Increasing Precision of Credible Case-Based Inference

Thomas Gabel and Martin Riedmiller

Neuroinformatics Group
Department of Mathematics and Computer Science
Institute of Cognitive Science
University of Osnabrück, 49069 Osnabrück, Germany
{thomas.gabel|martin.riedmiller}@uni-osnabrueck.de

Abstract. Credible case-based inference (CCBI) is a new and theoretically sound inferencing mechanism for case-based systems. In this paper, we formally investigate the level of precision that CCBI-based retrieval results may yield. Building upon our theoretical findings, we derive a number of optimization criteria that can be utilized for learning such similarity measures that bring about more precise predictions when used in the scope of CCBI. Our empirical experiments support the claim that, given appropriate similarity measures, CCBI can be enforced to produce highly precise predictions while its corresponding level of confidence is only marginally impaired.

1 Introduction

Credible case-based inference (CCBI) has been recently proposed as a new retrieval paradigm for case-based problem solving [6]. It features a number of desirable theoretical properties and allows for deriving formal statements about its performance. Furthermore, it makes few assumptions regarding the application domain for which it can be used and concerning the case structure and similarity measures employed during the inference process.

The issue mentioned last – the use of fixed similarity measures – depicts one point of departure for the work described in the paper at hand. We are going to consider the similarity measure CCBI builds upon as a variable. The second point of departure stems from the fact that the level of precision obtained when doing inference with CCBI has been recently shown to be only of moderate quality. Combining these two issues, our goal is to increase the precision of CCBI's predictions by modifying and optimizing the similarity measures that CCBI builds upon.

In so doing, we will first formalize the notion of a precise retrieval result in the context of CCBI and prove a number of its theoretical properties (Section 3). Then, we suggest the learning of high-precision similarity measures using a recently proposed learning framework and utilizing a number of novel precision-oriented error functions that we develop (Section 4). Finally, we empirically evaluate our findings using several benchmark data sets (Section 5). Before starting off, we briefly summarize the core concepts of CCBI in Section 2.

2 Credible Case-Based Inference

In [6], Hüllermeier introduced credible case-based inference as a novel method for retrieving candidate solutions in case-based problem solving. CCBI is built upon a sound formalization of the CBR paradigm and allows for proving some of its theoretical properties. In this section, we briefly outline those specifics of CCBI and of its inference mechanism that are of relevance in the scope of this paper, and we also point to some possibilities for improving its performance.

2.1 Notation and Outline of CCBI

Throughout this paper, we denote by $\mathcal{X}$ a problem space and by $\mathcal{L}$ a solution space, where a case consists of a problem part $x \in \mathcal{X}$ and solution part $\lambda_x \in \mathcal{L}$. Further, a case base $\mathcal{M}$ is a collection of cases $\langle x_i, \lambda_{x_i} \rangle \in \mathcal{M}$, $1 \leq i \leq |\mathcal{M}|$.

Motivating CCBI, the well-known CBR hypothesis that "similar problems have similar solutions" has been equipped with the formal interpretation that

$$\forall x, y \in \mathcal{X} : sim_{\mathcal{L}}(\lambda_x, \lambda_y) \geq sim_{\mathcal{X}}(x, y). \tag{1}$$

Since, however, in practice this requirement will typically be frequently violated, [6] introduces the concept of a similarity profile ζ that is defined by

$$\zeta(\alpha) := \inf_{\substack{x,y \in \mathcal{X} \\ sim_{\mathcal{X}}(x,y)=\alpha}} sim_{\mathcal{L}}(\lambda_x, \lambda_y) \quad \text{for all } \alpha \in [0,1].$$

As ζ is generally unknown, the notion of a *similarity hypothesis* $h : [0,1] \to [0,1]$ is introduced which is meant to approximate ζ. Of special interest are similarity hypotheses that are consistent with a given data set $\mathcal{M}$, i.e. for which it holds

$$\forall \langle x, \lambda_x \rangle, \langle y, \lambda_y \rangle \in \mathcal{M} : sim_{\mathcal{X}}(x, y) = \alpha \Rightarrow sim_{\mathcal{L}}(\lambda_x, \lambda_y) \geq h(\alpha).$$

One such data-consistent hypothesis that will play a major role throughout this paper takes the form of a step function over a partition A_k of the problem similarity interval $[0,1]$ and is called *empirical similarity profile*. It is defined as a function $h_{\mathcal{M}} : [0,1] \to [0,1]$ with

$$h_{\mathcal{M}} : x \mapsto \sum_{k=1}^{m} \beta_k \cdot \mathbb{I}_{A_k}(x) \quad \text{and} \quad \beta_k := \min_{\substack{\langle x, \lambda_x \rangle, \langle y, \lambda_y \rangle \in \mathcal{M} \\ sim_{\mathcal{X}}(x,y) \in A_k}} sim_{\mathcal{L}}(\lambda_x, \lambda_y) \tag{2}$$

where $A_k = [\alpha_{k-1}, \alpha_k)$ for $1 \leq k < m$, $A_m = [\alpha_{m-1}, \alpha_m]$ and $0 = \alpha_0 < \alpha_1 < \ldots < \alpha_m = 1$ ($\mathbb{I}_A(x) = 1$ if $x \in A$, $\mathbb{I}_A(x) = 0$ else, and $\min \emptyset = 1$ by definition). Thus, by definition the following relaxation of the constraint in Equation 1 holds

$$\forall \langle x, \lambda_x \rangle, \langle y, \lambda_y \rangle \in \mathcal{M} : sim_{\mathcal{L}}(\lambda_x, \lambda_y) \geq h_{\mathcal{M}}(sim_{\mathcal{X}}(x, y)).$$

In contrast to, for example, k-NN prediction, CCBI does not provide point predictions, but sets of candidate solutions. So, for predicting the label λ_q of a new query problem $q \in \mathcal{X}$, the notion of a *credible solution set* $C(q)$ is introduced

and, when doing inference with a finite data set $\mathcal{M}$, it is suggested that the requested solution is an element of the following *estimated* credible solution set

$$C^{est}(q) = \bigcap_{c \in \mathcal{M}} \{\lambda | sim_{\mathcal{L}}(\lambda, \lambda_c) \geq h_{\mathcal{M}}(sim_{\mathcal{X}}(q, c))\}. \quad (3)$$

For this inference mechanism, an estimation can be derived concerning the probability that a correct prediction ($\lambda_q \in C^{est}(q)$) is made subject to $|\mathcal{M}|$ and m.

2.2 Weaknesses of CCBI

As indicated before, high confidence levels in CCBI typically come along with poor levels of precision, meaning that the solution set $C^{est}(q)$ returned for some query q, contains a large number of elements. While some extensions to pure CCBI have been suggested to combat that shortcoming (e.g. the use of probabilistic similarity profiles [6]), the underlying problem of low precision is not a flaw in CCBI's inferencing mechanism, but is actually caused by poor and unsuitable problem similarity measures employed. Consequently, our goal pursued in this paper is to improve the problem similarity measures in such a manner that the imprecision of returned credible solution sets is reduced, while we rely on the basic form of CCBI (cf. Equation 3) to actually perform the retrieval.

