Some Perspectives on Similarity Learning for Case-Based Reasoning and Analogical Transfer

Fadi Badra[1,*,†], Marie-Jeanne Lesot[2,†], Esteban Marquer[3,†] and Miguel Couceiro[3,†]

[1]Université Sorbonne Paris Nord, Laboratoire d'Informatique Médicale et d'Ingénierie des Connaissances en e-Santé, LIMICS, Sorbonne Université, INSERM, F-93000, Bobigny, France

[2]Sorbonne Université, CNRS, LIP6, F-75005 Paris, France

[3] University of Lorraine, CNRS, Loria, Nancy, France

Abstract

In this paper we investigate interactions between recent advances in the modeling of analogical transfer and similarity learning. Indeed, a unifying principle of case-based prediction methods was recently established, according to which the plausible inference principle of analogical transfer can be interpreted as a transfer of similarity knowledge from a situation space to an outcome space. Following this principle, the task of analogical transfer can be addressed using a global indicator of the compatibility between two similarity measures. Such an indicator can also be used to assess the quality of the situation space similarity measure with respect to the case-based prediction task. We discuss several perspectives opened by such an interpretation of the task of analogical transfer as the optimisation of the compatibility criterion: we explore interactions with similarity learning, as well as with energy function optimisation.

Keywords

Case-Based Reasoning, Analogical transfer, Similarity learning, Quality measure

1. Introduction

Analogical transfer is a cognitive process that allows to derive some new information about a target situation by applying a plausible inference principle, according to which if two situations are similar with respect to some criteria, then it is plausible that they are also similar with respect to other criteria [1]. Case-based reasoning (CBR) systems implement analogical transfer in order to infer some information about a new situation directly by comparing it to a set of past experiences (called cases) stored in memory [2]. In that process, similarity knowledge is a critical component and is dependent on the task and data considered. For instance, several approaches have been proposed to measure similarities between data represented as Boolean vectors and between sequences in the context of analogical reasoning, as described in [3].

IARML@IJCAI'2023: Workshop on the Interactions between Analogical Reasoning and Machine Learning, at IJCAI'2023, August, 2023, Macao, China

badra@sorbonne-paris-nord.fr (F. Badra); Marie-Jeanne.Lesot@lip6.fr (M. Lesot); esteban.marquer@loria.fr (E. Marquer); miguel.couceiro@loria.fr (M. Couceiro)

https://limics.fr (F. Badra); https://lip6.fr (M. Lesot); https://emarquer.github.io/ (E. Marquer); https://members.loria.fr/mcouceiro/ (M. Couceiro)

0000-0002-2437-8230 (F. Badra); 0000-0002-3604-6647 (M. Lesot); 0000-0003-2315-7732 (E. Marquer); 0000-0003-2316-7623 (M. Couceiro)

Recent work [4] showed that a common principle underlying case-based prediction methods is that they interpret the plausible inference principle of analogical transfer as a transfer of similarity knowledge from a situation space to an outcome space. This idea of modeling analogical transfer as a transfer of similarity knowledge is a powerful idea, that can have many implications. One of them is that learning a similarity measure can be framed as the problem of optimizing the compatibility between two similarity measures on a data set.

In this paper, we discuss some perspectives and directions that could be given to this line of research. A global indicator of the compatibility between two similarity measures has already been proposed in the CoAT method [5], and preliminary experiments showed that such an indicator can be used as an intrinsic indicator of the quality of the similarity measure with respect to the case-based prediction task [6]. A natural perspective to this research is to apply these results to similarity learning, and to design a similarity learning method that would optimise such an indicator on the data set. To this aim, we explore in this paper the connections between the CoAT method and existing work in the domain of similarity learning. We then show that interpreting CoAT in an energy-based model is quite straightforward, so that the similarity learning task can be stated as the task of learning an energy function.

The paper is organized as follows. In Section 2 we recall the previous work on the CoAT method. We then briefly survey in Section 3 some approaches to learning (dis)similarities that seem relevant to CoAT, and discuss to how to leverage CoAT to obtain suitable similarity measures. We also explore techniques based on the optimisation of energy function that we propose in Section 4 and discuss further perspectives in Section 5.

2. The CoAT Method

In the CoAT method [5, 6, 7], the analogical transfer inference is made by minimizing a global indicator of compatibility between two similarity measures. Such an indicator can also be used as an intrinsic indicator of the quality of the similarity measure w.r.t. the transfer task.

2.1. Definition of the Indicator

Let $\mathcal{S}$ denote an input space, and $\mathcal{R}$ an output space. An element of $\mathcal{S}$ is called a *situation*, and an element of $\mathcal{R}$ is called an *outcome*, or a result. A set $CB = \{(s_1, r_1), \ldots, (s_n, r_n)\}$ of elements in $\mathcal{S} \times \mathcal{R}$ is called a *case base*. An element $c = (s, r) \in CB$ is called a *source case*. In addition, the spaces $\mathcal{S}$ and $\mathcal{R}$ are respectively equipped with two similarity measures $\sigma_{\mathcal{S}}$ and $\sigma_{\mathcal{R}}$, that respectively denote the similarity measure on situations and on outcomes.

