Toward PAC-Learning of Weights from Qualitative Distance Information

Ken Satoh, Seishi Okamoto
Fujitsu Laboratories Limited
1015 Kamikodanaka, Nakahara-ku, Kawasaki 211, Japan
Email:ksatoh@flab.fujitsu.co.jp, seishi@flab.fujitsu.co.jp
Fax: +81-44-754-2664
Tel: +81-44-754-2661

Abstract

This paper discusses a mathematical analysis for learning weights in a similarity function. Although there are many works on theoretical analyses of case-based reasoning systems (Aha et al. 1991; Albert & Aha 1991; Langley & Iba 1993; Janke & Lange 1993), none has yet theoretically analyzed methods of producing a proper similarity function in accordance with a tendency of cases which many people have already proposed and empirically analyzed (Stanfill & Waltz 1986; Cardie 1993; Aha 1989; Callan et al. 1991).

In this paper, as the first step, we provide a PAC learning framework for weights with qualitative distance information. Qualitative distance information in this paper represents how a case is similar to another case. We give a mathematical analysis for learning weights from this information.

In this setting, we show that we can efficiently learn a weight which has an error rate less than ϵ with a probability more than $1-\delta$ such that the size of pairs in qualitative distance information is polynomilally bounded in the dimension, n, and the inverses of ϵ and δ, and the running time is polynomially bounded in the size of pairs.

Introduction

One of the most important mechanisms of CBR system is a similarity function. It, however, is usually defined in such a way that it is closely dependent on each considered domain. Thus, if we build a CBR system in a new domain, we have to consider this mechanism again by making various experiments to produce a plausible similarity function. We, therefore, should have some theoretical framework for this mechanism to reduce costs of finding a proper similarity function.

Some people have been investigating theoretical analyses of case-based reasoning systems (Aha et al. 1991; Albert & Aha 1991; Langley & Iba 1993; Janke & Lange 1993). (Aha et al. 1991) analyses the nearest neighbor algorithms in a framework similar to the PAC (probably approximately correct) learning (Valiant 1984) and (Albert & Aha 1991) generalizes the results in (Aha et al. 1991) to the k-nearest neighbor algorithms. (Langley & Iba 1993) analyses an average case behavior of the nearest neighbor algorithm, and (Janke & Lange 1993) compares case-based learning with inductive learning in the context of pattern language learnability. Although these theoretical results are important and give some insight for designing similarity functions, these are analyses for *fixed* similarity functions and none has yet proposed any theoretical analyses of producing a proper similarity function in accordance with a tendency of cases.

On the other hand, there are many empirical proposals of changing similarity functions based on stored cases (Stanfill & Waltz 1986; Cardie 1993) or results of classification (Aha 1989; Callan et al. 1991). (Stanfill & Waltz 1986) uses statistical information from the stored data to compute weights in the weighted feature metric which is more successful in the English pronunciation task than the simple nearest neighbor algorithm without weights. (Cardie 1993) selects relevant attributes by using a decision tree technique which are subsequently used in the k-nearest neighbor algorithm. These methods change metric by the tendency of stored cases. (Aha 1989) changes weights in weighted nearest neighbor in accordance with success or failure of classification and (Callan et al. 1991) adjusts a weight vector in a weighted distance function so that the distance to a near miss is made greater than the distance between the current case and the correct case. However, these are shown to be effective from *empirical* evaluations and we need theoretical analyses for these proposals to understand the behavior of these changeable similarity functions in general.

In this paper, as the first step, we provide a framework for learning weights from qualitative distance information. Qualitative distance information represents how a case is similar to another case. We give a mathematical analysis for learning weights from this information.