3 Imprecision in CCBI

Aiming at the reduction of imprecision in inferencing with CCBI by adjusting problem similarity, we start off by formally investigating what it means for a credible set C^{est} to be precise or imprecise.

3.1 Formalization

Precision is usually defined as the share of correct items retrieved to the overall number of items retrieved [1]. Therefore, intuitively, we might say that a credible solution set $C^{est}(q)$ as prediction for the solution of q is of maximal precision if it contains the correct solution λ_q and no further elements. However, in the scope of CCBI, we need to extend that view slightly.

Definition 1 (Precise Solution Set). *Let $\mathcal{M}$ be a case base and $\langle q, \lambda_q \rangle$ be a case with $q \in \mathcal{X}$ and $\lambda_q \in \mathcal{L}$ as corresponding solution. Then, we call*

$$C^{prec}(q) = \bigcap_{\langle c, \lambda_c \rangle \in \mathcal{M}} \{\lambda | sim_{\mathcal{L}}(\lambda, \lambda_c) \geq sim_{\mathcal{L}}(\lambda_q, \lambda_c)\}$$

the precise solution set for q.

The following lemma lets us conclude that $C^{prec}(q) \subseteq C^{est}(q)$ is the smallest, hence, maximally precise solution set that (a) will be returned when doing case-based inference with CCBI and that (b) is correct in the sense that $\lambda_q \in C^{est}(q)$. In other words, besides the correct solution λ_q, all other elements from $C^{prec}(q)$ are always included in $C^{est}(q)$, no matter which similarity profile h is used during retrieval.

Lemma 1. *For any similarity hypothesis h consistent with the case data $\mathcal{M}$ and any query q, the credible solution set $C^{est}(q)$ contains $C^{prec}(q)$ as a subset.*

Proof: Let $\lambda_d \in C^{prec}(q)$. So, for all $\langle c, \lambda_c \rangle \in \mathcal{M}$ it holds: $sim_{\mathcal{L}}(\lambda_d, \lambda_c) \geq sim_{\mathcal{L}}(\lambda_q, \lambda_c)$. As h is assumed to be consistent with the data in $\mathcal{M}$, it holds for all $\langle x, \lambda_x \rangle, \langle y, \lambda_y \rangle \in \mathcal{M}$ that $sim_{\mathcal{L}}(\lambda_x, \lambda_y) \geq h(sim_{\mathcal{X}}(x, y))$. Thus, for all $\langle c, \lambda_c \rangle \in \mathcal{M}$ it also holds $sim_{\mathcal{L}}(\lambda_d, \lambda_c) \geq h(sim_{\mathcal{X}}(q, c))$. Therefore, $\lambda_d \in \bigcap_{\langle c, \lambda_c \rangle \in \mathcal{M}} \{\lambda | sim_{\mathcal{L}}(\lambda, \lambda_c) \geq h(sim_{\mathcal{X}}(q, c))\} = C^{est}(q)$. (Note: $\lambda_q \in C^{prec}(q)$ by definition.) □

Note that for $|C^{prec}(q)| > 1$ to occur, we must require that there is at least one $\langle d, \lambda_d \rangle$ whose solution λ_d is at least as similar to all other solutions in $\mathcal{M}$ as λ_q. This situation is not as unrealistic as it might seem: It may occur even for symmetric and reflexive solution similarity measures, e.g. if they contain "plateaus" of maximal similarity (see Figure 1). In the remainder of this paper, however, we focus on regression tasks using the Euclidean distance as the basis for determining solution similarity, such that $sim_{\mathcal{L}}$ is a strongly monotonous function and therefore always $|C^{prec}| = 1$ (proof omitted).

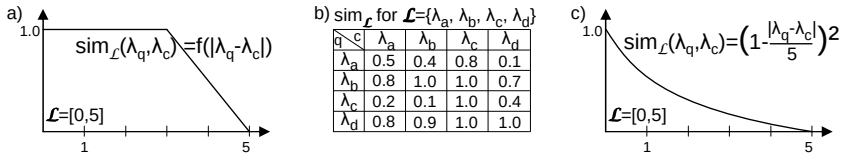

Fig. 1. Examples of solution similarity measures with $|C^{prec}| \geq 1$. In a), for $\lambda_q = 1$, for example, $C^{prec}(q) = \{\lambda_c | c \in \mathcal{M}, \lambda_c \in [1, 3]\}$. In b), it holds $C^{prec}(a) = \{\lambda_a, \lambda_b, \lambda_d\}$, whereas in c), $sim_{\mathcal{X}}$ is decreasing strongly monotonically and hence $|C^{prec}(q)| = 1$.

3.2 Similarity Measures for High-Precision CCBI

We now focus on the relation between problem similarity measures, empirical similarity profiles that can be induced from them, and their impact on CCBI.

From Lemma 1, we observe that CCBI attains its maximal precision when it holds that $C^{prec}(q) = C^{est}(q)$ for all $q \in \mathcal{X}$. Assuming the case base $\mathcal{M}$ to be fixed and considering $sim_{\mathcal{X}}$ as a variable, improving the precision of CCBI means searching for a problem similarity measure such that $C^{est}(q)$ contains as few elements as possible for as many q as possible.

Definition 2 (Maximally Precise Problem Similarity Measure). *A function $sim^{\star}_{\mathcal{X}} : \mathcal{X}^2 \to [0, 1]$ is called a maximally precise problem similarity measure for a given case base $\mathcal{M}$ and number of intervals m, if for the predictions produced by CCBI, based on the corresponding empirical similarity profile $h^{sim^{\star}_{\mathcal{X}}}_{\mathcal{M}}$, it holds that $Pr(C^{prec}(q) = C^{est}(q)) = 1$ for all $q \in \mathcal{X}$.*

Assuming the existence of an optimal problem similarity measure, it is straightforward to prove the following lemma.