The compatibility of $\sigma_{\mathcal{R}}$ with $\sigma_{\mathcal{S}}$ is measured globally on the case base CB, by introducing a global indicator $\Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB)$. This indicator measures the compatibility of $\sigma_{\mathcal{R}}$ with $\sigma_{\mathcal{S}}$ from an ordinal point of view on the whole case base CB, by checking if the order induced by $\sigma_{\mathcal{R}}$is the same as the one induced by $\sigma_{\mathcal{S}}$. The following continuity constraint is tested on each triple of cases (c_0, c_i, c_j), with $c_0 = (s_0, r_0)$, $c_i = (s_i, r_i)$, and $c_j = (s_j, r_j)$:

$$\text{if } \sigma_{\mathcal{S}}(s_0, s_i) \geq \sigma_{\mathcal{S}}(s_0, s_j), \text{ then } \sigma_{\mathcal{R}}(r_0, r_i) \geq \sigma_{\mathcal{R}}(r_0, r_j). \qquad (C)$$

Constraint (C) expresses that anytime a situation s_i is more similar to a situation s_0 than situation s_j, this order should be preserved on outcomes. A triple (c_0, c_i, c_j) does *not* satisfy (C) if the

case c_i is more similar to the case c_0 (that we will refer to as *anchor*) than the case c_j for situations, but less similar for outcomes, *i.e.*, when $\sigma_{\mathcal{S}}(s_0, s_i) \geq \sigma_{\mathcal{S}}(s_0, s_j)$ and $\sigma_{\mathcal{R}}(r_0, r_i) < \sigma_{\mathcal{R}}(r_0, r_j)$. Such a violation of the constraint is called an *inversion of similarity*. The indicator $\Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB)$ counts the total number of inversions of similarity observed on a case base CB:

$$\Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB) = |\{((s_0, r_0), (s_i, r_i), (s_j, r_j)) \in CB \times CB \times CB \text{ such that } \sigma_{\mathcal{S}}(s_0, s_i) \geq \sigma_{\mathcal{S}}(s_0, s_j) \text{ and } \sigma_{\mathcal{R}}(r_0, r_i) < \sigma_{\mathcal{R}}(r_0, r_j)\}|.$$

2.2. Inference

When the case base is fully known, except for the outcome r_t of one case $c_t = (s_t, r_t)$, the transfer inference consists in finding the outcome r_t that minimizes the value of the indicator:

$$r_t = \arg\min_{r \in \mathcal{R}} \Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB \cup \{(s_t, r)\}).$$

2.3. An Intrinsic Indicator of the Quality of a Similarity Measure

The indicator $\Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB)$ can be used to assess the quality of the situation space similarity measure $\sigma_{\mathcal{S}}$ with respect to the transfer task, independently of the algorithm used for the inference. We report here some first experiments made in [6] that show a strong correlation between the value of the $\Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB)$ indicator obtained for a chosen similarity measure $\sigma_{\mathcal{S}}$ and the corresponding performance of the CoAT prediction algorithm.

Experimental Protocol. The experiment is conducted on 200 instances extracted from the Balance Scale data set[1]. As the instances of these data sets are described only by d numeric features, each situation can be represented by a vector of $\mathbb{R}^d$. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ be two such vectors. These data induce a classification task: the outcomes are categorical classes and the outcome similarity measure $\sigma_{\mathcal{R}}$ is the class membership, i.e. $\sigma_{\mathcal{R}}(u, v) = 1$ if $u = v$, and 0 otherwise. The performance of the CoAT algorithm is measured by generating 100 different classification tasks $\{(\sigma_i, \sigma_{\mathcal{R}}, CB)\}_{1 \leq i \leq 100}$, each of which is obtained by choosing for $\sigma_{\mathcal{S}}$ a decreasing function of a randomly weighted Euclidean distance. More precisely, a set of random linear maps $\{L_i : \mathbb{R}^d \longrightarrow \mathbb{R}^d\}_{1 \leq i \leq 100}$ are generated, and for each map L_i, σ_i is defined as a decreasing function of the Euclidean distance computed in the L_i's embedding space:

$$\sigma_i(\mathbf{x}, \mathbf{y}) = e^{-d_i(\mathbf{x}, \mathbf{y})} \text{ with } d_i(\mathbf{x}, \mathbf{y}) = \|L_i\mathbf{x} - L_i\mathbf{y}\|_2 = \sqrt{(\mathbf{x} - \mathbf{y})^T L_i^T L_i (\mathbf{x} - \mathbf{y})}.$$

The performance is also measured on the task $(\sigma_E, \sigma_{\mathcal{R}}, CB)$, in which $\sigma_E(\mathbf{x}, \mathbf{y}) = e^{-\|\mathbf{x} - \mathbf{y}\|_2}$ is a decreasing function of the Euclidean distance, which amounts to taking as linear map the identity matrix. For each task, the performance is measured by the prediction accuracy, with 10-fold cross validation.