For example, suppose we have two attributes which take the real value. Each case is represented as a point in two dimensional Euclidean space. In Figure 1, we show points in the Euclidean space. We assume that we know that A is similar to B, and A is not similar to C without knowing why. This information can be represented as $NEAR(A,B)$ and $FAR(A,C)$ if we assume a proper distance function. If we use a usual (non-weighted) Euclidean distance function:

$$dist(A,B) = (A_{(x)} - B_{(x)})^2 + (A_{(y)} - B_{(y)})^2,$$

$dist(A,C)$ is lesser than $dist(A,B)$ and so the information of qualitative distance is inconsistent. However, if we shrink Y dimension to a half, that is, we use the following distance function:

$$dist(A,B) = (A_{(x)} - B_{(x)})^2 + \frac{1}{4}(A_{(y)} - B_{(y)})^2,$$

then the information becomes consistent (Figure 2). This distance function means that the importance of the attribute Y is a quarter of that of the attribute X. We would like to find such a proper transformation consistent with qualitative distance information. This corresponds with finding proper weights for attributes.

Learning weights in this manner is particularly important when no categorical information is available. For example, (Callan et al. 1991) considers the domain of the game OTHELLO and retrieves a similar state to the current state in the previous game plays. These states are not classified, and so, the learning methods of weights in (Stanfill & Waltz 1986; Cardie 1993) are not applicable. This kind of situation seems to arise in synthesis problem domains such as scheduling, design and planning.

By generalizing the above example, we consider two qualitative distance information $NEAR$ and FAR for n-dimensional Eucledian space which consist of pairs of points in the space. The intended meaning is that if a pair in $NEAR$, distance of the pair should be less than distance of any pair in FAR. Our problem is to find some weight vector W in $[0,\infty)^n$ and a positive constant D such that:

For $(A,B) \in NEAR$,

$$\sum_{i=1}^{n} W_{(i)}(A_{(i)} - B_{(i)})^2 \leq D,$$

and for $(A,B) \in FAR$,

$$\sum_{i=1}^{n} W_{(i)}(A_{(i)} - B_{(i)})^2 > D,$$

where $W_{(i)}$, $A_{(i)}$ and $B_{(i)}$ are i-th component of W, A and B respectively.

This corresponds with the problem to find a hyperoval in the n-dimensional Eucledian space such that

1. If $(A,B) \in NEAR$, then B is inside the oval whose center is A and vice versa.
2. If $(A,B) \in FAR$, then B is outside the oval whose center is A and vice versa.

In the above setting, we show that we can efficiently learn a weight W which has an error rate less than ϵ with a probability of at least $1-\delta$ such that the size of pairs in qualitative distance information is polynomially bounded in the dimension, n and the inverses of ϵ and δ, and the running time is polynomially bounded in the size of pairs.

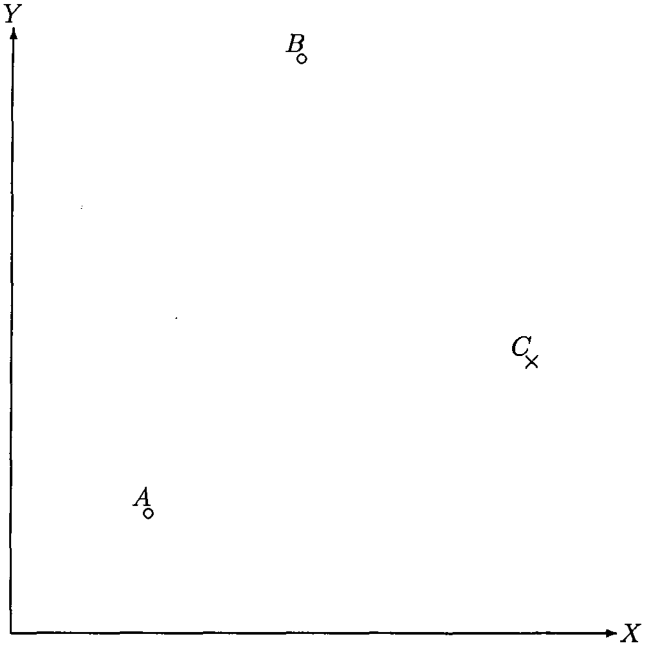

Figure 1: Original Space: inconsistent with the distance information $(A,B) \in NEAR$ and $(A,C) \in FAR$.

Formal Problem Description

Let $\mathbf{P}$ be any probability distribution over n-dimensional Euclidean space, E^n. The teacher selects a weight vector W^* from $[0,\infty)^n$ and a positive real number D^* which we call a *threshold*.