Lemma 2. *Let $\mathcal{M}$ be a case base, $m > 0$ the number of intervals used for determining an empirical similarity profile $h_{\mathcal{M}}$, $sim_{\mathcal{X}}$ and $sim_{\mathcal{L}}$ be problem and solution similarity measures, respectively. If for all $\langle x, \lambda_x\rangle, \langle y, \lambda_y\rangle \in \mathcal{M}$ it holds that $sim_{\mathcal{L}}(\lambda_x, \lambda_y) = h_{\mathcal{M}}(sim_{\mathcal{X}}(x, y))$, then $sim_{\mathcal{X}}$ is a maximally precise problem similarity measure for $\mathcal{M}$, i.e. $h_{\mathcal{M}} = h_{\mathcal{M}}^{sim^{\star}_{\mathcal{X}}}$.*

Proof: We show $Pr(C^{prec}(q) = C^{est}(q)) = 1$ by proving by contradiction that, under the assumptions made, $C^{est}(q) \setminus C^{prec}(q) = \emptyset$ for all $q \in \mathcal{X}$. Assume there is a case $\langle u, \lambda_u\rangle \in \mathcal{M}$ such that $\lambda_u \in C^{est}(q) \setminus C^{prec}(q)$. The CCBI inference scheme tells that $\lambda_u \in C^{est}(q)$ implies that for all $\langle c, \lambda_c\rangle \in \mathcal{M}$ it holds $sim_{\mathcal{L}}(\lambda_u, \lambda_c) \geq h_{\mathcal{M}}(sim_{\mathcal{X}}(q, c))$. Knowing that $sim_{\mathcal{L}}(\lambda_x, \lambda_y) = h_{\mathcal{M}}(sim_{\mathcal{X}}(x, y))$ for all $\langle x, \lambda_x\rangle, \langle y, \lambda_y\rangle \in \mathcal{M}$ (precondition of Lemma 2), we conclude that $sim_{\mathcal{L}}(\lambda_u, \lambda_c) \geq sim_{\mathcal{L}}(\lambda_q, \lambda_c)$ $(\star)$ for all $\langle c, \lambda_c\rangle \in \mathcal{M}$. Further, as $\lambda_u \notin C^{prec}(q) = \bigcap_{\langle c,\lambda_c\rangle \in \mathcal{M}} \{\lambda | sim_{\mathcal{L}}(\lambda, \lambda_c) \geq sim_{\mathcal{L}}(\lambda_q, \lambda_c)\}$, there must exist a $\langle d, \lambda_d\rangle \in \mathcal{M}$ such that $sim_{\mathcal{L}}(\lambda_u, \lambda_d) \lneq sim_{\mathcal{L}}(\lambda_q, \lambda_d)$. This contradicts $(\star)$. □

Accordingly, we can force predictions produced by CCBI to be of maximal precision for a given case base $\mathcal{M}$, if we manage to provide a problem similarity measure such that the corresponding empirical similarity profile $h_{\mathcal{M}}$ features no interval in which any two pairs of cases have different levels of solution similarity. For further investigations, we introduce the notion of the empirical similarity boundary that represents a kind of counterpart to an empirical similarity profile.

Definition 3 (Empirical Similarity Boundary). *Let A_k be a partition of $[0, 1]$ as in Equation 2. We call*

$$\hat{h}_{\mathcal{M}} : x \mapsto \sum_{k=1}^{m} \gamma_k \cdot \mathbb{I}_{A_k}(x) \quad \textit{with} \quad \gamma_k := \max_{\substack{\langle x,\lambda_x\rangle, \langle y,\lambda_y\rangle \in \mathcal{M}, \\ sim_{\mathcal{X}}(x,y) \in A_k}} sim_{\mathcal{L}}(\lambda_x, \lambda_y),$$

the empirical similarity boundary for $\mathcal{M}$ (here, $\mathbb{I}_A$ is the indicator function of set A and $\max \emptyset = 0$ by definition).

From Lemma 2, it follows that CCBI has maximal precision for $\mathcal{M}$, if the corresponding empirical similarity profile and boundary are identical for intervals containing data (and by definition, $\hat{h}_{\mathcal{M}}$ is zero while $h_{\mathcal{M}}$ is one for intervals that contain no data). Note, however, that the inverse statement is not generally true.

Corollary 1. *If, for a case base $\mathcal{M}$ and a similarity measure $sim_{\mathcal{X}}$, it holds $h_{\mathcal{M}}^{sim_{\mathcal{X}}}(x) \geq \hat{h}_{\mathcal{M}}^{sim_{\mathcal{X}}}(x)$, then $sim_{\mathcal{X}}$ is maximally precise, i.e. $sim_{\mathcal{X}} = sim^{\star}_{\mathcal{X}}$.*

3.3 Modifying Problem Similarity

Next, we investigate how to exploit the statements made so far for tuning similarity measures in order to increase CCBI's precision. Speaking about modifications applied to similarity measures, we stress that we consider the solution similarity measure $sim_{\mathcal{L}}$ to be fixed. By contrast, $sim_{\mathcal{X}}$ is a variable and may (at least in theory[1]) take any value from the space of functions definable over $\mathcal{X}^2 \to [0, 1]$.

[1] In practice, we will usually confine ourselves to some "reasonable" or appropriately representable sub-space of functions.

3.3.1 Partitioning Problem Similarity

A naive approach that allows for frequently fulfilling the constraint from Corollary 1, i.e. $h_{\mathcal{M}}(x) = \hat{h}_{\mathcal{M}}(x)$ for many intervals, and so increases the probability for precise solution sets, can be realized by incrementing the number m of intervals used for determining the similarity profile and boundary (see Figure 2).

Lemma 3. *If for all* $\langle x_1, \lambda_{x_1}\rangle, \langle y_1, \lambda_{y_1}\rangle, \langle x_2, \lambda_{x_2}\rangle, \langle y_2, \lambda_{y_2}\rangle \in \mathcal{M}$ *with* $sim_{\mathcal{L}}(\lambda_{x_1}, \lambda_{y_1}) \neq sim_{\mathcal{L}}(\lambda_{x_2}, \lambda_{y_2}))$ *it holds* $sim_{\mathcal{X}}(x_1, y_1) \neq sim_{\mathcal{X}}(x_2, y_2)$*, then for any* $m \geq |\mathcal{M}|$ *there is a partition* A_k *(where* $A_k = [\alpha_{k-1}, \alpha_k]$ *for* $1 \leq k \leq m$*,* $A_m = [\alpha_{m-1}, \alpha_m]$*,* $0 = \alpha_0 < \alpha_1 < \ldots < \alpha_m = 1$*) so that* $h_{\mathcal{M}}(x) \geq \hat{h}_{\mathcal{M}}(x)\ \forall x \in [0, 1]$.

