Information Sciences 307 (2015) 39–52

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Learning similarity with cosine similarity ensemble

Peipei Xia [a], Li Zhang [a,b,*], Fanzhang Li [a]

[a] School of Computer Science and Technology & Provincial Key Laboratory for Computer Information Processing Technology, Soochow University, Suzhou 215006, Jiangsu, China
[b] Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing 210000, Jiangsu, China

ARTICLE INFO

Article history:
Received 22 July 2014
Received in revised form 9 February 2015
Accepted 14 February 2015
Available online 20 February 2015

Keywords:
Similarity learning
Cosine similarity
Ensemble learning
Selective ensemble
Machine learning

ABSTRACT

There is no doubt that similarity is a fundamental notion in the field of machine learning and pattern recognition. How to represent and measure similarity appropriately is a pursuit of many researchers. Many tasks, such as classification and clustering, can be accomplished perfectly when a similarity metric is well-defined. Cosine similarity is a widely used metric that is both simple and effective. This paper proposes a cosine similarity ensemble (CSE) method for learning similarity. In CSE, diversity is guaranteed by using multiple cosine similarity learners, each of which makes use of a different initial point to define the pattern vectors used in its similarity measures. The CSE method is not limited to measuring similarity using only pattern vectors that start at the origin. In addition, the thresholds of these separate cosine similarity learners are adaptively determined. The idea of using a selective ensemble is also implemented in CSE, and the proposed CSE method outperforms other compared methods on various data sets.

1. Introduction

Similarity is a fundamental issue in classification and clustering tasks. The concept of similarity is related to the concept of distance. However, the concepts of similarity and distance are not exactly the same. For example, similarity is used to measure the common characteristics between two instances, and distance is adopted to indicate the differences between them. Still, there is still a strong link between similarity and distance. We can first calculate the distance between two instances and then set an appropriate threshold to decide whether they are similar or not. Two instances will be more similar as the distance between them decreases.

How to select a well-defined distance metric is often a huge challenge due to the absence of prior knowledge. Many problems in pattern recognition can be easily solved if a similarity metric can be well estimated from the known data. The Euclidean distance is usually considered the simplest measure of similarity in many machine learning and data mining tasks. However, this metric often fails to generate discriminative representations. For this reason, even state-of-the-art algorithms, such as K-nearest neighbors (KNN) [8], support vector machines (SVM) [2,9,32] and artificial neural networks (ANN) [31,15,10] cannot achieve optimal performances. As a result, similarity learning, which attempts to learn similarity metrics adaptively for given tasks, has become an important research subject that has attracted considerable attention for the past decades.

Some similarity learning methods have been proposed that learn similarity functions directly from pairwise relationships, or constraints [27,24,21,6,20]. Phillips applied an SVM model to learn a similarity function in *difference space* [24], where the

[*] Corresponding author at: School of Computer Science and Technology & Provincial Key Laboratory for Computer Information Processing Technology, Soochow University, Suzhou 215006, Jiangsu, China.

E-mail address: zhangliml@suda.edu.cn (L. Zhang).

http://dx.doi.org/10.1016/j.ins.2015.02.024
0020-0255/© 2015 Elsevier Inc. All rights reserved.

distance between two patterns is measured by the difference between them. Melacci et al. proposed a novel neural network model, called a similarity neural network (SNN), to learn similarity in 2008 [21]. Kernel based methods also play an important role in this area of research. In these methods, the key point is to learn a feature mapping function and to then define an appropriate distance metric based on this mapping. Single kernel similarity learning methods were proposed in [6,20]. Tang et al. presented a new multikernel similarity learning method that outperforms these single kernel approaches [27]. For the multikernel method, a gradient descent algorithm is employed as a weak algorithm to generate the basic kernels. Because the gradient descent algorithm is relatively slow when the solution approaches the minimum [14], the multikernel method can be quite costly.

As mentioned before, the Euclidean distance is the most frequently used metric due to its simplicity. In the Euclidean space, the distance between two points is measured by the length of the line segment connecting them. Unfortunately, the Euclidean distance suffers from a high sensitivity to magnitudes. As an alternative, cosine similarity is another commonly used metric, which measures similarity as the angle between two vectors. For any two patterns, the patterns are considered less similar as the Euclidean distance between them increases, but they are considered more similar as the cosine similarity between them increases. The basic measure of cosine similarity is not sensitive to magnitudes. Unfortunately, this property is not always advantageous. For example, even two patterns with very different attribute values may have a very high similarity measure. This outcome is obviously undesirable. An adjusted cosine similarity metric [26] can remedy this drawback easily by taking the different scales between the two patterns into consideration and subtracting the corresponding average from each pattern.