The teacher also selects N pairs of points in $E^n \times E^n$ according to $\mathbf{P}^2$ which denotes the two-fold product probability distribution on $E^n \times E^n$. Let a set of selected pairs be $X = \{(A_1,B_1),...,(A_N,B_N)\}$.

The teacher gives the following dichotomy of X, $NEAR$ and FAR which corresponds with the qualitative distance information:

if $dist_{W^*}(A,B) \leq D^*$ then $(A,B) \in NEAR$
if $dist_{W^*}(A,B) > D^*$ then $(A,B) \in FAR$

where $dist_{W^*}(A,B)$ is defined as:

$$\sum_{i=1}^{n} W^*_{(i)}(A_{(i)} - B_{(i)})^2.$$

The learning algorithm is required to approximate the vector W^* and D^* from the given qualitative distance information in a finite time. Let W be a vector in $[0,\infty)^n$ and D be a positive real number.

The difference between a set of pairs according to (W,D) and a set of pairs according to (W^*,D^*), $diff((W,D),(W^*,D^*))$ is defined as the union of

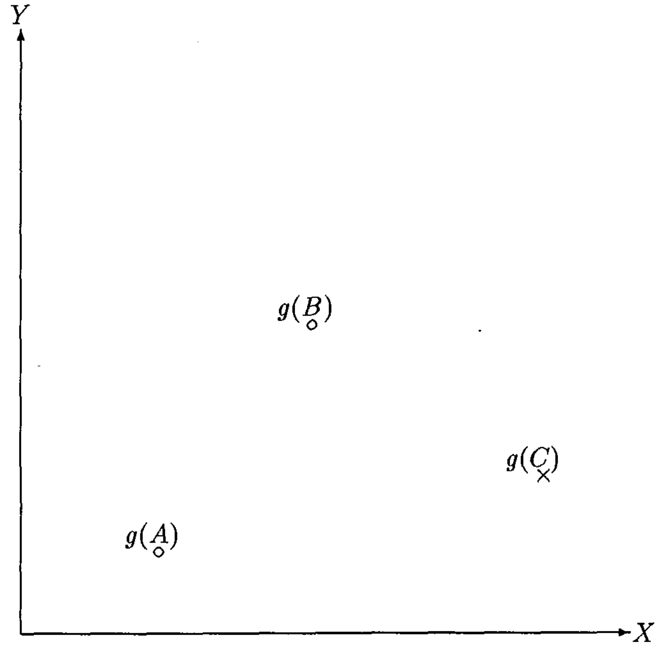

Figure 2: Transformed Space shrinked half in the Y-dimension: consistent with the distance information $(A,B) \in NEAR$ and $(A,C) \in FAR$.

$$\{(A,B) \in E^n \times E^n | \; dist_W(A,B) \leq D \text{ and } dist_{W^*}(A,B) > D^*\}$$

and

$$\{(A,B) \in E^n \times E^n | \; dist_W(A,B) > D \text{ and } dist_{W^*}(A,B) \leq D^*\}.$$

A pair (W,D) is said to be an *ϵ-approximation of (W^*,D^*) w.r.t. difference vectors for* $\mathbf{P}^2$ if the probability

$$\mathbf{P}^2(\mathit{diff}((W,D),(W^*,D^*)))$$

is at most ϵ.

The following theorem shows that this framework is PAC-learnable.

Theorem 1 *There exists a learning algorithm which satisfies the following conditions for all probability distribution over E^n, $\mathbf{P}$, and all constants ϵ and δ in the range $(0,1)$:*

1. *The teacher selects W^* and D^*.*
2. *The teacher gives N pairs according to $\mathbf{P}^2$ with qualitative distance information represented as two sets of pairs $NEAR$ and FAR according to W^* and D^* to the algorithm.*
3. *The algorithm outputs a weight vector W and a threshold D.*
4. *The probability that (W,D) is not an ϵ-approximation of (W^*,D^*) w.r.t. difference vectors for $\mathbf{P}^2$ is less than δ.*
5. *The size of required pairs N is bounded by a polynomial in n, $\frac{1}{\epsilon}$, $\frac{1}{\delta}$ and its running time is bounded by a polynomial in the size of required pairs.*

Proof: Let w be a vector in $[0,\infty)^n$ and d be a positive real number and $\mathbf{P}'$ be a probability distribution over $[0,\infty)^n$. We say that (w,d) is an ϵ-approximation of (w^*,d^*) w.r.t. points for $\mathbf{P}'$ if

$$\mathbf{P}'(\{x \in [0,\infty)^n | w \cdot x \leq d \text{ and } w^* \cdot x > d^*\} \cup \{x \in [0,\infty)^n | w \cdot x > d \text{ and } w^* \cdot x \leq d^*\}) \leq \epsilon$$

where $\cdot$ is the inner product of vectors.