[1] https://archive.ics.uci.edu/ml/datasets/balance+scale

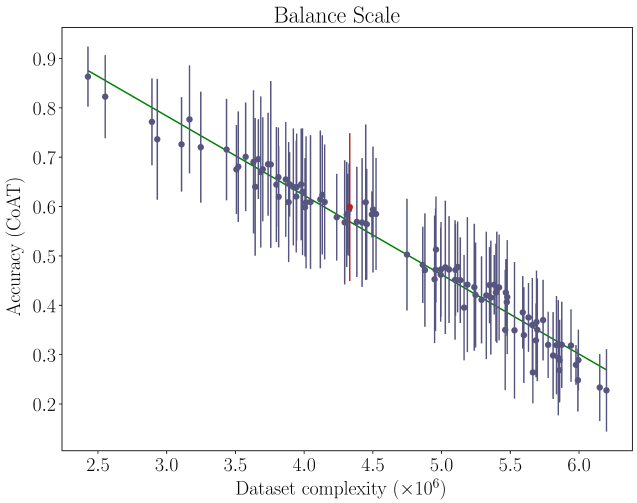

Figure 1: Relation between CoAT performance (accuracy) and value of the Γ indicator on the Balance Scale data set (as of [6]).

Results. Fig. 1 shows for each classification task the average accuracy and standard deviation of the CoAT algorithm according to the value of the Γ indicator ("Dataset complexity" axis on the figure). The blue points correspond to the randomly generated σ_i similarity measures. The red point gives the results for the σ_E similarity measure based on the standard Euclidean distance. The green line shows the result of a linear regression on the data. The Pearson's coefficient is -0.97. The results clearly show a correlation between the value of the indicator and the performance of the CoAT algorithm.

3. Perspectives on Learning (Dis)similarity Measures

While it is possible to use CoAT to quantify the suitability of a similarity measure for a CBR task, we argue that it should be possible to adapt CoAT to learn suitable similarity measures. Below we describe some existing methods to learning similarity (or dissimilarity) that appear relevant to adapt CoAT, before discussing how optimizing the indicator of CoAT relates to these (dis)similarity measure learning methodologies.

In what follows, we will not make distinction between similarity and dissimilarity measures since they are the counterpart of one another. It is possible to define one from the other, for instance, given a dissimilarity $d(u, v)$ defined on $\mathbb{R}^+$ we define the similarity $\sigma(u, v)$ on $[0, 1]$ with the inverse $\sigma(u, v) = \frac{1}{1+d(u,v)}$ or the exponential $\sigma(u, v) = e^{-d(u,v)}$.

3.1. Related Works on (Dis)similarity Measure Learning

Constructing a similarity measure for a given task is difficult and time-consuming, especially if domain knowledge is to be taken into account into the process. It is possible to use data to

support and facilitate this process, either to guide the design of the measure [8] or to learn suitable parameters for a similarity measure.

Designing or learning (dis)similarity measures from data has long been studied [9]. Here, we briefly discuss three approaches, namely, by combining local similarities, by unsupervised approaches based on clustering techniques, and by supervised or semi-supervised metric learning approaches. Note that there is a particular focus in CBR on the explainability of the similarity measures as well as on using complex data (*i.e.*, heterogeneous or structured), which constrains the learning of (dis)similarity measures.

Combining local (dis)similarities. Computing (dis)similarities in heterogeneous data can be performed by transforming the input dataset into a homogeneous one. An interesting approach is to consider the overall similarity measure as a weighted sum of ad-hoc measures. For instance, the *k-Prototypes* algorithm [10] computes a dissimilarity $d(x, y)$ between two instances x and y as

$$d(x, y) = d_E(x, y) + \lambda d_C(x, y), \tag{1}$$

where $d_E(x, y)$ is the Euclidean distance for a subset of continuous attributes, $d_C(x, y)$ the number of mismatched categorical attributes, and where λ a weighting parameter. Gower's similarity [11] is a popular measure that works in a similar fashion.

More generally, it is possible to rely on existing similarity measures for each aspect of the data, and combine them to obtain a global similarity. For instance, [8, 12, 13] learn the weights of linear combinations of local similarity functions for CBR tasks. Another example is [14], in which a set of local similarities estimated by artificial neural networks are aggregated. Note that the above mentioned weights can be thought as the importance that each local measure has, and thus used for explanation and fairness purposes [14, 15, 16, 17, 18, 19].

The main drawback of combining local dissimilarities is that it requires additional preprocessing and learning as well as supervision.

Unsupervised learning of (dis)similarities. Shi and Horvath [20] proposed a method to compute dissimilarities between instances in unsupervised settings using Random Forest (RF). RF [21] is a popular algorithm for supervised learning tasks, and is widely used in many applied fields, *e.g.,* in biology [22] and in image recognition [23]. Essentially, it is an ensemble method that combines decision trees in order to obtain better classification results in supervised learning on high-dimensional data.

The algorithm begins by creating several new training sets, each one being a bootstrap sample of elements from the initial data set X. A decision tree is built on each training set, using a random sample of m_{try} features at each split. The prediction task is then performed by a majority vote or by averaging the results of the decision trees, according to the problem at hand (classification or regression). This approach leads to better accuracy and generalization capacity of the model compared to single decision trees, while reducing the variance [24]. However, this ensemble approach requires labelled data.

