Teaching Machines to Learn by Metaphors

Omer Levy
Teaching Machines to Learn by Metaphors

Research Thesis

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Omer Levy

Submitted to the Senate of the Technion — Israel Institute of Technology
Shvat 5772 Haifa February 2012
The research thesis was done under the supervision of Prof. Shaul Markovitch in the Computer Science Department.

I would like to thank Shaul, for orchestrating my transition from kid programmer to computer scientist. Shaul’s insights were invaluable, and his warm attitude made our joint venture a true pleasure.

I would also like to thank Anat and Dudu, whose feedback and moral support are greatly valued.

I thank my parents, for supporting and encouraging me on my academic endeavor, and for their infinite wisdom and advice.

Finally, I would like to thank Yaya, for making me smile every day.

The generous financial support of the Technion is gratefully acknowledged.
Contents

Abstract .................................................. 1

1 Introduction .............................................. 2

2 Problem Definition ....................................... 8
   2.1 Concept Learning ...................................... 8
   2.2 Transfer Learning .................................... 8

3 Metaphors: Theoretical Background .................. 10
   3.1 What Is A Metaphor? .................................. 10
   3.2 Learning Metaphors ................................... 13

4 How to Learn Metaphors .................................. 15
   4.1 Metaphor Spaces ...................................... 15
   4.2 Metaphor Evaluation .................................. 17
   4.3 Algorithms for Metaphor Learning ................. 19
      4.3.1 Orthogonal Linear Transformations .......... 19
   4.4 Automatic Selection of Metaphor Spaces .......... 20
   4.5 Metaphor Learning with Multiple Source Datasets .. 22

5 Empirical Evaluation .................................... 24
   5.1 Methodology .......................................... 24
      5.1.1 Protocol ........................................ 24
   5.2 Performance of Metaphor Learning ................. 29
      5.2.1 Metaphor Space Selection ...................... 29
   5.3 Performance with Automatic Selection of Metaphor Spaces .. 35
   5.4 Performance with Multiple Source Datasets ....... 36
6 Related Work

6.1 Learning from Few Examples .................................. 41
6.2 Transfer Learning ................................................. 42
6.3 Metaphors and Analogies ........................................ 44

7 Discussion .................................................................. 45

Abstract in Hebrew ..................................................... n
List of Figures

1.1 An illustrated example of two related concepts. The source concept is the set \( \{(x, y) | x \in [4, 9]\} \), and the target \( \{(x, y) | y \in [-3, -2] \cup [2, 3]\} \). The relation between target and source is defined as \( x_s = y_t^2 \). 4

4.1 Per-class comparison ................................. 18
4.2 Feature reordering as a bipartite graph .................. 21

5.1 The 2D circles task .............................. 28
5.2 The digit 8 and its negative .......................... 28
5.3 Comparison of transfer learning methods as function of target sample size ................................. 32
5.4 Performance of the Metaphor Space Selection Algorithm (Digits: Negative Image) .............................. 37
5.5 Performance of the Greedy Multiple-Source Metaphor Algorithm as function of target sample size ................................. 40
Abstract

Humans have an uncanny ability to learn new concepts with very few examples. Cognitive theories have suggested that this is done by utilizing prior experience of related tasks. We propose to emulate this process in machines, by transforming new problems into old ones. These transformations are called metaphors. Obviously, the learner is not given a metaphor, but must acquire one through a learning process. We show that learning metaphors yield better results than existing transfer learning methods. Moreover, we argue that metaphors give a qualitative assessment of task relatedness.
Chapter 1

Introduction

Since the dawn of the 21st Century, machine learning has had a transforming effect on the way we experience technology. Be it the Internet, medical diagnosis, the automobile industry, or even agriculture - applications of machine learning have found their way into nearly every aspect of modern life. Despite its incredible success, machine learning still falls short of the human ability to recognize and induce new concepts from merely a few examples. Even state-of-the-art machine learning algorithms require significant amounts of data in order to learn a new non-trivial concept.

Inevitably, we ask the age-old question of artificial intelligence: "How do humans do it?". Among many cognitive theories regarding the manner in which humans acquire new concepts, one has attracted particular attention within the machine learning community - Transfer Learning [10]. This theory claims that humans learn new concepts by relating them to old, familiar concepts, and utilizing known facts from those domains.

The computational dimension of transfer learning has been thoroughly researched, and many algorithms have been proposed for using existing (source) data while learning from new (target) examples. Existing state-of-the-art methods can be roughly categorized into three main approaches, depending on their assumptions [40]:

**Common Inductive Bias** Inductive bias that performed well on the source should perform well on the target. [42, 52, 2, 4, 23, 48]

**Common Instances** Certain instances of the source data can be used as examples in the target. [57, 31, 17, 41, 58]
Common Features  Features that were discriminating in the source data should be discriminating in the target. [13, 19, 3, 43, 39]

All of the above methods have shown to improve the learning rate when their assumptions hold. Nevertheless, each method makes its own assumptions on the underlying relation between the source and the target, and these assumptions do not necessarily coincide. So which approach is more correct? As a matter of fact, all of these assumptions are too strict to grasp a general notion of concept relatedness. For starters, all of these approaches demand an identical feature space. Are there no concepts that can be described by different feature spaces and still be closely related? Imagine two cameras, each one set to a different resolution, capturing images of the same objects. If we were to represent each set of photos as pixel arrays, none of the above transfer learning methods would be applicable (due to resolution differences), although the data is obviously related. Yet, even if the feature spaces are identical, these assumptions dismiss many intuitive relations.