Proof: Let $\mathbb{S}_{\mathcal{M}} = \{sim_{\mathcal{X}}(x, y) | \langle x, \lambda_x\rangle, \langle y, \lambda_y\rangle \in \mathcal{M}\}$ be the set of all problem similarity levels occurring for cases within $\mathcal{M}$. Thus, $|\mathbb{S}_{\mathcal{M}}| \leq |\mathcal{M}|^2$, and $\forall s \in \mathbb{S}_{\mathcal{M}}$ the set $\{sim_{\mathcal{L}}(\lambda_x, \lambda_y) | \langle x, \lambda_x\rangle, \langle y, \lambda_y\rangle \in \mathcal{M}, sim_{\mathcal{X}}(x, y) = s\}$ contains exactly one element. We define $\mathbb{S}^L_{\mathcal{M}} = [s_1, \ldots, s_{|\mathcal{M}|}]$ as an ordered list that arranges all elements from $\mathbb{S}_{\mathcal{M}}$ in ascending order. Next, we set $\alpha_k = \mathbb{S}^L_{\mathcal{M}}[k]$ for $1 \leq k \leq m$, and for $k > |\mathbb{S}_{\mathcal{M}}|$ we set α_k distributed equidistantly over $[1 - \mathbb{S}^L_{\mathcal{M}}[|\mathbb{S}_{\mathcal{M}}|], 1]$. Obviously, A_k is a well defined partition. If $k > |\mathbb{S}_{\mathcal{M}}|$, then $1 = h_{\mathcal{M}}(x) > \hat{h}_{\mathcal{M}}(x) = 0$ for $x \in A_k$ by definition, because there are no $\langle x, \lambda_x\rangle, \langle y, \lambda_y\rangle \in \mathcal{M}$ with $sim_{\mathcal{X}}(x, y) \in A_k$. If, however, $k \leq |\mathbb{S}_{\mathcal{M}}|$ it holds that $|\{sim_{\mathcal{L}}(\lambda_x, \lambda_y) | \langle x, \lambda_x\rangle, \langle y, \lambda_y\rangle \in \mathcal{M}, sim_{\mathcal{X}}(x, y) \in A_k\}| = 1$. Consequently, $\min_{\langle x,\lambda_x\rangle, \langle y,\lambda_y\rangle \in \mathcal{M}, sim_{\mathcal{X}}(x,y) \in A_k} sim_{\mathcal{L}}(\lambda_x, \lambda_y) = \max_{\langle x,\lambda_x\rangle, \langle y,\lambda_y\rangle \in \mathcal{M}, sim_{\mathcal{X}}(x,y) \in A_k} sim_{\mathcal{L}}(\lambda_x, \lambda_y)$ and, hence, $h_{\mathcal{M}}(x) = \hat{h}_{\mathcal{M}}(x)$ for all $x \in A_k$. □

Lemma 3 suggests that increasing the value of m may support the precision of the solution sets returned by CCBI. Unfortunately, there are two important drawbacks to be considered. On the one hand, as shown in [6], increasing m also decreases the probability that the correct solution λ_q for some problem $q \in \mathcal{X}$ is not in the solution set, because $Pr(\lambda_q \notin C^{est}(q)) \leq 2m/(1 + |\mathcal{M}|)$.

On the other hand, even if $m \to \infty$, that lemma fails to guarantee maximal precision, if there exist pairs of cases in $\mathcal{M}$ whose problem parts have identical values of problem similarity, but whose solution parts differ in their solution similarities, i.e. $\exists \langle x_1, \lambda_{x_1}\rangle, \langle y_1, \lambda_{y_1}\rangle, \langle x_2, \lambda_{x_2}\rangle, \langle y_2, \lambda_{y_2}\rangle \in \mathcal{M} : sim_{\mathcal{X}}(x_1, y_1) = sim_{\mathcal{X}}(x_2, y_2)$ and $sim_{\mathcal{L}}(\lambda_{x_1}, \lambda_{y_1}) \neq sim_{\mathcal{L}}(\lambda_{x_2}, \lambda_{y_2})$. In particular, the latter problem can be avoided only by modifying the problem similarity measure $sim_{\mathcal{X}}$.

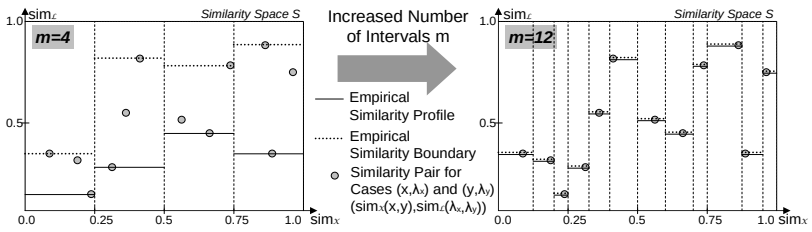

Fig. 2. By incrementing the number m of intervals, in principle maximal precision can be attained, although such an approach is not valuable in practice.

3.3.2 Basic Problem Similarity Modifiers

Drawing from the preceding remarks, the need for adapting $sim_{\mathcal{X}}$ becomes obvious. As a very special case of Lemma 2, $sim_{\mathcal{X}}$ would trivially be a maximally precise problem similarity measure, if for all cases $\langle x, \lambda_x \rangle, \langle y, \lambda_y \rangle \in \mathcal{M}$ it held

$$sim_{\mathcal{X}}(x, y) = sim_{\mathcal{L}}(\lambda_x, \lambda_y). \tag{4}$$

This idea of employing the solution similarity measure as a kind of similarity teacher for learning a suitable problem similarity measure is not new. It has already been employed for practical tasks [5], has been formalized in [9], and empirically investigated in [3, 4]. Although striving for a problem similarity measure that fulfills the constraint from Equation 4 in order to increase the precision of CCBI seems appealing at first glance, we must be aware that such a naive approach neglects all the knowledge about how the inferencing mechanism of CCBI (cf. Section 2.1) works and, hence, would waste useful background knowledge that can guide the search for a $sim_{\mathcal{X}}$ that yields high precision predictions.

By the same arguments, the strength of an empirical similarity profile ($h_{\mathcal{M}}^{sim_{\mathcal{X}}^1}$ is stronger than $h_{\mathcal{M}}^{sim_{\mathcal{X}}^2}$ iff. $h_{\mathcal{M}}^{sim_{\mathcal{X}}^1}(\cdot) \geq h_{\mathcal{M}}^{sim_{\mathcal{X}}^2}(\cdot)$, cf. [6]) as a function of the problem similarity measure and with m fixed is only of limited use, when searching for a $sim_{\mathcal{X}}$ that induces high-precision predictions. We will empirically support this claim in Section 5.

A final remark concerns the practical representation of similarity measures. When the problem domain $\mathcal{X}$ is finite, $sim_{\mathcal{X}}$ can be represented using a table and thus, $sim_{\mathcal{X}}(x, y)$ may be adjusted individually for any pair of problems from $\mathcal{X}^2$. Typically, however, $\mathcal{X}$ is a multi-dimensional, continuous space and problem similarity measures defined over $\mathcal{X}^2$ are represented in a parameterized way. For example, for $\mathcal{X} = \mathbb{R}^n$ one may set $sim_{\mathcal{X}}(x, y) = \frac{1}{1+||x-y||_p}$, where $p \geq 1$ is a parameter determining the norm used (e.g. $p = 2$ for Euclidean distance). Here, when modifying p, $sim_{\mathcal{X}}$ is changed for vast parts of its domain. As a consequence of such a parameterized similarity measure representation, fulfilling the constraint from Equation 4 is in general infeasible (see Figure 3).