The adaptation of RF to unsupervised settings was made possible by the generation of synthetic instances, that enable a binary classification between the latter and the observed (unlabelled) instances. The use of Unsupervised Random Forest (URF) for measuring (dis)similarity

presents several advantages. For instance, instances described by mixed types of variables as well as missing values can be handled. In fact, this method has been successfully used in many applications [25, 26, 27, 28].

Albeit its appealing character, the method suffers from two main drawbacks. Firstly, the generation step is not computationally efficient: since the obtained trees highly depend on the generated instances, it is necessary to construct many forests with different synthetic instances and average their results, leading to a computational burden. Secondly, the synthetic instances may bias the model being constructed to discriminate instances on specific features.

More recently, Ting *et al.* [29] proposed a similar approach to compute a mass-based dissimilarity between instances, based on isolation forests [30]. While their approach is similar, it differs on some key points, such as the fact that self-similarities are not constant in mass-based dissimilarity, since it they depend on the distribution of the data. This property is interesting and may lead to good results in cases where clusters are of varying density. However, this method does not apply to heterogeneous data.

Following the tracks of [20] and [31], [32] proposed a method, called Unsupervised Extremely Randomised Trees (UET), to compute similarities on unlabelled data. The main idea is to randomly split the data in an iterative fashion until a stopping criterion is met, and to compute a similarity based on the co-occurrence of instances in the leaves of each generated tree. It was shown to provide tailor made multidimensional similarity measures for complex and heterogeneous data [33] and to be easily adaptable to structured data such as labelled graphs [34]. The empirical study of UET showed that it outperforms existing methods (such as URF) in terms of computational time, while giving better cluster results and, consequently, more relevant similarities. Moreover, it has interesting invariance properties such as such as invariance under monotonic transformations of variables and robustness to correlated variables and noise, that drastically reduces preprocessing.

Despite of producing tailor made measures for data at hand, the main drawback of UET is that it computes similarities on each space (the situation and outcome) without establishing links between the two.

Metric learning. Learning dissimilarity measures from data has been tackled in the field of metric learning (for an extended introduction, see [35, 36]) by learning the parameters of parametric distance functions d_θ, following either relative (ordinal) constraints or link/cannot link (similarity/dissimilarity) constraints. Metric learning techniques have been used for representation learning: combining a parametric representation model with a simple non-parametric distance function (typically the Euclidean distance) allows to learn a representation model suitable to preserve the relative or link/cannot link constraints. These constraints are usually implemented by minimizing the triplet loss or the contrastive loss as follows.

On the one hand, contrastive loss [37] is used to enforce link/cannot link constraints on training pairs s_i, s_j associated with labels r_i, r_j. If s_i, s_j, associated with labels r_i, r_j, is a pair of similar elements ($r_i \approx r_j$), then we want to minimize $d_\theta(s_i, s_j)$, and we want to maximize the latter if the pair is not similar ($r_i \neq r_j$). The contrastive loss is defined as

$$L(s_i, s_j) = \sigma_{\mathcal{R}}(r_i, r_j) d_\theta(s_i, s_j) - (1 - \sigma_{\mathcal{R}}(r_i, r_j)) d_\theta(s_i, s_j),$$

where $\sigma_{\mathcal{R}}$ is the already mentioned class membership similarity measure, such that $\sigma_{\mathcal{R}}(u, v) = 1$ if $u = v$, and 0 otherwise.

On the other hand, triplet loss [38, 39] methods use training triplets s_0, s_i, s_j associated with labels r_0, r_i, r_j. that are selected such that r_0 (called the *anchor* as in CoAT) is closer to r_i than r_j. For such triplets, it is desired that $d_\theta(s_0, s_i) < d_\theta(s_0, s_j)$ which translates into the triplet loss

$$L(s_0, s_i, s_j) = \max(d_\theta(s_0, s_i) - d_\theta(s_0, s_j) + \alpha, 0)$$

where the *margin* α is used to enforce a gap between the clusters of situations.

To implement relative constraints with triplet loss, it is enough to have r_0, r_i, r_j verify an ordinal relation of the form $r_0 \leq r_i < r_j$ or $r_0 \geq r_i > r_j$. In a classification setting, the labels are classes that do not necessarily have an order defined, so the link/cannot link constraint $r_0 = r_i \neq r_j$ is used instead. This latter constraint corresponds to $\sigma_{\mathcal{R}}(r_0, r_i) < \sigma_{\mathcal{R}}(r_0, r_j)$, where $\sigma_{\mathcal{R}}$ is the class membership similarity measure mentioned above.

Note that while metric learning was initially designed to use class labels, making it a supervised methodology, semi-supervised and unsupervised variants have been also proposed [40, 41].

3.2. Links Between CoAT and Metric Learning Approaches

The Γ indicator defined in Section 2.1 measures how suitable a similarity measure is for a particular CBR task. As such, it could be used to identify or, following metric learning methodology, to learn a similarity measure or a suitable representation space. To help make such a parallel, we propose to leverage striking similarities between CoAT and triplet loss.