Consider the following classification problem. Let our feature space be the real numbers (one dimension). The source concept is the interval $[4, 9]$, whereas the target concept is $[-3, -2] \cup [2, 3]$. The relation between target and source can be clearly defined as $x_s = x_t^2$, so intuitively, these concepts are related. Let us review how each approach would attempt to solve this problem.

Common Inductive Bias  Since a single interval can capture the source concept, inductive transfer would attempt to find such an interval that is consistent with the examples from the target. No such interval exists, and the method will fail. Therefore, these two concepts are considered unrelated under the assumption of a common inductive bias.

Common Instances  There is no geometrical overlap between the source and target concepts. This means that the best instance weighting would completely ignore the source data, because every source instance contradicts the target concept. The common instances approach considers these two concepts unrelated as well.

Common Features  With only one feature, this approach is meaningless. However, it is possible to expand our example and show that feature selection can also be destructive. We will do this by adding a dimension
Figure 1.1: An illustrated example of two related concepts. The source concept is the set \{ (x, y) | x \in [4, 9] \}, and the target concept is the set \{ (x, y) | y \in [-3, -2] \cup [2, 3] \}. The relation between target and source is defined as \( x_s = y_t^2 \).

of noise to each concept, in a manner that the noisy dimension in the source is actually the meaningful dimension in the target. Formally, let our feature space be the two-dimensional real plane, the source concept be the sub-plane \( x \in [4, 9] \), and the target concept be the sub-plane \( y \in [-3, -2] \cup [2, 3] \). The relation between source and target is now \( x_s = y_t^2 \), where \( y \) is a meaningless feature in the source and \( x \) is a meaningless feature in the target. Figure 1.1 visualizes this example. It is clear that performing feature selection on the source, and then projecting that selection onto the target would result in an unsolvable classification problem.

The question persists: how should we define whether two concepts are related? It is clear that none of the above approaches supplies an adequate definition. Though several studies have been conducted on when transfer learning should be used [53, 46], and some metrics for measuring relatedness
between learning tasks have been proposed [49, 23, 6, 34], we still lack a qualitative definition of concept relatedness.

This study presents a computational framework for solving the problem of transfer learning, based on an understanding of how concepts are related to one another. The core notion of our framework, inspired by cognitive intuition, is the metaphor - a transformation that converts one feature space into another. We will see that a metaphor is actually the relation between two concepts. Before we dive into explanations, let us consider the following intuitive scenario.

One of the most striking examples of knowledge transfer in children, is their amazing ability to learn new animals from very few examples. Take a hypothetical three-year-old child, for instance. Like most children his age, he can recognize horses, and classify every animal he sees as "horse" or not with superb accuracy. What if we were to take this child to the zoo, and show him a zebra for the first time in his life? It is only reasonable to assume that the child will make some sort of association between the never-before-seen zebra and his old acquaintance, the horse. A new rule for zebra classification could theoretically form in the child’s mind:

\[
\text{zebra} = \text{horse} + \text{black and white stripes}
\]

In the future, the child will be able to classify zebras as accurately as he is able to classify horses.

A metaphor is a mapping of instances from a new problem (target) into instances of an old problem (source). Zebras, for example, would be mapped into horses by removing their stripes. A transformed instance does not need to be a specific example from the source, but must be of the same instance space; the de-striped zebra must be associated with some horse, but not necessarily with one that the child has previously seen.

Metaphors are useful in conjunction with a source classifier (hypothesis) that aggregates the knowledge of the old problem. Once an instance has been transformed from target to source, the source hypothesis classifies it, and its result should indicate whether the original target instance belongs to the target concept. In other words, given a metaphor \( \mu \) and a source hypothesis \( h_{s} \), we can construct a target hypothesis:

\[
h_{t}(x) = h_{s}(\mu(x))
\]
In our example, the metaphor would remove the white stripes of any given animal, and classify the result according to the horse classifier. A zebra would come out positive, while a tiger would not.

Let us examine another (more concrete) example. In terms of computer applications, a popular scenario is that of optical character recognition. The learner is given a very small labeled dataset of images depicting handwritten characters from the (uppercase) Latin alphabet, for example, and is required to recognize the letters that appear in new images. Let us assume that the learner develops 26 binary classifiers, corresponding with each of the characters in the Latin alphabet. Since this dataset does not contain many examples, one can assume that the classification error would be unsatisfactory.

However, what if the learner was given an abundance of labeled Cyrillic letters (as a source dataset)? Could the learner improve its ability to classify Latin characters based on its vast knowledge of the Cyrillic alphabet? Sure, a common letter like 'A' could be classified with little error by using one of the existing Cyrillic classifiers. Surprisingly, even letters such as 'R', which do not appear in the Cyrillic alphabet, could also be classified by performing a reflection (mirror) transformation of the image and then using the Cyrillic 'Я' classifier to recognize it.

Reflection is only one metaphor in a family of optical manipulation metaphors. As we shall presently show, there are many such "families" or metaphor spaces that can be used to translate new problems into old. The core result of this study is our ability to select the best metaphor from a variety of metaphor spaces. In fact, we will show that it is possible to learn a metaphor from a very small set of target examples, thus, solving the original transfer learning problem.

As mentioned earlier, the untackled issue of transfer learning is how to determine when two learning tasks are related. The notion of metaphors sheds new light on the very definition of concept relatedness. Instead of "measuring the distance" between two concepts, metaphors describe the difference; they can explain how concepts relate to one another. Learning a metaphor from one concept to another is, in effect, learning their difference.

