circles) representing each different set of features over a circumference, computing the weights $\alpha_1, \alpha_2, \ldots, \alpha_h$ according to their distances to a "dial" contained in the circumference. If $\tilde{f}_i$

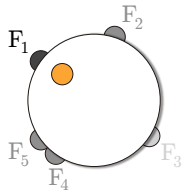

Figure 2: Feature Combination Widget. Using the orange "dial" users can control the contributions of the different types of features to the final feature combination.

are the coordinates of the anchor representing the feature F_i on the plane and $\tilde{d}$ the coordinates of the "dial", the weight α_i related to F_i is calculated as $\alpha_i = 1 \Big/ \left(\sum_j^h \frac{(1+\|\tilde{f}_i - \tilde{d}\|)^2}{(1+\|\tilde{f}_j - \tilde{d}\|)^2} \right)$

To help the perception of the weights, we change the transparency level of the anchors and fonts according to $\alpha_1, \alpha_2, \ldots, \alpha_h$. Furthermore, anchors can be moved together in case the users want to assign similar weights to a subset of features. Figure 2 shows our combination widget. In this Figure, the "dial", in orange, is closer to the anchor representing the feature F_1, so the corresponding anchor is darker than the other anchors. In the widget, anchors F_4 and F_5 are positioned close to each other, because users consider both features have the same importance.

3 Results

We evaluate our mapping and feature combination processes using different datasets in order to show that the sample manipulation effectively controls the complete feature fusion. These datasets come from a variety of different domains, including STL-10, Animals, Zappos, CIFAR-10, and Photographers. We use 4 distinct methods to extract features, representing low-level and highlevel image components. For the low-level features, we represent color with LAB color histogram, texture with Gabor filters, and shape with HoG technique. For the high-level, we extract deep-features from the pool5 layer using a pretrained CNN CaffeNet. For the feature combination, we assess the degree the distance relationships of the sample are preserved into the feature fusion of the whole dataset, intending to demonstrate the effectiveness of the user sample manipulation to the produced dataset. In this evaluation, we first generate 30 different weight combinations randomly summing up to 1 and apply it to sample data. Then, we reuse these weights for the whole data fusion and measure if the distance relationships induced by the weights on the sample are presented in the whole dataset. We use the Nearest Neighbor Measure (NNM) [1] in this analysis. NNM quantifies the similarity of each instance in the whole data with its nearest neighbor in the sampled data. NNM is given by $NNM = 1.0 - \frac{\sum_i^N D_i}{N}$, where D_i is the smallest distance among the i-th instance in the complete dataset and the instances in the sample, and N denotes the number of instances. The authors normalized each dimension of the data to the range $[0, 1]$. However, this results in the loss of the magnitude of the dimensions. So, we change the normalization per dimension by a unit vector normalization per instance to avoid such an effect. The output of NNM is in the interval $[0, 1]$ with larger values indicating better results.

We compare the NNM values of our feature fusion with two baselines: feature concatenation and distance fusion. Boxplots in Figure 3 show that our approach outperforms the other two baselines by at least 5%. The mean value for our method is 0.9365, and the baselines achieve 0.8877 and 0.8958, respectively. Hence, our method preserves more accurately the data distribution of the sample in the whole dataset fusion.

We also present an example based on projections for qualitative evaluation. The reasoning is to project the complete combined dataset ($\overline{V}$), showing that the patterns observed in the sample projection ($\overline{U}$) are preserved on the complete projection. In this example, we use our approach to explore large photo collections considering different user perspectives about similarity among images. We use the **photographers** dataset. Based on a sample and using our approach, users can combine different features by employing the combination widget (see Figure 2) until the sample visualization reflects a particular understanding regarding the similarity among photos. Figure 4 provides more importance to color and the objects contained in photos.

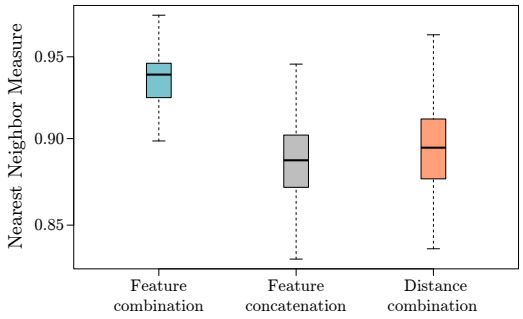

Figure 3: NNM evaluation. We compare our approach of user-guided feature fusion, with two baselines: feature concatenation, and feature combination trough distance measures.