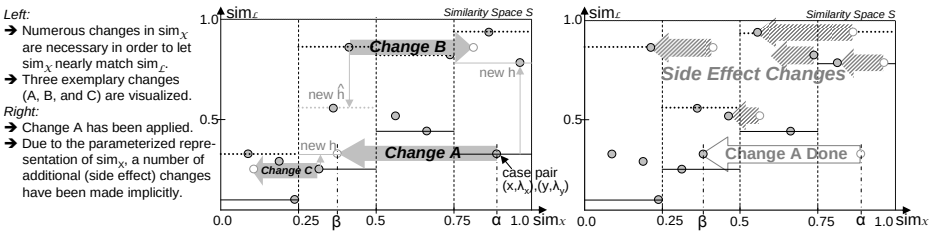

Fig. 3. Depending on the used representation for similarity measures, changing $sim_{\mathcal{X}}$ may entail changes in the problem similarity for numerous problems $x, y \in \mathcal{X}$. Here, for the case pair $\langle x, \lambda_x \rangle, \langle y, \lambda_y \rangle$ a change of $sim_{\mathcal{X}}(x, y)$ from α to β is desired. As indicated in the right part, conducting change A results in a number of side effect changes for other case pairs that may also cause the similarity profile h to change.

4 Precision-Oriented Tuning of Similarity Measures

Optimizing similarity measures in CBR is not a novel issue. A lot of work in this direction has been done, e.g. in the area of nearest-neighbor classification. Here, one tries to adjust feature weights by examining pre-classified training data [10, 2, 11]. Stahl [7] introduced a comprehensive methodology and a widely applicable framework for learning similarity measures which we utilize and further develop in the scope of this work. Optimizing similarity measures for the use within CCBI and with the goal of increasing the precision of predictions, however, is novel. In this section, we first briefly outline the learning framework mentioned. Subsequently, we develop and analyze required error measures that are geared towards improving the precision of case-based inferencing with CCBI.

4.1 A Framework for Learning Similarity Measures

The framework for learning similarity measures we utilize does not rely on absolute information of a case's utility for some query, but it allows for exploiting *relative* utility feedback [7]. A second important feature boosting its applicability is that it is not restricted to learning feature weights, but allows for optimizing a broad class of similarity measures [8].

For the representation of problem similarity, typical knowledge-intensive similarity measures consisting of feature weights w_i and feature-specific local similarity measures $sim_{\mathcal{X}_i}$ are assumed, where $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$, and for the features of cases $x, y \in \mathcal{X}$ it holds that $x_i, y_i \in \mathcal{X}_i$:

$$sim_{\mathcal{X}}(x, y) = \sum_{i=1}^{n} w_i \cdot sim_{\mathcal{X}_i}(x_i, y_i). \tag{5}$$

Local similarity measures are commonly represented as similarity tables which assess all pairwise similarity values for symbolic features or as difference-based similarity functions which map feature differences to similarity values for numerical features (see [8] for an illustration).

For the task of optimizing feature weights as well as local similarity measures, we developed an algorithm that performs search in the space of representable similarity measures using evolutionary algorithms (EA). An EA maintains a population of individuals (individuals correspond to similarity measures) and evolves it using specialized stochastic operators (crossover and mutation) by which new individuals (offspring) are created. Each individual is associated with a fitness value and the least fit individuals are periodically excluded from the evolution process (selection). So, the learning algorithm searches for the *fittest individual*, whose corresponding similarity measure yields the minimal value of an error function on the training data. For more details on this learning approach and on the representation of similarity measures as individuals, we refer to [8].

Fitness Functions
A crucial component when using an evolution-based optimization technique is

the fitness function used for assessing the usefulness of the respective individual. Thus, for the task at hand, we must associate each similarity measure with a fitness value. While for learning similarity measures from relative case utility feedback, the retrieval *index error* [7] represents an appropriate fitness function, we found that more effort must be put into the fitness function's definition [3] when similarity measure optimization is to be performed for classification and regression tasks, where often only some kind of binary feedback (e.g. retrieved case has correct class or not) is available. Most of the corresponding fitness functions we investigated made use of a solution similarity measure and/or tried to induce relative utility feedback such that an index error was applicable. Being developed for usage in combination with k-nearest neighbor retrieval those error functions are unfortunately no longer usable if we work with CCBI and intend to improve the precision of the retrieved solution sets it returns. Hence, next we derive a number of candidate error functions that may be used as fitness functions when performing problem similarity measure optimization for CCBI.

4.2 Precision-Oriented Error Measures

Considering a fixed set of cases $\mathcal{M}$, a fixed number of intervals m, and a fixed solution similarity measure $sim_{\mathcal{L}}$, we can observe that

a) changing the problem similarity measure $sim_{\mathcal{X}}$ yields a shifting of data points in the similarity space $S = [0,1] \times [0,1]$ (see Figure 3) along the x-axis,
b) the precision of returned solution sets is heavily influenced by the data distribution in that space,
c) maximal precision can be attained, if the data is distributed in such a manner that the statement of Lemma 2 holds,
d) imprecision can arise, if there are cases $\langle x, \lambda_x \rangle, \langle y, \lambda_y \rangle$ such that $h_{\mathcal{M}}(sim_{\mathcal{X}}(x, y)) < sim_{\mathcal{L}}(\lambda_x, \lambda_y)$, in particular if this inequality holds for all $\langle y, \lambda_y \rangle \in \mathcal{M}$,

where we refer by $h_{\mathcal{M}}$ to the empirical similarity profile for the currently considered problem similarity measure $sim_{\mathcal{X}}$.

Departing from observation c), it is intuitive to employ the squared distance between the empirical similarity profile and boundary, summed over all intervals, as an error function (high fitness subsequently corresponds to a low error value).

Definition 4 (Boundary to Profile Error). *Given a case base $\mathcal{M}$, a partition A_k of $[0,1]$ into m intervals, a problem and solution similarity measure $sim_{\mathcal{X}}$ and $sim_{\mathcal{L}}$, and the respective empirical similarity profile $h_{\mathcal{M}}$ and boundary $\hat{h}_{\mathcal{M}}$,*

$$E_{B2P}(sim_{\mathcal{X}}) = \sum_{i=1}^{m} \left(\hat{h}_{\mathcal{M}}(x_i) - h_{\mathcal{M}}(x_i) \right)^2$$

defines the boundary to profile error *of $sim_{\mathcal{X}}$ for $\mathcal{M}$ (where $\forall x_i$, it holds $x_i \in A_i$).*

Thus, $E_{B2P} = 0$ implies that $sim_{\mathcal{X}}$ is a maximally precise problem similarity measure. Despite this, E_{B2P} is apparently only of limited use, because

the precision a problem similarity measure yields also strongly depends on the distribution of similarity pairs within each interval A_i (cf. observation b)). An example of two measures with $E_{B2P}(sim^1_{\mathcal{X}}) = E_{B2P}(sim^2_{\mathcal{X}})$ where $sim^2_{\mathcal{X}}$ is the presumably more precise one, is shown in Figure 4. Thus, a straightforward extension of Definition 4 takes observation d) into account by summing the squared distances between individual data points in S and their respective profile values.