This paper provides algorithms and theoretical justification for using the metaphor framework. In addition, we will demonstrate our algorithms across a variety of problems, and compare them to other state-of-the-art methods.
of transfer learning. We will finally address the issue of metaphor learning in a multiple source setting.
Chapter 2

Problem Definition

We present a few introductory notions and definitions in transfer learning, based on Thrun and Mitchell [52].

2.1 Concept Learning

First, let us define the notions of domain and concept learning.

Definition 2.1 (Domain) A domain $D$ is a trio $⟨X, P, f⟩$ where $X$ denotes a feature space, $P$ a probability distribution over $X$, and $f : X \rightarrow \{0, 1\}$ a characteristic function of some subset of $X$.

Intuitively, a real-world object is represented as a point in $X$, while $P$ tells us how likely we are to encounter it. The labeling function $f$ designates which instances belong to a special subset of $X$, known as the concept.

Definition 2.2 (Concept Learning Problem) Let $D = ⟨X, P, f⟩$ be a domain, unobservable to the learner. Given a loss function $\ell : \{0, 1\}^2 \rightarrow [0, 1]$, and a sample $S \subseteq X \times \{0, 1\}$ drawn from $P$ and labeled by $f$, find a hypothesis $h : X \rightarrow \{0, 1\}$ for which the expected loss $E_{x \sim P} [\ell (f (x), h (x))]$ is as small as possible. A concept learning problem is also called a learning task and is denoted by $T = ⟨D, \ell , S⟩$.

2.2 Transfer Learning

We will now define the transfer learning problem.
Definition 2.3 (Transfer Learning Problem) Let \( \langle T_s, T_t \rangle \) be two learning tasks, source and target, respectively. Assume that both tasks share the same loss function. Solve the target learning task \( T_t \); i.e., find a hypothesis \( h_t : X_t \rightarrow \{0, 1\} \) for which the expected loss \( E_{x_t \sim P_t} [\ell (f_t (x_t), h_t (x_t))] \) is as small as possible.

In essence, the transfer learning problem is a relaxation of the original concept learning problem. Instead of utilizing only target domain data, the learner can make use of other (source) data it has already collected. We will generally assume that the learner is significantly more familiar with the source task (\(|S_s| \gg |S_t|\)).
Chapter 3

Metaphors: Theoretical Background

In this section, we propose a solution for the transfer learning problem, using a new notion - metaphors.

3.1 What Is A Metaphor?

A metaphor is a mapping from the target learning task to the source, which preserves label and probability.

Let’s examine the task of spam filtering, for example. Suppose that each domain in this case consists of all the emails in a language, Swedish for source and Thai for target. Our goal is to classify Thai emails as spam or ham, based on previous knowledge of Swedish emails. We shall solve this problem using a metaphor. A metaphor will translate emails from Thai to Swedish, in such a manner as to preserve the spam label; if an email is spam in Thai, its translation will be spam in Swedish. We will then apply our previous knowledge, in the form of a Swedish spam filter, and check whether or not the email was labeled as spam.

Definition 3.1.1 (Perfect Metaphor) Let \( \langle X_s, P_s, f_s \rangle \), \( \langle X_t, P_t, f_t \rangle \) be two domains, source and target. A function \( \mu : X_t \rightarrow X_s \) is a perfect metaphor if:

1. \( f_t(x_t) = f_s(\mu(x_t)) \) for all \( x_t \in X_t \).
Perfect metaphors contain two very powerful assumptions. The first assumption, label preservation, demands that $x$ is part of the target concept if and only if $\mu(x)$ is part of the source concept. This, however, is not enough; for a transformation to be a perfect metaphor, it must also preserve the probability of sampling instances. This means that a set of instances sampled from $P_t$ must be translated into a set that is distributed by $P_s$. Without this criterion, a metaphor might convert target instances into unexplored regions of the source feature space, where $h_s$ may perform poorly.

Revisiting our previous example of spam filtering, a perfect metaphor should not only convert emails in terms of spam labeling, but in context as well. Assuming that people talk significantly more about food than geography, an advertisement for a Pad Thai restaurant in Bangkok could be converted into a commercial for salmon in Stockholm, but not into an article about fjords.

One may notice the similarity between metaphors and reductions (from complexity theory); this is because they represent the same principle: solving problems by translating their instances into those of other, previously-solved, problems. So technically, if we were to obtain a perfect metaphor, we would be able to solve a given transfer learning problem at least as well as we can solve the source concept learning problem.

**Theorem 3.1.1** Let:

1. $\langle T_s, T_t \rangle$ be a transfer learning problem.
2. $h_s: X_s \rightarrow \{0, 1\}$ be a hypothesis for $T_s$ with less than $\varepsilon_s$ loss.
3. $\mu: X_t \rightarrow X_s$ be a perfect metaphor.

Then $h_t(x) = h_s(\mu(x))$ is a hypothesis for $T_t$ with less than $\varepsilon_s$ loss.