Indeed, as in triplet loss methods, the CoAT method considers similarity judgements that are data triplets of the form $\{(s_0, s_i, s_j) \mid \sigma_{\mathcal{R}}(r_0, r_i) < \sigma_{\mathcal{R}}(r_0, r_j)\}$, but then counts the number of triplets violating the constraint (C), *i.e.*, such that $\sigma_{\mathcal{S}}(s_0, s_i) \geq \sigma_{\mathcal{S}}(s_0, s_j)$ and $\sigma_{\mathcal{R}}(r_0, r_i) < \sigma_{\mathcal{R}}(r_0, r_j)$. In triplet loss terminology, this corresponds to counting the number of hard negatives among all possible triplets formed with instances of the data set. Semi-hard negatives (*i.e.*, triplets such that $\sigma_{\mathcal{S}}(s_0, s_i) + \alpha \geq \sigma_{\mathcal{S}}(s_0, s_j)$ for some margin α) are excluded from this procedure. Therefore, when applied to classification settings, the contribution of a triplet to the CoAT indicator Γ can be seen as a simplified version of the loss $L(s_0, s_i, s_j)$ used in triplet loss methods, that would take value 1 if the triplet is a hard negative, and 0 otherwise.

However, the idea of the CoAT method is to sum up these contributions on all possible triplets of a case base. Although in our first experiments, the case base consisted in the whole data set, a more case-based approach would require crafting a (preferably small but informative) case base for the task before attempting to learn a similarity measure. Moreover, one contribution of the work done on the CoAT method has been to show that the prediction for a new case depends only on the *new* similarity relations that result from the addition of the new case to the case base [6]. This suggests that learning should be done by carefully selecting a case base from whole data set, and training for a test case (t, r) by minimizing

$$\Delta\Gamma(t, r, \sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB) = \Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB \cup \{(t, r)\}) - \Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB).$$

This could lead to giving additional theoretical justification of triplet loss methods and give new insights on how to solve the sampling issue (*i.e.*, which training triplets to select).

4. Perspectives on Learning an Energy Function

This section discusses another perspective opened by the analogical inference interpretation as the optimisation of the proposed Γ indicator, as established in Section 2.2. Indeed, this view allows to exploit the formalism of energy-based models proposed for machine learning tasks by [42] reminded below. As detailed in the following, the interpretation of CoAT in an energy-based model is quite straightforward: the global indicator Γ of the CoAT approach can be seen as an energy function, that measures the compatibility between two similarity measures $\sigma_{\mathcal{S}}$ and $\sigma_{\mathcal{R}}$ on the case base CB. In this perspective, CoAT's transfer strategy is an energy-based inference, that consists in completing the description of the case base in order to minimize its energy, and learning the energy function (and hence, the similarity measure) could be achieved by optimizing a contrastive loss function.

Energy-Based Models. Inspired from statistical physics, energy-based models specify a probability distribution

$$p(x;\theta) = \frac{e^{-E_\theta(x)/T}}{\int e^{-E_\theta(x)/T} dx}$$

directly via a parameterized scalar-valued function $E_\theta(x)$ called an *energy function*. In machine learning, energy-based models are trained to be optimized on the data manifold: the energy function is learned to give low values to training data, and higher values to data points that are far from the data manifold [42]. In its conditional version, the definition of an energy function $E_\theta : \mathcal{X} \times \mathcal{Y} \longrightarrow \mathbb{R}$ assumes the existence of an input space $\mathcal{X}$, an output space $\mathcal{Y}$, and a set of parameters θ. The energy function E_θ associates to each pair $(x, y) \in \mathcal{X} \times \mathcal{Y}$ a scalar value $E_\theta(x, y)$ that represents the compatibility between the input x and the output y under the set of parameters θ. The energy function E_θ takes low values when y is compatible with x, and higher values when y and x are less compatible. The goal of the energy-based *inference* is to find, among a set of outputs $\mathcal{Y}$, the output $y^* \in \mathcal{Y}$ that minimizes the value of the energy function:

$$y^* = \arg\min_{y \in \mathcal{Y}} E_\theta(x, y).$$

Given a family of energy functions $E_\theta(x, y)$ indexed by a set of parameters θ, the goal of *learning* is to optimize the θ parameters in order to "push down" (*i.e.*, assign lower energy values to) the points on the energy surface that are around the training samples, and to "pull up" all other points. Contrastive divergence [43] is a common learning strategy that consists in optimizing a contrastive loss function such as the hinge loss, which is defined, for a training sample (x_k, y_k) and a generated out of distribution sample $(x_k, \hat{y})$ by:

$$\ell(\theta, x_k, y_k) = \max(0, \beta + E_\theta(x_k, y_k) - E_\theta(x_k, \hat{y})).$$

The hinge loss associates a loss value to a training sample (x_k, y_k) whenever its energy is not lower by at least a margin β than the energy of the incorrect sample $(x_k, \hat{y})$.