According to the result in (Blumer et al. 1989) for learning half-spaces separated by a hyperplane, there exists a learning algorithm which satisfies the following conditions for every distribution $\mathbf{P}'$ on $[0,\infty)^n$ and every ϵ and δ in the range of $(0,1)$,

1. The teacher selects w^* in $[0,\infty)^n$ and a positive real number d^*.
2. The teacher gives a set X of N points according to $\mathbf{P}'$ with dichotomy (X^+,X^-) of X defined below:
   $$\text{for every } x \in X^+, w^* \cdot x \leq d^*, \text{ and}$$
   $$\text{for every } x \in X^-, w^* \cdot x > d^*.$$
3. The algorithm outputs a vector w and a positive real number d such that the probability that (w,d) is not an ϵ-approximation of (w^*,d^*) w.r.t. points for $\mathbf{P}'$ is less than δ.
4. Since the VC dimension of this problem is $n+1$, according to Theorem 2.1 in (Blumer et al. 1989), the number of required points N is at most
   $$max(\frac{4}{\epsilon}log_2\frac{2}{\delta}, \frac{8(n+1)}{\epsilon}log_2\frac{13}{\epsilon}) \qquad (1)$$
   and any algorithm which produces consistent values of w and d with the following constraints:
   $$\text{for every } x \in X^+, w \cdot x \leq d, \text{ and}$$
   $$\text{for every } x \in X^-, w \cdot x > d \qquad (2)$$
   can be a learning algorithm.
   We can use a linear programming algorithm (for example, Karmarkar's algorithm(Karmarker 1984)) for the above algorithm by considering the following constraints:
   $$\text{for every } x \in X^+, w \cdot x \leq d, \text{ and}$$
   $$\text{for every } x \in X^-, w \cdot x \geq d+1.$$
   Clearly, there exists a solution for these constraints if and only if there exists a solution for the constraints (2) and the time of finding w and d is bounded by a polynomial of n.

Then, we translate the problem in the statement of the theorem into the following:

1. Instead of a pair (A,B), we consider a point x_{A-B} in $[0,\infty)^n$ such that
   $$x_{A-B} = ((A_{(1)}-B_{(1)})^2, (A_{(2)}-B_{(2)})^2, ..., (A_{(n)}-B_{(n)})^2).$$
   The probability distribution $\mathbf{P}''$ for x_{A-B} is defined according to the original probability distribution $\mathbf{P}^2$.

```
Learn_from_qualitative_distance($\epsilon$,$\delta$,$n$)
$\epsilon$: accuracy
$\delta$: confidence
$n$: the number of dimension
begin
    Receive $max(\frac{4}{\epsilon}log_2\frac{2}{\delta}, \frac{8(n+1)}{\epsilon}log_2\frac{13}{\epsilon})$ pairs of
    points and their dichotomy from the teacher.
    for every pair $(A,B) \in NEAR$
        add the following inequality to the constraint set:
            $dist_W(A,B) \leq D$
    for every pair $(A,B) \in FAR$
        add the following inequality to the constraint set:
            $dist_W(A,B) \geq D+1$
    Get consistent values for the above constraint set
    by linear programming and output $W_{(i)}$s and $D$.
end
```