Definition 5 (Solution Similarity to Profile Error). *Using the same preconditions as before, the* solution similarity to profile error *is defined as*

$$E_{SS2P}(sim_{\mathcal{X}}) = \sum_{\langle x,\lambda_x\rangle\in\mathcal{M}} \sum_{\langle y,\lambda_y\rangle\in\mathcal{M}} \left(sim_{\mathcal{L}}(\lambda_x,\lambda_y) - h_{\mathcal{M}}(sim_{\mathcal{X}}(x,y))\right)^2 .$$

Again, although $E_{SS2P} = 0$ assures that $h = \hat{h}$ and although E_{SS2P} regards the distribution of similarity pairs within intervals more smartly than E_{B2P}, the distance between solution similarities and profile values is only a coarse indicator of whether imprecise solution sets C^{est} will occur.

Having taken a closer look at how $C^{est}(q)$ is defined, i.e. on Equation 3, and knowing that we obtain $C^{est} = C^{prec}$ if $h_{\mathcal{M}}(sim_{\mathcal{X}}(x,y)) = sim_{\mathcal{L}}(\lambda_x,\lambda_y)$ everywhere, we can conclude that an "imprecise λ" is in one of the intersected sets $\{\lambda | sim_{\mathcal{L}}(\lambda,\lambda_c) \geq h_{\mathcal{M}}(sim_{\mathcal{X}}(q,c))\}$, if there exists a $\langle u,\lambda_u\rangle \in \mathcal{M}$ with $sim_{\mathcal{L}}(\lambda_u,\lambda_c) \in (h_{\mathcal{M}}(sim_{\mathcal{X}}(q,c)), sim_{\mathcal{L}}(\lambda_q,\lambda_c)]$[2]. This gives rise to defining:

Definition 6 (Pairs in Imprecision Interval Error). *Let the same preconditions be given as before, and define* $I_{x,y} = (h_{\mathcal{M}}(sim_{\mathcal{X}}(x,y)), sim_{\mathcal{L}}(\lambda_x,\lambda_y)]$ *as the* imprecision interval *for the case pair* $\langle x,\lambda_x\rangle, \langle y,\lambda_y\rangle$*. Then, we call*

$$E_{PII}(sim_{\mathcal{X}}) = \sum_{\langle x,\lambda_x\rangle\in\mathcal{M}} \sum_{\langle y,\lambda_y\rangle\in\mathcal{M}} \sum_{\substack{\langle u,\lambda_u\rangle\in\mathcal{M} \\ u\neq x}} f(x,y,u)$$

where $f(x,y,u) = \begin{cases} 1 & \text{if } sim_{\mathcal{L}}(\lambda_u,\lambda_y) \in I_{x,y} \\ 0 & \text{else} \end{cases}$ *the pairs in imprecision interval error.*

Function f in Definition 6 indicates whether for two cases $\langle x,\lambda_x\rangle, \langle y,\lambda_y\rangle$ from the case base there is a $\langle u,\lambda_u\rangle \in \mathcal{M}$ $(u \neq x)$ such that $sim_{\mathcal{L}}(\lambda_u,\lambda_y) > h_{\mathcal{M}}(sim_{\mathcal{X}}(x,y))$ and $sim_{\mathcal{L}}(\lambda_u,\lambda_y) \leq sim_{\mathcal{L}}(\lambda_x,\lambda_y)$. Consequently, one may say that the case pair $\langle x,\lambda_x\rangle$ and $\langle y,\lambda_y\rangle$ bears some potential for yielding imprecision (see Figure 4a for an illustration).

However, for $C^{est}(x) \setminus C^{prec}(x) \neq \emptyset$ to actually occur and for λ_u to be in that difference set, the two inequations mentioned in the previous paragraph must not just hold for y, but also for *all* $\langle z,\lambda_z\rangle \in \mathcal{M}$ $(z \neq y)$. In other words, if we can find a $\langle z,\lambda_z\rangle \in \mathcal{M}$ such that $sim_{\mathcal{L}}(\lambda_u,\lambda_z) < h_{\mathcal{M}}(sim_{\mathcal{X}}(x,z))$, then the considered case pair $\langle x,\lambda_x\rangle$ and $\langle y,\lambda_y\rangle$ no longer bears potential for causing imprecision concerning λ_u. The following error function takes care of that fact.

[2] Note that λ_u is in the mentioned set *anyway*, if $sim_{\mathcal{L}}(\lambda_u,\lambda_c) > sim_{\mathcal{L}}(\lambda_q,\lambda_c)$, even if $h = \hat{h}$.

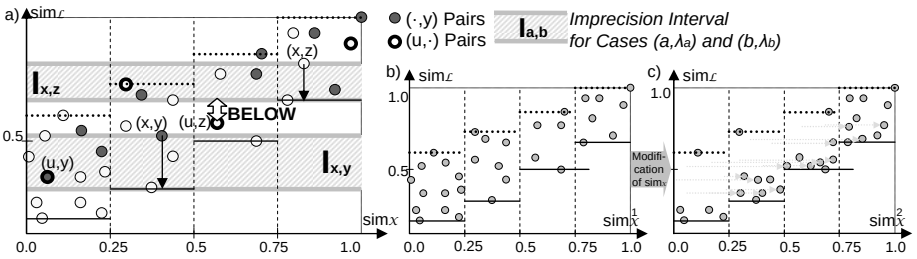

Fig. 4. In b+c), the similarity space for two problem measures $sim^1_\mathcal{X}$ and $sim^2_\mathcal{X}$ is shown (both yield the same value of E_{B2P}) which are presumed to be of different precision (see text). In a), an illustration for Definitions 6+7 is provided. Looking at case pair $\langle x, \lambda_x\rangle, \langle y, \lambda_y\rangle$, there are two data points $(\cdot, y)$ in the corresponding imprecision interval $I_{x,y}$. However, w.r.t. u, (x, y) bears no potential for yielding imprecision: Apparently, there exists a $\langle z, \lambda_z\rangle \in \mathcal{M}$ such that (u, z) is below $I_{x,z}$, which is why $\lambda_u \notin C^{est}(x)$.

Definition 7 (Pairs Causing Imprecision Error). *Let the same preconditions and definition of the imprecision interval for a case pair $\langle x, \lambda_x\rangle, \langle y, \lambda_y\rangle$ as well as the definition of f be given as before. Then, we call*

$$E_{PCI}(sim_\mathcal{X}) = \sum_{\langle x,\lambda_x\rangle\in\mathcal{M}} \sum_{\langle y,\lambda_y\rangle\in\mathcal{M}} \sum_{\substack{\langle u,\lambda_u\rangle\in\mathcal{M}\\ u\neq x}} \left(f(x,y,u) \cdot \min_{\substack{\langle z,\lambda_z\rangle\in\mathcal{M}\\ z\neq y}} g(x,u,z) \right)$$

with $g(x,u,z) = \begin{cases} 0 \text{ if } sim_\mathcal{L}(\lambda_u,\lambda_z) < h_\mathcal{M}(sim_\mathcal{X}(x,z)) \\ 1 \text{ else} \end{cases}$ *the pairs causing imprecision error.*

Assume, we are given a case base $\mathcal{M}$ with $M = |\mathcal{M}|$. Evaluating the fitness of a problem similarity measure $sim_\mathcal{X}$ using one of the functions from Definitions 4 to 7, we have to acknowledge substantial differences in the computational effort required for computing E. First of all, the time complexity of (re-)calculating an entire empirical similarity profile[3] subject to a changed problem similarity measure is quadratic in the number of cases, as can be concluded from [6]. Thus, any fitness evaluation will *at least* have quadratic complexity in M.