Both the basic and adjusted cosine similarity metrics focus on orientations. However, the data distribution is often unknown in real world problems. When the patterns are in a very dense distribution, the angles between them may be very small. In such cases, even if two patterns are dissimilar, a classifier based on cosine similarity is very likely to misclassify them as similar. How can one make the angle between dissimilar patterns larger in such cases? In this paper, a novel similarity learning method, a cosine similarity ensemble (CSE), is proposed, that makes a trade off between computability and flexibility. Our CSE method enlarges the angles between patterns by changing the initial point of these patterns. Usually, the origin is specified as the initial point of a vector. When the terminal points of two given vectors are held constant, the angle between these two vectors is entirely determined by their shared initial point. In CSE, different points are chosen as initial points in the feature space and then combined according to weighting factors.

Section 2 surveys related work in similarity learning. Section 3 introduces the proposed method, CSE, in detail. Section 4 presents our experiments and their results, and Section 5 presents our conclusions.

2. Related work

We first give a formal description of the problem in similarity learning and then review some representative work that is related to ours.

2.1. Problem formulation

Suppose the input space $\mathcal{X}$ is a d-dimensional space. $\mathbf{x}$ and $\mathbf{x}'$ are two arbitrary patterns in $\mathcal{X}$, where $\mathbf{x} = [x_1, x_2, \ldots, x_d]^T$ and $\mathbf{x}' = [x'_1, x'_2, \ldots, x'_d]^T$. Let $(\mathbf{x}, \mathbf{x}')$ be the pairwise-patterns constructed by $\mathbf{x}$ and $\mathbf{x}'$. $l_{\mathbf{x}}$ and $l_{\mathbf{x}'}$ are class labels of $\mathbf{x}$ and $\mathbf{x}'$, respectively. $r(\mathbf{x}, \mathbf{x}') \in \{+1, -1\}$ indicates whether $\mathbf{x}$ and $\mathbf{x}'$ are similar to each other. If $\mathbf{x}$ and $\mathbf{x}'$ are similar, $r(\mathbf{x}, \mathbf{x}') = +1$; otherwise $r(\mathbf{x}, \mathbf{x}') = -1$. $r(\mathbf{x}, \mathbf{x}')$ can be described as:

$$r(\mathbf{x}, \mathbf{x}') = \begin{cases} +1, & \text{if } l_{\mathbf{x}} = l_{\mathbf{x}'}, \\ -1, & \text{otherwise}. \end{cases} \tag{1}$$

The intrinsic model of a similarity learning problem can be defined as a map, $(\mathbf{x}, \mathbf{x}') \mapsto \{+1, -1\}$. If $(\mathbf{x}, \mathbf{x}') \mapsto +1$, the model asserts that $\mathbf{x}$ and $\mathbf{x}'$ are similar. Otherwise, if $(\mathbf{x}, \mathbf{x}') \mapsto -1$, the model asserts that $\mathbf{x}$ and $\mathbf{x}'$ are dissimilar.

The objective of learning similarity is to develop a well-defined similarity metric which can fit the map well. Obviously, any arbitrary function fitting the map $(\mathbf{x}, \mathbf{x}') \mapsto \{+1, -1\}$ can be a similarity metric. However, which function is best among all possible similarity metrics? Let the best similarity metric be g. Ideally, $g(\mathbf{x}, \mathbf{x}') \equiv r(\mathbf{x}, \mathbf{x}')$ for any two arbitrary patterns $\mathbf{x}$ and $\mathbf{x}'$ in $\mathcal{X}$. Unfortunately, patterns with different labels may partially overlap in the input feature space. In addition, noise is unavoidable in real world problems. Hence, the best metric g that asserts similarity/dissimilarity correctly for all of the pairwise-patterns in a training set is likely to be an over-fitting one with poor generalization ability. In such case, the best similarity metric will not correctly indicate similarity/dissimilarity for unseen pairwise-patterns. For this reason, it is difficult to find the best similarity metric that also generalizes well to unseen data.

2.2. Representative work

As described in Section 1, many different distance metrics can be directly applied to the problem of similarity learning. The further the distance between a pair of patterns, the less similar these patterns are to one another.