**Proof:**

\[
E_{x_t \sim P_t}[\ell(f_t(x_t), h_t(x_t))] = E_{x_t \sim P_t}[\ell(f_t(x_t), h_s(\mu(x_t)))]
\]

(Definition of $h_t$)

\[
E_{x_t \sim P_t}[\ell(f_t(x_t), h_s(\mu(x_t)))] = E_{x_t \sim P_s}[\ell(f_s(\mu(x_t)), h_s(\mu(x_t)))]
\]

(First property of $\mu$)

\[
E_{x_s \sim P_s}[\ell(f_s(x_s), h_s(x_s))]
\]

(Second property of $\mu$)

\[
(\text{Definition of } h_s) \leq \varepsilon_s
\]
This result is theoretically encouraging, but has little significance in practice; having a perfect metaphor at hand is very improbable. That said, obtaining an approximated perfect metaphor (simply, a metaphor) seems drastically more feasible. Our main theoretical result shows that even non-perfect metaphors can perform well in conjunction with a source hypothesis.

**Definition 3.1.2** (ε-Perfect Metaphor) Let \( \langle X_s, P_s, f_s \rangle \), \( \langle X_t, P_t, f_t \rangle \) be two domains, source and target. A function \( \mu : X_t \to X_s \) is an \( \varepsilon \)-perfect metaphor if:

1. \( P_t (f_t (x_t)) \neq f_s (\mu (x_t)) \leq \varepsilon_f \)
2. \( \| \mu (P_t) - P_s \|_1 \leq \varepsilon_P \)
3. \( \varepsilon_f + \varepsilon_P \leq \varepsilon \)

**Theorem 3.1.2** (The Metaphor Theorem) Let:

1. \( \langle T_s, T_t \rangle \) be a transfer learning problem.
2. \( h_s : X_s \to \{0, 1\} \) be a hypothesis for \( T_s \) with less than \( \varepsilon_s \) loss.
3. \( \mu : X_t \to X_s \) be an \( \varepsilon \)-perfect metaphor.

Then \( h_t (x) = h_s (\mu (x)) \) is a hypothesis for \( T_t \) with less than \( \varepsilon_s + \varepsilon \) loss.
Proof: For any

\begin{align*}
E_{x_t \sim P_t} [\ell (f_t (x_t), h_t (x_t))] & = E_{x_t \sim P_t} [\ell (f_t (x_t), h_s (\mu (x_t)))] \\
& = \int_{x_t} P_t (x_t) \cdot \ell (f_t (x_t), h_s (\mu (x_t))) \cdot dx_t
\end{align*}

(Definition of \( h_t \))

\begin{align*}
(\text{First property of } \mu) & \leq \int_{x_t} P_t (x_t) \cdot \ell (f_s (\mu (x_t)), h_s (\mu (x_t))) \cdot dx_t + \varepsilon_f
\end{align*}

\begin{align*}
\left( \text{Second property of } \mu, \quad \text{Holder’s inequality} \right) & \leq \int_{x_t} P_s (\mu (x_t)) \cdot \ell (f_s (\mu (x_t)), h_s (\mu (x_t))) \cdot dx_t + \varepsilon_f + \varepsilon_P \\
& = \int_{x_t} P_s (x_s) \cdot \ell (f_s (x_s), h_s (x_s)) \cdot dx_s + \varepsilon_f + \varepsilon_P \\
& = E_{x_s \sim P_s} [\ell (f_s (x_s), h_s (x_s))] + \varepsilon_f + \varepsilon_P
\end{align*}

(Definition of \( h_s \))

\begin{align*}
(\text{Third property of } \mu) & \leq \varepsilon_s + \varepsilon_f + \varepsilon_P
\end{align*}

\begin{align*}
& \leq \varepsilon_s + \varepsilon_f + \varepsilon_P
\end{align*}

\begin{align*}
& \leq \varepsilon_s + \varepsilon
\end{align*}

3.2 Learning Metaphors

Unfortunately, most real-life transfer learning problems do not include a metaphor in their description. Therefore, the problem of obtaining a metaphor becomes our core focus.

Definition 3.2.1 (Metaphor Learning Problem) Given a transfer learning problem \( \langle T_s, T_t \rangle \), find an \( \varepsilon \)-perfect metaphor such that \( \varepsilon \) is as small as possible.

According to the Metaphor Theorem, we can solve a transfer learning problem by composing a learnt metaphor with a previously learnt hypothesis: \( h_t = \mu \circ h_s \). Basically, we have transformed the original learning problem (transfer learning) into a new learning problem (metaphor learning).

So why should learning a metaphor be any easier than learning the original (target) concept from the target sample \( S_t \) alone? The answer is two-fold. First, when learning a metaphor, one makes use of source data
(S_k) in addition to target data. This, however, is no different from other approaches to transfer learning.

The major advantage of learning metaphors lies within their essence: metaphors represent only the difference between concepts. Each concept may be overwhelmingly intricate by itself, but fairly easy to explain using an already known concept. Let’s revisit our zoological example, and try to define "zebra". This may be an incredibly difficult task without any prior knowledge. However, assuming we already know what a horse is, a zebra can be easily defined as a "horse with black and white stripes". A fundamental assumption of this research is that if two concepts are closely enough related, the associated metaphor will be a relatively simple function, and therefore, considerably easier to learn than the entire target concept.
Chapter 4

How to Learn Metaphors

The common approach in many machine learning scenarios is selecting an appropriate hypothesis space (inductive bias) and searching for the best hypothesis in that space with respect to some utility function. We will present a similar framework for learning metaphors.

4.1 Metaphor Spaces

With resemblance to hypothesis spaces, metaphor spaces define the family of possible transformations. This is a key ingredient when learning metaphors; they must be generic enough to capture the relation between the target and source concepts. On the other hand, metaphor spaces must have a limited amount of degrees of freedom to render them learnable from small target samples. Revisiting our horse and zebra example, the space of texture manipulations contains the desired metaphor, and is therefore general enough; it is also a relatively simple space, making it easy to learn from. The combination of these two traits make texture manipulations an ideal metaphor space for the horse and zebra problem.