Figure 3: algorithm for qualitative distance

2. Then, the condition that an ϵ-approximation of (W^*, D^*) w.r.t. difference vectors for $\mathbf{P}^2$ becomes an ϵ-approximation of (W^*, D^*) w.r.t. points for the distribution $\mathbf{P}''$.
3. And the qualitative distance information is equivalent to the following conditions:

$$\text{For every } (A,B) \in NEAR, W^* \cdot x_{A-B} \leq D^*,$$

$$\text{and for every } (A,B) \in FAR, W^* \cdot x_{A-B} > D^*.$$

From the above discussion, by using a linear programming algorithm, we can find W and D such that the probability that (W,D) is not an ϵ-approximation of (W^*, D^*) w.r.t. points for $\mathbf{P}''$ is less than δ with required points bounded by (1) and the time of finding w and d is bounded by a polynomial of n. Since if (W,D) is an ϵ-approximation of (W^*, D^*) w.r.t. points for $\mathbf{P}''$ then (W,D) is an ϵ-approximation of (W^*, D^*) w.r.t. difference vectors for $\mathbf{P}^2$, W and D are a wanted weight and a wanted threshold for the original problem. □

We show the overall algorithm to compute a weight and a threshold from qualitative distance information in Fig 3.

Extension

In this section, we discuss an extension to relative distance information. In the previous section, we divide a sample set of pairs into two sets FAR and $NEAR$. Instead of that, we now consider triples of points (A,B,C) which express the order of $dist_{W^*}(A,B)$ and $dist_{W^*}(A,C)$. This kind of setting has been used in the multidimensional scaling and called the *method of triads* (Torgerson 1952).

Let $\mathbf{P}$ be any probability distribution over n-dimensional Euclidean space, E^n. The teacher selects a weight vector W^* from $[0,\infty)^n$. The teacher selects N triples of points in $E^n \times E^n \times E^n$ according to $\mathbf{P}^3$. Let a set of selected triples be $X = \{(A_1,B_1,C_1), ..., (A_N,B_N,C_N)\}$.

The teacher gives the following relative distance information for every triple $(A,B,C) \in X$:

if $dist_{W^*}(A,B) \leq dist_{W^*}(A,C)$ then $(A,B) \leq (A,C)$
if $dist_{W^*}(A,B) > dist_{W^*}(A,C)$ then $(A,B) > (A,C)$

Let W be a vector in $[0,\infty)^n$. The difference between W and W^*, *diff*(W,W^*) is defined as the union of

$$\{(A,B,C) \in E^n \times E^n \times E^n | dist_W(A,B) \leq dist_W(A,C) \text{ and } dist_{W^*}(A,B) > dist_{W^*}(A,C)\}$$

and

$$\{(A,B,C) \in E^n \times E^n \times E^n | dist_W(A,B) > dist_W(A,C) \text{ and } dist_{W^*}(A,B) \leq dist_{W^*}(A,C)\}.$$

W is said to be an *ϵ-approximation of W^* w.r.t. relative distance for* $\mathbf{P}^3$ if the probability of $\mathbf{P}^3(\textit{diff}(W,W^*))$ is at most ϵ.

The following theorem shows that this framework is also PAC-learnable.

Theorem 2 *There exists a learning algorithm which satisfies the following conditions for all probability distribution over E^n, $\mathbf{P}$, and all constants ϵ and δ in the range $(0,1)$:*

1. *The teacher selects W^*.*
2. *The teacher gives N triples according to $\mathbf{P}^3$ with relative distance information according to W^* to the algorithm.*
3. *The algorithm outputs a weight vector W.*
4. *The probability that W is not an ϵ-approximation of W^* w.r.t. relative distance for $\mathbf{P}^3$ is less than δ.*
5. *The size of required triples N is bounded by a polynomial in n, $\frac{1}{\epsilon}$, $\frac{1}{\delta}$ and its running time is bounded by a polynomial in the size of required triples.*

Proof: We translate the above problem into the following problem:

1. Instead of a triple (A,B,C), we consider a point $x_{\langle A,B,C \rangle}$ in E^n such that

$$x_{\langle A,B,C \rangle} = (x_{(1)}, ..., x_{(n)})$$

where $x_{(i)} = (A_{(i)} - B_{(i)})^2 - (A_{(i)} - C_{(i)})^2$.
The probability distribution $\mathbf{P}'$ for $x_{\langle A,B,C \rangle}$ is defined according to the original probability distribution $\mathbf{P}^3$.