An Energy-Based Model of Analogical Transfer. The input space $\mathcal{X}$ (from which similarity knowledge is transferred) is the situation space $\mathcal{S}$. The output space $\mathcal{Y}$ (to which similarity knowledge is transferred) is the outcome space $\mathcal{R}$. The situation space $\mathcal{S}$ is equipped with a similarity measure $\sigma_{\mathcal{S}}$, and the outcome space is equipped with a similarity measure $\sigma_{\mathcal{R}}$. The energy function $E_\theta : \mathcal{S} \times \mathcal{R} \longrightarrow \mathbb{R}$ measures the compatibility of the outcome similarities with the added situation similarities when a potential new case $\hat{c}_t = (t, r)$ is added to the case base. The energy function E_θ is parameterized by a hyperparameter $\theta = (\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB)$, which includes the case base CB. Indeed, assuming that $\sigma_{\mathcal{S}}$ and $\sigma_{\mathcal{R}}$ are defined on different sets of attributes, the compatibility between two similarity measures can not be evaluated *per se*, but only relatively to a given set of case pairs. For a new situation t, the goal of the energy-based *inference* is to find, among a set of potential outcomes $r \in \mathcal{R}$, the outcome r_t that minimizes the value of the energy function:

$$r_t = \arg\min_{r \in \mathcal{R}} E_\theta(t, r).$$

Among the three parameters of $\theta = (\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB)$, the case base CB and the outcome similarity measures $\sigma_{\mathcal{R}}$ are usually fixed, so that learning θ amounts to learning the situation similarity measure $\sigma_{\mathcal{S}}$ for the task at hand. This can be done by contrastive divergence using the hinge loss defined as follows: for a training sample $(s_k, r_k) \in \mathcal{S} \times \mathcal{R}$ and a chosen outcome $\hat{r} \in \mathcal{R}$,

$$\ell(\theta, s_k, r_k) = \max(0, m + E_\theta(s_k, r_k) - E_\theta(s_k, \hat{r})).$$

The CoAT case-based prediction method directly implements this energy-based model by taking as energy function the global indicator Γ:

$$E_\theta^{\text{CoAT}}(t, r) = \Gamma(\sigma_{\mathcal{S}}, \sigma_{\mathcal{R}}, CB \cup \{(t, r)\}).$$

Illustration on Some Synthetic Data Sets. Fig. 2 gives some examples of energy maps that are obtained for different synthetic data sets on a binary classification task. On each figure, the dataset size is the same ($|CB| = 100$), but the instances span differently on the 2D description space. The instances are equally split into two classes (orange and blue). The similarity measure on situations $\sigma_{\mathcal{S}}$ is a decreasing function of the Euclidean distance as in Sec. 3 (*i.e.*, $\sigma_{\mathcal{S}} = \sigma_E$), except for Fig. 2 d, where $\sigma_{\mathcal{S}}$ is constructed from a linear transformation of the Euclidean distance, by choosing from a set of 100 randomly generated transformations, the one that minimizes the energy of the case base. Let us denote by σ^* the resulting similarity measure. The similarity measure on outcomes $\sigma_{\mathcal{R}}$ represents class membership as previously.On the figures, the colors indicate for each point of space the class that would be predicted by the CoAT algorithm : green for the blue class, and orange for the orange class. The color saturation is proportional to the difference between the energy of the predicted class and the energy of the other class.

Results In Fig. 2 a, the two classes are well separated, and no instance is more similar to an instance of a different class than it is to an instance of the same class, hence, $E(\sigma_E, \sigma_{\mathcal{R}}, CB) = 0$. In Fig. 2 b, the two classes are closer, and even overlap, and some inter-class similarities

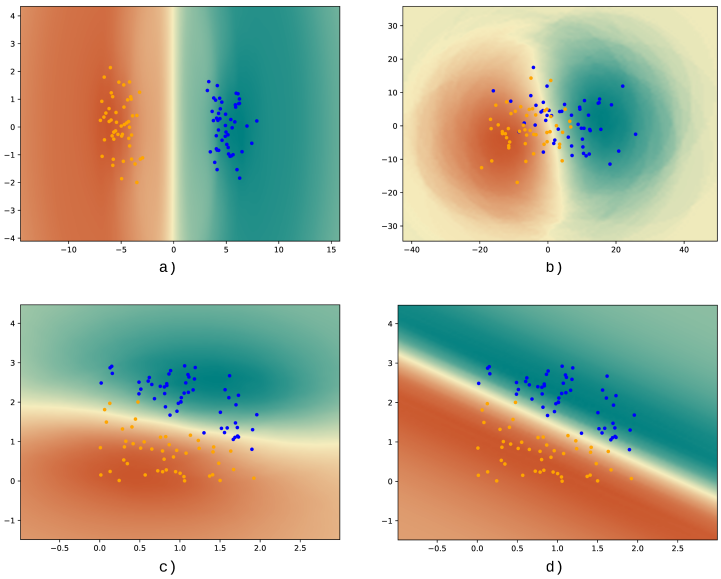

Figure 2: Energy maps illustrating the confidence values associated to each class by the CoAT algorithm for different synthetic datasets in a binary classification scenario. Green areas correspond to areas where new instances would be predicted as belonging to the blue class, and red areas correspond to areas where new instances would be predicted as belonging to the orange class. The color saturation is proportional to the difference between the energy of the predicted class and the energy of the other class. On figures (a), (b), and (c), $\sigma_{\mathcal{S}}$ is constructed from the Euclidean distance. On the lower right figure (d), $\sigma_{\mathcal{S}}$ is optimized to minimize the energy of the case base.