Metaphor spaces are also a means of inserting representation-specific bias. For example, if the transfer learning problem is that of image recognition, optical manipulations (such as rotation) may be used. Other representations, such as text, will have no use for optical manipulations, but may have their own specific metaphors. Below are a few examples of metaphor spaces.

**Orthogonal Linear Transformations** Metaphors that perform a linear
transformation on each feature independently. This metaphor space has only $2n$ degrees of freedom, making it easily learnable, as we shall see.

$$M_{lin} = \{ \mu(x_1, \ldots, x_n) = (w_1 \cdot x_1 + v_1, \ldots, w_n \cdot x_n + v_n) | w_i, v_i \in \mathbb{R} \}$$

**Orthogonal Polynomial Transformations** Metaphors that perform a polynomial transformation on each feature independently. This space can be divided into sub-spaces by the highest degree polynomial that was used. The space of orthogonal linear transformations, for example, is one such sub-space.

$$M_{pol(p)} = \{ \mu(x_1, \ldots, x_n) = \left( \sum_{i=0}^{p} \alpha_{1,i} \cdot x_1^i, \ldots, \sum_{i=0}^{p} \alpha_{n,i} \cdot x_n^i \right) | \alpha_{j,i} \in \mathbb{R} \}$$

**Feature Reordering** Metaphors that re-order features, reassigning the values of each feature. For example, rearranging pixels in a bitmap image is a feature reordering. Another example is word-by-word translation from two different bag-of-words representations. Assuming the output has $m$ dimensions, we will mark this metaphor space as:

$$M_{ord} = \{ \mu(x_1, \ldots, x_n) = (x_{i_1}, \ldots, x_{i_m}) | i_j \in \{1, \ldots, n\} \}$$

**Linear Transformations** Metaphors that generate new feature spaces by applying matrix multiplication. Assuming the input and the output have $n$ and $m$ dimensions, respectively, we will mark this metaphor space as:

$$M_{mat} = \{ \mu(x) = A \cdot x | A \in \mathbb{R}^{m \times n} \}$$

**Geometric Transformations** Metaphors that perform geometric manipulations based on rotation, scaling, translation, and reflection. Using the family of geometric metaphors assumes that the data represents images. Formally, if we define $B$ as the union of all basic geometric transformations, we can define the geometric transformations metaphor space as their closure under composition:

$$M_{geo} = \text{Closure}_B (\circ)$$
Note that some of these metaphor spaces contain others. For example, \( \forall p : \mathcal{M}_{pol(p)} \subset \mathcal{M}_{pol(p+1)} \). While larger metaphor spaces increase our descriptive power, they may also hinder our ability to generalize by overfitting. We shall address this issue later on, when discussing algorithms for multiple metaphor spaces.

### 4.2 Metaphor Evaluation

The Metaphor Theorem dictates that a good metaphor adheres to two criteria: label and distribution preservation.

To assure label preservation, we would like to minimize \( P_t (f_t (x) \neq f_s (\mu (x))) \). This value can be estimated by the empirical error over the target training set. For regression problems, mean square error (MSE) is an admissible estimate.

Distribution preservation demands that we minimize the distance between \( \mu (P_t) \) and \( P_s \). The statistical distance between two samples has many empirical estimates, such as the Earth Mover’s Distance [47] and the kernel method [27]. We will use the method of moments [29] to estimate distribution parameters and measure the statistical distance by comparing these parameters. This metric is easily computable, and has strong analytical properties. These properties will be necessary for our analytical methods, presented in the following section.

To combine these two metrics, we could use their weighted sum. This method is naive, since it requires us to tune the label-distribution trade-off for each problem. In addition, label preservation and statistical distance may have entirely different scales, so in essence, their sum is meaningless.

We propose a different strategy for combining label and distribution preservation: while the naive heuristic calculates the statistical distance between the entire source and target samples, we calculate the distance per class. In other words, positive target instances are compared only to positive source instances. In the case of a binary class:

\[
SD (S_t, S_s) = SD (S_t^+, S_s^+) + SD (S_t^-, S_s^-)
\]

where \( SD \) is the statistical distance metric and the sign notation indicates that only instances of that class are considered. This method can easily be
generalized to accommodate multiple classes, and binning techniques may be useful for regression problems. Figure 4.1 visualizes the idea of per-class distance on a simple two-dimensional task.

The per-class heuristic aggregates label and distribution preservation in a meaningful way. Unlike summing average error (a value in the unit interval) and statistical distance (unbounded), statistical distances within the same problem are of identical scale, and can therefore be compared or summed. In addition, if we were to use a weighted sum of statistical distances, it would be equivalent to using a cost matrix in a classification problem; the more weight we put on $SD(S_t^+, S_t^+)$, the less false negatives we are likely to get. As we will see in the following sections, balancing between distances of different classes may be crucial for success, but can be done a-priori, with no need for parameter tuning.

We shall call the per-class heuristic the metaphor heuristic from now on.
4.3 Algorithms for Metaphor Learning

Given a metaphor space $\mathcal{M}$, we can use the metaphor heuristic to search for the most suitable metaphor $\mu \in \mathcal{M}$. This is a de-facto optimization problem, where search algorithms from the hill-climbing/gradient-descent family can be used. While these algorithms have proven themselves empirically across many domains, we can actually tailor efficient algorithms for certain metaphor spaces, by using the analytical properties of the heuristic. We shall present two such examples.