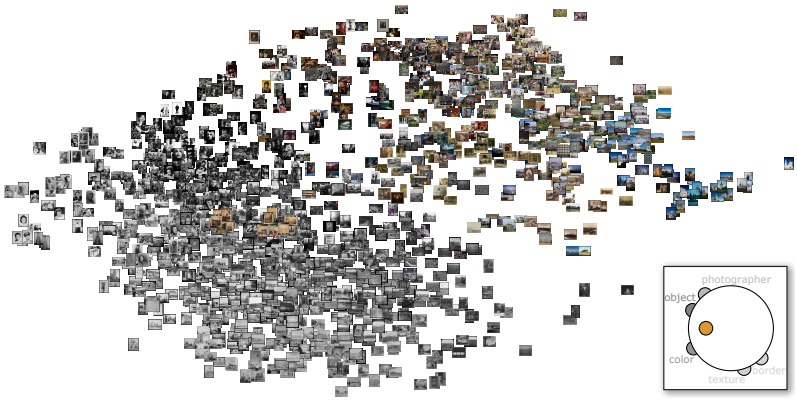

Figure 4: User-defined similarity configurations. Based on a small sample, users can interactively combine different features seeking for the combination that best approaches a particular point of view. This combination is then propagated to the entire dataset for a complete projection.

Once a feature combination has been defined that reflects the users' point of view, a projection representing the complete photo collection is constructed. Figure 5 shows the produced layout using the weights established in Figure 4. In this figure, since the color is an important feature, we observe a clear separation between black-and-white and colorful images. Also, given the weight assigned to the features representing objects, it is possible to notice a separation among photos of people, landscapes, and houses in certain regions of the figure. We zoom in on two small portions of the projection (at the top and at the right side) to show this effect. On the colored images (right), we observe images with sky and forest. On the gray images (top), we observe houses, sky, and forest.

3.1 Conclusion

In this work, we proposed a novel approach for feature fusion that successfully allows users to control the fusion process. It is a two-step strategy where, starting from a small sample of the input data, users can quickly test different feature combinations and check in real-time the resulting similarity relationships. Once a combination that matches the user expectation is defined, it is propagated to the whole dataset through an affine transformation. Our experiments show that the complete dataset combination preserves the similarities from the sample configuration, making our approach a very flexible mechanism to assist the feature fusion process.

References

[1] Q. Cui, M. Ward, E. Rundensteiner, and J. Yang. Measuring data abstraction quality in multiresolution visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):709–716, Sept 2006.

Figure 5: Photographers dataset projection using the weight combination of Figure 4. Since a larger weight is assigned to the color feature, a clear global separation between black-and-white and colorful photos can be observed.

[2] Paulo Joia, Danilo Coimbra, Jose A. Cuminato, Fernando V. Paulovich, and Luis G. Nonato. Local affine multidimensional projection. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2563–2571, December 2011.

[3] P. Liu, J. M. Guo, C. Y. Wu, and D. Cai. Fusion of deep learning and compressed domain features for content-based image retrieval. *IEEE Transactions on Image Processing*, 26(12):5706–5717, Dec 2017.

[4] Gang Ma, Xi Yang, Bo Zhang, and Zhongzhi Shi. Multi-feature fusion deep networks. *Neurocomput.*, 218(C):164–171, December 2016.

[5] Y. Wang, W. Zhang, L. Wu, X. Lin, and X. Zhao. Unsupervised metric fusion over multiview data by graph random walk-based cross-view diffusion. *IEEE Transactions on Neural Networks and Learning Systems*, 28(1):57–70, Jan 2017.

[6] Jing Zhao, Xijiong Xie, Xin Xu, and Shiliang Sun. Multi-view learning overview: Recent progress and new challenges. *Information Fusion*, 38:43 – 5, 2017.