Because evaluating E_{B2P} requires just one sweep over m intervals, the complexity does not rise, $E_{B2P} \in O(M^2)$. The same holds for E_{SS2P}, although here an additional sweep over all combinations of cases is required, thus $E_{SS2P} \in O(M^2)$. Counting the number of similarity points that fall into the imprecision interval $I_{x,y}$ for any pair of cases, necessitates another iteration over all cases, such that $E_{PII} \in O(M^3)$. Finally, for E_{PCI} the *min* operator (see Definition 7) must be evaluated. In the worst case, here the complexity of evaluating the inner sum can grow quadratically in the number of cases such that $E_{PCI} \in O(M^4)$, although a practical implementation may ease that by exploiting the fact that the *min* operator does not have to be evaluated when $f(\cdot) = 0$ or that evaluating *min* can be ceased as soon as a $\langle z, \lambda_z\rangle$ with $g(x,u,z) = 0$ has been discovered.

[3] The effort for computing an empirical similarity boundary is the same as for the corresponding profile.

A final remark concerns the strength of an empirical similarity profile (see Section 3.3.2) that may, in accordance to the other error functions, be defined as $E_{STR}(sim_{\mathcal{X}}) = \sum_{i=1}^{m}(1 - h_{\mathcal{M}}(x_i))^2$ with arbitrary $x_i \in A_i$. Note that such an error function will in general not yield maximal precision according to Lemma 2. Nevertheless, we include E_{STR} in our experiments in the next section.

5 Empirical Evaluation

The focus of this evaluation is on a comparison of the performance of CCBI when doing inference utilizing a knowledge-poor default similarity measure sim_{def} (corresponding to the Euclidean distance) and the measures acquired during learning using the different error functions introduced above. All application domains we consider depict regression tasks, i.e. there is a single real-valued solution attribute for which we use a transformation of the Euclidean distance measure as solution similarity measure $sim_{\mathcal{L}}$. In accordance to [6], we measure the performance of CCBI in terms of confidence (share of retrievals with $\lambda_q \in C^{est}(q)$) and imprecision which is the length of the prediction interval (difference of the biggest and smallest element in C^{est}). Further, we provide the average point prediction errors for the respective regression task, where the point prediction of CCBI is determined as the center of the solution interval it predicts.

Note that the imprecision and point prediction of a retrieval result can only be calculated for $q \in \mathcal{X}$ for which $C^{est}(q) \neq \emptyset$. Therefore, we also provide an indication of the share of retrievals during which $C^{est} = \emptyset$ was returned. However, for larger case bases ($|\mathcal{M}| \geq 100$) it generally holds that $Pr(C^{est}(q) = \emptyset) < 0.01$, so that the influence of empty solution sets becomes negligible.

5.1 Proof of Concept

The atomic power plant domain is a small data set covering German nuclear power stations. Since German law dictates the discommisioning of all plants, the task here is to predict the remaining allowed running time of individual stations.

In this experiment, we pursued a leave-one-out validation strategy. Obviously, all error functions suggested are capable of yielding learning improvements regarding the level of precision CCBI achieves (see Figure 5). However, the computational complexity of an error measure seems to heavily correlate to its capabilities in reducing imprecision and the point prediction error. In particular, E_{PCI} reduces the length of the prediction interval represented by C^{est} after 60 evolutionary generations to 0.13 years[4], as opposed to an error of 1.78 years for sim_{def}. Interestingly, the confidence share ($Pr(\lambda_q \in C^{est}(q))$) is not impaired, i.e. stays above the confidence level of the default similarity measure.

5.2 Benchmark Results

Next, we studied the behavior of our learning algorithms on several UCI data benchmark sets. In contrast to the experiments in Section 5.1 (LOO validation),

[4] In 85% of all retrievals performed – in the remaining 15% it holds $C^{est}(q) = \emptyset$.

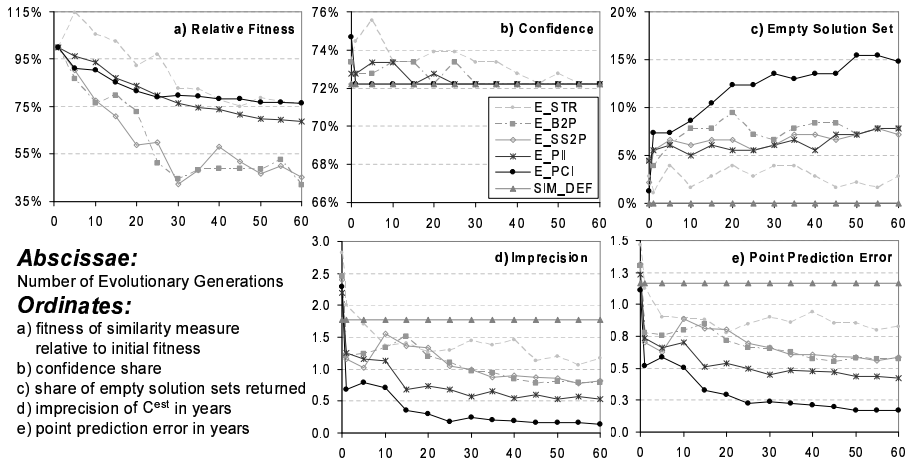

Fig. 5. Results for the Atomic Power Plant Domain (evaluations performed on a LOO basis where the experiments were repeated 10 times, $|\mathcal{M}| = 20$, and $m = 15$).

we now split the case bases, learned on the first part of training cases, and conducted all evaluations of learning results on the remaining part of independent test cases.

The first question of our concern was on the influence of data-sparseness. Here, our findings are in line with [6], revealing that confidence strongly correlates to the size of $\mathcal{M}$. This dependency is even magnified when optimizing the problem similarity measures for increasing precision: Learning with small data sets, not only the imprecision, but also the level of confidence is clearly reduced. This effect is visualized in the top row of Figure 6 where for the *Servo* domain learning curves are shown for optimization processes with 25 and 50 training instances only. The bottom row shows how the situation improves when a more comprehensive training data set is used. Here, it can be concluded that in particular an optimization process using E_{SS2P}, E_{PII}, and E_{PCI} as error function yields excellent precision improvements while confidence stays at a satisfying level. Moreover, the point predictions that CCBI produces using that acquired optimized problem similarity measure clearly outperform the predictions of a k-NN regression ($k = 1, \ldots, 9$).