4.3.1 Orthogonal Linear Transformations

When using moment comparison for measuring statistical distance, an analytical solution that finds a metaphor with minimal distance exists. Since each feature is manipulated orthogonally, we can reduce the original optimization problem to $n$ different optimization problems, one for each feature.

We would like to find the parameters $w$ and $v$ that minimize:

$$
\begin{align*}
(E (w \cdot x_t^+ + v) - E (x_t^+))^2 + (E (w \cdot x_t^- + v) - E (x_t^-))^2
\end{align*}
$$

(4.1)

$$
\begin{align*}
(Var (w \cdot x_t^+ + v) - Var (x_t^+))^2 + (Var (w \cdot x_t^- + v) - Var (x_t^-))^2
\end{align*}
$$

(4.2)

Variance loses $v$, so we can use the partial derivative of equation 4.2 and solve for $w$:

$$
\begin{align*}
w^2 = \frac{Var (x_t^+) \cdot Var (x_t^+) + Var (x_t^-) \cdot Var (x_t^-)}{Var (x_t^+)^2 + Var (x_t^-)^2}
\end{align*}
$$

(4.3)

This yields two values for $w$ - positive and negative. A positive $w$ rescales instances and maintains their internal ordering, but a negative $w$ inverts that order. We can detect when an inversion has occurred if the class means are no longer ordered in the same manner. Below is a formal definition of this disambiguation rule:

$$
\text{sign} (w) = \text{sign} \left( (E (x_t^+) - E (x_t^-)) \cdot (E (x_t^+) - E (x_t^-)) \right)
$$

(4.4)

Now that we have obtained $w$, we will use the partial derivative of equa-
tion 4.1 and solve for $v$:

$$
v = \frac{E(x_s^+) + E(x_s^-)}{2} - w \cdot \frac{E(x_i^+) + E(x_i^-)}{2}
$$

Using empirical estimations of expectation and variance, $w$ and $v$ can be easily calculated. This solution can also be generalized to both multiple classes and cost-sensitive scenarios.

**Feature Reordering**

In the case of feature reordering, we would like to match each target feature to the closest source feature, according to the metaphor heuristic. This can be modeled as a bipartite graph, in which one side (left) represents the target feature space and the other represents the source (right). Each node on the left is connected to every node on the right, with a weight edge. Figure 4.2 provides a schematic visualization of the model. Each weight is calculated according to the metaphor heuristic. Assuming there are $n$ features in the target and $m$ in the source, we can formally define the graph $G = (V,E)$ as follows:

$$
V = \{t_1, ..., t_n\} \cup \{s_1, ..., s_m\}
$$

$$
E = \{(t_i, s_j) | i \in \{1, ..., n\}, j \in \{1, ..., m\}\}
$$

This graph describes an assignment problem with weighted edges (costs). Polynomial-time solutions such as the Hungarian algorithm [32] may be used to find the best assignment in terms of minimal total cost; i.e. the feature reordering that minimizes the metaphor heuristic.

### 4.4 Automatic Selection of Metaphor Spaces

As mentioned previously, selecting a suitable metaphor space is critical for the learner’s success. Alas, matching a metaphor space to a given problem is not a trivial task. For the metaphor framework to be truly robust, we require a method of selecting a metaphor space - that fits the problem at hand - from the arsenal of available spaces. This subsection presents such an algorithm.

Given multiple metaphor spaces $\mathcal{M}_1, ..., \mathcal{M}_m$, we will require that they be sorted by preference. Informally, this preference relation will be called
complexity, and its goal is to bias our choice of metaphor space towards simpler spaces, coinciding with Occam’s Razor. While there are many definitions of hypothesis space complexity, our algorithm does not require that the ordering be dependent on one specific metric or another.

The algorithm learns the best metaphor from each space, with respect to the metaphor heuristic, and evaluates each metaphor’s accuracy on the target sample. However, the selected metaphor is not chosen by maximal accuracy alone, as that may result in over-fitting. Instead, a pairwise comparison of metaphors using McNemar’s test [35] is conducted; metaphors that originated in complex spaces are preferred to simpler metaphors only if they are significantly better, beyond a predetermined significance parameter $\alpha$. Hence, the selection algorithm will only select a complex metaphor if it is significantly better than its simpler counterparts. The algorithm’s pseudo-code is presented below.

**Algorithm Metaphor Space Selection Algorithm**

**Input:** A source dataset $S_s$, a target dataset $S_t$, the metaphor heuristic $SD$, a sorted list of metaphor spaces $M_1, \ldots, M_m$, a significance threshold $\alpha$.

**Output:** The simplest metaphor that classifies significantly better than its alternatives.
1. for each $M_i$
2. \[ \mu_i = \arg\min_{\mu \in M_i} SD(\mu(S_t), S_s) \]
3. $\mu = \mu_1$
4. for $i = 2$ to $n$
5. if $\mu_i$ classifies $S_t$ better than $\mu$ with $\alpha$ significance
6. \[ \mu = \mu_i \]
7. return $\mu$

The reader may ponder as to why accuracy is used to compare metaphor spaces, and not the metaphor heuristic? The reason is simple: to avoid over-fitting. Had we used the metaphor heuristic, we would be giving an unfair advantage to complex metaphor spaces, which have more degrees of freedom. The more flexible a model is, the better it can tighten itself around the sampled examples - or in other words, over-fit. In practice, the resulting metaphor heuristic values differ significantly in range; comparing a simple metaphor to a complex one using the metaphor heuristic would be like comparing apples and oranges. Therefore, a different metric, which can also be sanctioned by the tools of statistical significance, must be chosen. Accuracy is such a metric.