2. Then, the relative distance information is equivalent to the following conditions:

$$\text{For } (A,B) \leq (A,C), W^* \cdot x_{\langle A,B,C \rangle} \leq 0,$$

$$\text{and for } (A,B) > (A,C), W^* \cdot x_{\langle A,B,C \rangle} > 0.$$

```
Learn_from_relative_distance($\epsilon$,$\delta$,$n$)
$\epsilon$: accuracy
$\delta$: confidence
$n$: the number of dimension
begin
  Receive $max(\frac{4}{\epsilon}log_2\frac{2}{\delta}, \frac{8n}{\epsilon}log_2\frac{13}{\epsilon})$ triples of
  points and their relative distance information
  from the teacher.
  for every triple $(A,B,C)$
    if $(A,B) \leq (A,C)$
      then add the following inequality to the
        constraint set:

          $dist_W(A,B) \leq dist_W(A,C)$

    if $(A,B) > (A,C)$
      then add the following inequality to the
        constraint set:

          $dist_W(A,B) \geq dist_W(A,C) + 1$

  Get consistent values for the above constraint set
  by linear programming and output $W_{(i)}$s.
end
```

Figure 4: algorithm for relative distance

3. Then, the translated problem becomes a problem to learn a half-space separated by a hyperplane through the origin.

Since the VC dimension of translated problem is n, the number of required triples is

$$max(\frac{4}{\epsilon}log_2\frac{2}{\delta}, \frac{8n}{\epsilon}log_2\frac{13}{\epsilon}) \qquad (3)$$

and by using linear programming algorithm, we can find a consistent W for the above relative distance information with the running time bounded by polynomial of n. □

We show the overall algorithm to compute a weight from relative distance information in Fig 4.

Conclusion

This paper discusses about a theoretical framework for learning weights in a similarity function. In this paper, we show that by using two kinds of qualitative distance information *NEAR* and *FAR*, we can efficiently learn a weight which has an error rate less than ϵ with a probability of at least $1 - \delta$. We also discuss an extension to learning a weight from relative distance information. We are now planning to make experiments based on this method in various domains.

Acknowledgments

We thank Kazuo Asakawa and Fumihiro Mauryama for support for this research and Nobuhiro Yugami and Yasubumi Sakakibara for an insightful discussion. We also thank referees for helpful comments.

References

Aha, D. W. 1989. Incremental Instance-Based Learning of Independent and Graded Concept Descriptions. Proceedings of 6th International Workshop on Machine Learning, 387 - 391.

Aha, D. W., Kibler, D., and Albert, M. K. 1991. Instance-Based Learning Algorithms. *Machine Learning*, 2: 37 - 66.

Albert, M. K., and Aha, D. W. 1991. Analyses of Instance-Based Learning Algorithms. Proceedings of AAAI'91, 553 - 558.

Blumer, A., Ehrenfeucht, A., Haussler, D. and Warmuth, M. K. 1989. Learnability and the Vapnik-Chervonenkis Dimension. JACM, 36: 929 - 965.

Callan, J. P., Fawcett, T. E., and Rissland, E. L. 1991. CABOT: An Adaptive Approach to Case-Based Search. Proceedings of IJCAI'91, 803 - 808.

Cardie, C. 1993. Using Decision Trees to Improve Case-Based Learning. Proceedings of 10th International Workshop on Machine Learning, 25 - 32.

Janke, K. P., and Lange, S. 1993. Case-Based Representation and Learning of Pattern Languages. Proceedings of ALT'93, 87 - 100.

Karmarkar, N. 1984. A New Polynomial-time Algorithm for Linear Programming. Combinatorica, 4: 373 - 395.

Langley, P., and Iba, W. 1993. Average-Case Analysis of a Nearest Neighbor Algorithm. Proceedings of IJCAI'93, 889 - 894.

Stanfill, C., and Waltz, D. 1986. Toward Memory-Based Reasoning. CACM, 29: 1213 - 1228.

Torgerson, W. S. 1952. Multidimensional Scaling: I. Theory and Method. Psychometrika, 17: 401 - 409.

Valiant, L. G. 1984. A Theory of the Learnable. CACM, 27: 1134 - 1142.