happen to be lower than some intra-class similarities, leading to the non-zero data set energy $E(\sigma_E, \sigma_{\mathcal{R}}, CB) = 86,786$. Fig. 2 c and d show a data set with two linearly separable classes. In Fig. 2 c, $\sigma_{\mathcal{S}}$ is set to the (inverse of) the Euclidean distance, which leads to sub-optimal prediction performance: the prediction frontier does not correspond to the real class frontier, and some instances are misclassified. The energy of the case base is $E(\sigma_E, \sigma_{\mathcal{R}}, CB) = 43,264$. In Fig. 2 d, the similarity measure $\sigma_{\mathcal{S}}$ is optimized by choosing a similarity measure σ^* that minimizes the energy of the case base. The resulting prediction performance is improved: the prediction frontier corresponds to the real class frontier, and no instance of the case base are misclassified. The energy of the case base is $E(\sigma^*, \sigma_{\mathcal{R}}, CB) = 17,146$.

5. Conclusion

In this paper we investigated interactions between analogical transfer and similarity learning, in the framework of CoAT. In particular, we identified similarities between the Γ indicator and the triplet loss of metric learning, that may be used to obtain suitable similarities for analogical transfer. We also proposed an interpretation of the CoAT method in the formalism of energy-based models, so that the similarity learning task can be expressed as the task of learning an

energy function.

The established connections allow to envision other applications. For instance, it could be used for case base construction and maintenance. Indeed, if we consider the indicator as an energy function, the competence of a case should relate to its ability, when it is added to the case base, to lower the energy of other cases. Reasoning with a small but competent case base would solve one of the actual limitations of the CoAT method, which is the quadratic computational complexity of the inference procedure.

An additional direction for future works concerns the integration of expert knowledge, to promote interaction with domain experts when processing a case base. We envision this integration at two levels: the design of the similarity measure and the choice of suitable cases. We envision a semi-automatic approach to reach a suitable compromise between available data, expert input, and selection of competent cases.

References

[1] T. R. Davies, S. J. Russell, A logical approach to reasoning by analogy, in: IJCAI, 1987. arXiv:1011.1669v3.

[2] A. Aamodt, E. Plaza, Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, AI Communications 7 (1994) 39–59.

[3] L. Miclet, S. Bayoudh, A. Delhay, Analogical Dissimilarity: Definition, Algorithms and Two Experiments in Machine Learning, JAIR 32 (2008) 793–824.

[4] F. Badra, M.-J. Lesot, Case-Based Prediction – A Survey, IJAR (2023).

[5] F. Badra, A Dataset Complexity Measure for Analogical Transfer, in: IJCAI, 2020, pp. 1601–1607.

[6] F. Badra, M.-J. Lesot, Theoretical and Experimental Study of a Complexity Measure for Analogical Transfer, in: ICCBR, 2022, pp. 175–189.

[7] F. Badra, M.-J. Lesot, CoAT-APC: When Analogical Proportion-based Classification Meets Case-Based Prediction, in: ATA@ICCBR, CEUR-WS, 2022.

[8] D. Verma, K. Bach, P. J. Mork, Similarity Measure Development for Case-Based Reasoning–A Data-Driven Approach, in: NAIS, volume 1056, Springer, Cham, 2019, pp. 143–148.

[9] M. M. Deza, E. Deza, Encyclopedia of distances, in: Encyclopedia of Distances, Springer, 2009, pp. 1–583.

[10] Z. Huang, Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values, Data Mining and Knowledge Discovery 2 (1998) 283–304.

[11] J. Gower, A general coefficient of similarity and some of its properties, Biometrics (1971) 857–871.

[12] W. Cheng, E. Hüllermeier, Learning Similarity Functions from Qualitative Feedback, in: ECCBR, volume 5239, Springer, 2008, pp. 120–134.

[13] A. Jaiswal, K. Bach, A Data-Driven Approach for Determining Weights in Global Similarity Functions, in: ICCBR, volume 11680, Springer International Publishing, 2019, pp. 125–139.

[14] T. Gabel, E. Godehardt, Top-Down Induction of Similarity Measures Using Similarity Clouds, in: ICCBR, volume 9343, Springer, Cham, 2015, pp. 149–164.