4.5 Metaphor Learning with Multiple Source Datasets

The multiple sources scenario is a very natural one. In human cognition, many concepts are available, but only a select few are relevantly related to the target concept. Thus, the need to filter irrelevant concepts arises. Formally, we shall define this by generalizing the original transfer learning task; instead of a single source task $T_s$, we now have $N$ possible source tasks $T_s^1, ..., T_s^N$.

We propose an algorithm that solves this problem. Given a variety of source datasets, the algorithm returns a metaphor based only on one. This is done by learning a metaphor for each source, and then greedily selecting the best one, with respect to the metaphor heuristic. Note that we can safely use the metaphor heuristic across different sources because the metaphor space is identical. The algorithm’s pseudo-code is presented below.

**Algorithm Greedy Multiple-Source Metaphor Algorithm**
Input: A set of source datasets $S_1^s, ..., S_N^s$, a target dataset $S_t$, the metaphor heuristic $SD$, a metaphor space $\mathcal{M}$.

Output: The best metaphor w.r.t. the metaphor heuristic.

1. for each $S_i^s$
2. $\mu_i = \text{argmin}_{\mu \in \mathcal{M}} SD(\mu(S_t), S_i^s)$
3. return $\text{argmin}_{\mu_i \in \{\mu_1, ..., \mu_m\}} SD(\mu_i(S_t), S_i^s)$
Chapter 5

Empirical Evaluation

Though our theoretical results are encouraging, the algorithms presented in the previous section are of a heuristic nature, and must be evaluated empirically. For that reason, we have conducted a series of experiments to evaluate how well our algorithms perform with respect to other transfer learning methods. We also wish to gain a deeper understanding of metaphor learning, and examine the qualitative information acquired during the learning process.

5.1 Methodology

We present the protocol, datasets, and reference methods we used to enable reproduction of our tests.

5.1.1 Protocol

As mentioned in our introduction, transfer learning aims to solve machine learning problems when target data is scarce. An excellent method of determining how well an algorithm performs with small sample sizes is by observing its learning curve - a chart displaying the method’s error (or any other performance metric) as a function of the training data’s size. In our setting, the size parameter will affect only the amount of target instances available to the algorithm. The amount of source instances will remain constant at a large number (1000), since we are always under the core assumption of metaphor learning: the source concept is well-known and can be classified with small error.
Performing classic cross-validation is insufficient, because the minimal target training set is half of the original set (in the case of two-fold cross-validation). Since we are interested in understanding how metaphor learning performs under very small sample sizes, a variation of cross-validation was used. In essence, we partition the pool of target instances into small chunks, and perform two-fold cross-validation on each chunk. This gives us an amount of results that is double the number of chunks, and can then be used for testing statistical significance. Apart from the protocol’s versatility in terms of training set size, it also ensures that each instance is trained upon once and tested upon once (exactly). The protocol is presented below in detail.

Algorithm Experiment Protocol

Input: A source dataset $S_s$, a pool of target instances $pool$

1. for $n = 1$ to $\frac{|pool|}{2}$
2. partition $pool$ into chunks of size $2n$
3. for each $x$ in chunks
4. $A = \{x_1, ..., x_n\}$
5. $B = \{x_{n+1}, ..., x_{2n}\}$
6. learn from $S_s$, $S_t = A$ and test on $B$
7. learn from $S_s$, $S_t = B$ and test on $A$

This protocol was used in every experiment, with a source dataset of 1000 instances and a pool of 600 instances from the target task.\(^1\)

A variety of base learners was reviewed for each domain, including SVM, C4.5, Naive Bayes, and Nearest Neighbor. We used WEKA’s implementation for these algorithms with their default parameters [28]. Initially, results based on the Nearest Neighbor learner will be presented (to avoid redundancy). We will then demonstrate that similar behavior is observed across base classifiers; i.e. modifying the base classifier does not change a transfer learning algorithm’s superiority.

Statistical significance was tested for each sample size using Wilcoxon’s test [56] with 95% confidence. Since this is a pairwise test, we compared the metaphor learner with every other non-metaphor learner, on every fold. The range of significance will be presented alongside the average error.

\(^1\)With the exception of Latin and Cyrillic, which has 100 target instances.
Transfer Learning Tasks

As defined in the previous sections, transfer learning tasks are composed of two samples (datasets) from the source and target domains. These domains must be related; there is no sense in trying to learn one concept based on another when there is no clear relation between the two. To truly evaluate algorithms in transfer learning conditions, the domains must also be significantly different from one another. Finally, we will also assume that all features (excluding the class feature) are numerical. While we have also looked into metaphors for nominal features, numerical features simplify the experimental process due to the continuous nature of many metaphor spaces. Summing up, we would like our transfer learning tasks to be related, different, and numerical.

The research literature lists only few transfer learning tasks. However, even these few do not satisfy the above criteria. In addition, we would like to examine transfer learning tasks in which the source and target feature spaces differ, but unfortunately, we were unable to find such datasets. Therefore, we present several new transfer learning tasks.\(^2\)

1D Intervals This task is an automatically generated version of the problem we presented in our introduction. The feature space is the real numbers (one dimension), the source concept is the interval \([4, 9]\), and the target concept is \([-3, -2] \cup [2, 3]\).