The results for further benchmark data sets are summarized in Table 1. Since we found that precision-improving similarity measures for CCBI can be reliably obtained for $|CB| \geq 100$, we have omitted the results for smaller training sets. As the orders of magnitude of the solution attributes vary across the domains we considered, we have provided percentual improvements/impairments of the confidence and imprecision levels relative to the corresponding values the default similarity measure yields. It is interesting to note that the changes of the point prediction error are similar to those of the imprecision, which is why the former are omitted in Table 1.

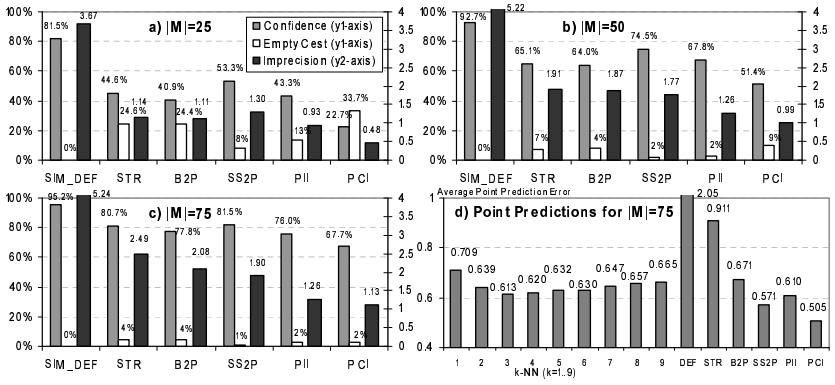

Fig. 6. Improved Precision for the Servo Domain (from UCI Repository).

While most of the results listed are based upon training data set of $|\mathcal{M}| = 200$ (except where noted) for learning on the basis of the pairs causing imprecision error E_{PCI}, we maximally employed 100 training instances, as the enormous computational complexity (cf. Section 4.2) prohibited the use of larger training sets. Consequently, due to effects of overfitting (comparable to, yet not as distinct as in the top row of Figure 6) the results given in the last two columns of the result table are likely to feature comparatively better imprecision and worse confidence levels than the other columns.

Summarizing, we can state that in most of the experiments conducted the gain achieved in reducing imprecision was significantly more distinct than the corresponding reduction of the confidence share. We thus can conclude that the proposed optimization of similarity measures using the error functions derived in Section 4 is highly beneficial for the performance of CCBI. Practically, our evaluation shows that the solution similarity to profile error E_{SS2P} as well as the pairs in imprecision interval error E_{PII} are most suitable for the realization of a precision-oriented similarity measure optimization. Averaged over our experiments they yield a confidence reduction of 11.3/11.8 percent[5] compared to sim_{def} and a simultaneous reduction of imprecision of 43.2/45.6%. The performance of E_{PCI} is evidently superior, but, as mentioned, its computation becomes quickly intractable for increasing amounts of training case data.

6 Conclusion

The contribution of this paper is three-fold. First, we have theoretically examined the notion of precision in the context of credible case-based inference and proved several formal statements concerning the relation between similarity measures and the level of precision inferencing with CCBI may yield. Second, utilizing the

[5] We emphasize that this confidence reduction turns out to be much lower when the amount of training data is further increased (beyond $|\mathcal{M}| = 200$).

Domain Name	Train/ Test Data	E_{STR} Conf	E_{STR} Impr	E_{B2P} Conf	E_{B2P} Impr	E_{SS2P} Conf	E_{SS2P} Impr	E_{PII} Conf	E_{PII} Impr	E_{PCI} Conf	E_{PCI} Impr
Abalone	200/1000	-2.0%	-3.2%	-3.1%	-5.8%	-7.7%	-41.2%	-11.3%	-40.0%	-6.1%	-32.9%
AutoMpg	200/198	-7.0%	+12.5%	-4.3%	+34.3%	-11.3%	-22.9%	-11.9%	-30.5%	-19.8%	-30.5%
Housing	200/306	-6.2%	-18.7%	-0.8%	+0.6%	-11.1%	-39.6%	-7.8%	-48.3%	-22.4%	-51.3%
Liver	200/145	-5.3%	-0.1%	-3.9%	+1.6%	-10.6%	-32.2%	-7.4%	-18.5%	-17.5%	-28.5%
Machines	100/109	-11.6%	+3.0%	-4.1%	+102%	-12.9%	-60.8%	-19.2%	-71.4%	-21.1%	-70.2%
Servo	75/91	-14.1%	-56.3%	-17.0%	-59.2%	-13.3%	-62.7%	-12.9%	-65.1%	-25.6%	-76.7%

Table 1. Results for different benchmark data sets. *Conf* refers to the confidence share and *Impr* to the level of imprecision, i.e. to the length of the predicted solution interval.

theoretical properties of precision in CCBI, we have derived a number of potential error functions that can be employed for tweaking the problem similarity measures CCBI uses towards increased precision. Finally, we have evaluated the proposed optimization approach using several standard benchmark data sets and found that two of the error measures proposed create excellent improvements of the precision when generating candidate solutions with CCBI.

References

1. R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, Addison-Wesley, New York, USA, 1999.
2. A. Bonzano, P. Cunningham, and B. Smyth. Using Introspective Learning to Improve Retrieval in CBR: A Case Study in Air Traffic Control. In *Proceedings of the 2nd International Conference on Case-Based Reasoning*. Springer, 1997.
3. T. Gabel. Learning Similarity Measures: Strategies to Enhance the Optimisation Process. Master thesis, Kaiserslautern University of Technology, 2003.
4. T. Gabel and A. Stahl. Exploiting Background Knowledge when Learning Similarity Measures. In *Proceedings of the 7th European Conference on Case-Based Reasoning*, pages 169–183, Madrid, Spain, 2004. Springer.
5. T. Gabel and M. Veloso. Selecting Heterogeneous Team Players by Case-Based Reasoning: A Case Study in Robotic Soccer Simulation. Technical Report CMU-CS-01-165, Carnegie Mellon University, 2001.
6. E. Hüllermeier. Credible Case-Based Inference Using Similarity Profiles. *IEEE Transactions on Knowledge and Data Engineering*, 19(6):847–858, 2007.
7. A. Stahl. *Learning of Knowledge-Intensive Similarity Measures in Case-Based Reasoning*, volume 986. dissertation.de, 2004.
8. A. Stahl and T. Gabel. Using evolution programs to learn local similarity measures. In *Proceedings of the 5th International Conference on CBR*, pages 537–551, Trondheim, Norway, 2003. Springer.
9. A. Stahl and S. Schmitt. Optimizing Retrieval in CBR by Introducing Solution Similarity. In *Proceedings of the International Conference on Artificial Intelligence (IC-AI'02)*, Las Vegas, USA, 2002. CSREA Press.
10. D. Wettschereck and D. Aha. Weighting Features. In *Proceeding of the 1st International Conference on Case-Based Reasoning*. Springer, 1995.
11. Z. Zhang and Q. Yang. Dynamic Refinement of Feature Weights Using Quantitative Introspective Learning. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, 1999.