[15] M. T. Ribeiro, S. Singh, C. Guestrin, "why should I trust you?": Explaining the predictions of any classifier, in: 22nd SIGKDD, ACM, 2016, pp. 1135–1144.
[16] S. M. Lundberg, S. Lee, A unified approach to interpreting model predictions, in: NIPS 2017, 2017, pp. 4765–4774.
[17] G. Alves, M. Amblard, F. Bernier, M. Couceiro, A. Napoli, Reducing unintended bias of ML models on tabular and textual data, in: 8th DSAA, IEEE, 2021, pp. 1–10.
[18] J.-B. Lamy, B. Sekar, G. Guezennec, J. Bouaud, B. Séroussi, Explainable artificial intelligence for breast cancer: A visual case-based reasoning approach, Artificial Intelligence in Medicine 94 (2019) 42–53.
[19] K. Bach, P. J. Mork, On the Explanation of Similarity for Developing and Deploying CBR Systems, in: AAAI, 2020.
[20] T. Shi, S. Horvath, Unsupervised learning with random forest predictors, Journal of Computational and Graphical Statistics 15 (2006) 118–138.
[21] L. Breiman, Random forests, Machine learning 45 (2001) 5–32.
[22] B. Percha, Y. Garten, R. B. Altman, Discovery and explanation of drug-drug interactions via text mining, in: PSB, 2012, pp. 410–421.
[23] M. Pal, Random forest classifier for remote sensing classification, Int. J. Remote Sensing 26 (2005) 217–222.
[24] J. Friedman, T. Hastie, R. Tibshirani, The elements of statistical learning, volume 1, Springer series in statistics New York, 2001.
[25] H. L. Kim, D. Seligson, X. Liu, N. Janzen, M. Bui, H. Yu, T. Shi, A. S. Belldegrun, S. Horvath, R. Figlin, Using tumor markers to predict the survival of patients with metastatic renal cell carcinoma, The Journal of urology 173 (2005) 1496–1501.
[26] M. Abba, H. Sun, K. Hawkins, J. Drake, Y. Hu, M. Nunez, S. Gaddis, T. Shi, S. Horvath, A. Sahin, *et al.*, Breast cancer molecular signatures as determined by sage: correlation with lymph node status, Molecular Cancer Research 5 (2007) 881–890.
[27] S. Rennard, N. Locantore, B. Delafont, R. Tal-Singer, E. Silverman, J. Vestbo, B. Miller, P. Bakke, B. Celli, P. Calverley, *et al.*, Identification of five chronic obstructive pulmonary disease subgroups with different prognoses in the eclipse cohort using cluster analysis, Annals of the American Thoracic Society 12 (2015) 303–312.
[28] K. Peerbhay, O. Mutanga, R. Ismail, Random forests unsupervised classification: The detection and mapping of solanum mauritianum infestations in plantation forestry using hyperspectral data, IEEE J. Sel. Top. Appl. Earth Obs Remote Sens 8 (2015) 3107–3122.
[29] K. Ting, Y. Zhu, M. Carman, Y. Zhu, T. Washio, Z. Zhou, Lowest probability mass neighbour algorithms: Relaxing the metric constraint in distance-based neighbourhood algorithms, Machine Learning (2018).
[30] F. Liu, K. Ting, Z. Zhou, Isolation forest, in: 8th ICDM, IEEE, 2008, pp. 413–422.
[31] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, Machine learning 63 (2006) 3–42.
[32] K. Dalleau, M. Couceiro, M. Smaïl-Tabbone, Unsupervised extremely randomized trees, in: 22nd PAKDD 2018, volume 10939 of *LNCS*, Springer, 2018, pp. 478–489. URL: https://doi.org/10.1007/978-3-319-93040-4_38. doi:10.1007/978-3-319-93040-4_38.
[33] K. Dalleau, M. Couceiro, M. Smaïl-Tabbone, Unsupervised extra trees: a stochastic approach to compute similarities in heterogeneous data, Int. J. Data Sci. Anal. 9 (2020) 447–459.

[34] K. Dalleau, M. Couceiro, M. Smaïl-Tabbone, Computing vertex-vertex dissimilarities using random trees: Application to clustering in graphs, in: 18th IDA, volume 12080 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 132–144.

[35] A. Bellet, A. Habrard, M. Sebban, A Survey on Metric Learning for Feature Vectors and Structured Data (2014). arXiv:1306.6709.

[36] A. Bellet, A. Habrard, M. Sebban, Metric Learning, AIM, Springer, 2015. URL: https://hal.science/hal-01121733. doi:10.2200/S00626ED1V01Y201501AIM030.

[37] S. Chopra, R. Hadsell, Y. LeCun, Learning a similarity metric discriminatively, with application to face verification, in: CVPR, volume 1, 2005, pp. 539–546. doi:10.1109/CVPR.2005.202.

[38] F. Schroff, D. Kalenichenko, J. Philbin, FaceNet: A Unified Embedding for Face Recognition and Clustering, in: CVPR, 2015, pp. 815–823. arXiv:1503.03832.

[39] D. P. Vassileios Balntas, Edgar Riba, K. Mikolajczyk, Learning local feature descriptors with triplets and shallow convolutional neural networks, in: Proceedings of the British Machine Vision Conference (BMVC), BMVA Press, 2016, pp. 119.1–119.11. URL: https://dx.doi.org/10.5244/C.30.119. doi:10.5244/C.30.119.

[40] W. Liu, S. Ma, D. Tao, J. Liu, P. Liu, Semi-supervised sparse metric learning using alternating linearization optimization, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '10, ACM, 2010, p. 1139–1148. URL: https://doi.org/10.1145/1835804.1835947. doi:10.1145/1835804.1835947.

[41] S. Kim, D. Kim, M. Cho, S. Kwak, Self-taught metric learning without labels, in: CVPR, IEEE, 2022, pp. 7421–7431. URL: https://doi.org/10.1109/CVPR52688.2022.00728. doi:10.1109/CVPR52688.2022.00728.

[42] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, F. J. Huang, A Tutorial on Energy-Based Learning, in: Predicting Structured Data, 2006, p. 59.

[43] G. Hinton, S. Osindero, M. Welling, Y.-W. Teh, Unsupervised Discovery of Nonlinear Structure Using Contrastive Backpropagation, Cognitive Science 30 (2006) 725–731.