2D Circles This problem lies in \(\mathbb{R}^2\). The source sample is normally distributed around the origin, and the unit-circle is the source concept. The target concept is similar, but instead of being centered on the origin, it is centered on \((10, 10)\). Figure 5.1 visualizes the distribution of data in this task.

Digits: Negative Image The problem of recognizing the content of an image’s negative (pre-development) is an intuitive transfer learning problem. In this task, the learner is required to identify numerical digits within negative images, after learning to recognize digits in their regular positive form. The source task is that of the original dataset: given an image, identify which of the ten numerals it represents. While

\(^2\)All tasks are available at: http://www.technion.ac.il/~omerlevy/datasets
the source data consisted of black ink on a white background, the target data is inverted - white on black. Figure 5.2 shows an example digit and its negative.

**Digits: Recognize 9, Recognize 6** We created another task from the digits dataset, based on two sub-tasks: separating 9 from the rest of the digits (source), and discriminating 6 likewise (target).

**Digits: Higher Resolution** This scenario simulates the case where we have much data from a low resolution camera, and little data of high resolution. The target data will be the original digits dataset. To create the lower resolution source data, we will merge each two-by-two quad of pixels into one pixel, where its intensity is their average; i.e. a quad of three white pixels (intensity 0) and one black pixel (intensity 16) in the original dataset will become one light-gray pixel (intensity 4). Note that the source feature space consists of only 16 dimensions, while the target feature space has 64.

**Latin and Cyrillic** Similar to the problem we presented in our introduction, we will now learn how to recognize Cyrillic letters based on previous knowledge of the Latin alphabet. For now, we will focus on recognizing the letter 'Я' by using an existing classifier for 'R'. In the following section, we will generalize this experiment to the entire Latin and Cyrillic alphabets. The source data contains images of typeface uppercase Latin characters where only those depicting the letter 'R' are labeled positive. Similarly, the target consists of uppercase Cyrillic letters in which 'Я' is labeled positive and any other letter negative.

**Wine** The wine quality task [16] can be found in the UCI Machine Learning Repository. It consists of two distinct datasets: red wine and white wine. The learner must predict a wine taster's score for any given wine (a regression task). Although both datasets contain instances in the same feature space, red and white wine are not equally distributed. Unlike other tasks presented here, this one is a regression problem and not one of classification.

Note that the three digit tasks are based on the optical digit recognition dataset from the UCI Machine Learning Repository [25].
Figure 5.1: The 2D circles task

Figure 5.2: The digit 8 and its negative
Reference Methods

Three baseline methods were compared against metaphor learning algorithms:

**Target Only** Learns only from the target dataset.

**Identity Metaphor** Learns only from the source dataset.

**Merge** Learns from the union of source and target datasets.

In addition, the following state-of-the-art algorithms were compared:

**Frustratingly Easy Domain Adaptation (FEDA)** As presented by [19]. Transforms both source and target data into a common representation, and learns a classifier from the new dataset.

**Multi-Task Learning (MTL)** Presented by [13]. Employs a neural network to concurrently learn two (or more) tasks. We used an implementation by [54], within the WEKA framework.

**TrAdaBoost** As presented by [17]. Weighs each instance (source and target) according to a boosting scheme. Number of epochs was set to 100.

Note that besides Target Only, all methods are applicable only when both source and target feature spaces are identical.

### 5.2 Performance of Metaphor Learning

We will show that when we are given a suitable metaphor space, heuristic search finds a good metaphor that describes this relation. Moreover, we shall demonstrate that classifiers based on these metaphors are more accurate than classifiers generated by other transfer learning algorithms.

#### 5.2.1 Metaphor Space Selection

To meet the assumption that our metaphor space \( M \) contains an adequate metaphor, we must select one that is generic enough on one hand, yet specific enough to enable some form of bias. We will later relax the specificity demand when discussing automatic selection of metaphor spaces.
The first two tasks (intervals and circles) have a pure mathematical relation between the source and the target, so polynomial metaphor spaces were used for each (quadratic for intervals and linear for circles). In the negative images task, each pixel in the target domain relates to a corresponding pixel in the source; therefore, an orthogonal metaphor space (such as orthogonal linear transformations) is a good choice. The same metaphor space was used for the wine task, which also requires feature alignment. Intuitively, sixes resemble rotated nines, so the family of geometric transformations should be suitable. The same logic follows for the Latin and Cyrillic task. Changes in resolution require metaphors that can handle different source and target feature spaces; non-orthogonal linear transformations were thus chosen.

Where applicable ($M_{pol}$), we used the analytical method presented in the previous chapter to find the best metaphor. To search the remaining metaphor spaces, steepest-ascent hill-climbing was used, returning a metaphor upon reaching a local minimum. Table 5.1 presents each transfer learning task and the metaphor space that was chosen.

The previous section described $M_{geo}$ in a manner that is too abstract to reconstruct, so we shall therefore elaborate. The base transformations are: 3 rotations ($90^\circ$, $180^\circ$, $270^\circ$), 16 translations (8 horizontal, 8 vertical), and 2 reflections (horizontal axis, vertical axis). This space is closed under composition, allowing a combination of several base transformations to be a metaphor in $M_{geo}$. The hill-climbing algorithm starts with no transformations (the identity metaphor) and fuses base transformations with the current state until a local minimum has been reached. This state (metaphor)
is eventually returned.

Quantitative Results

Figure 5.3 shows the performance of each method on each transfer learning task, as a function of the target sample size. Table 5.2 shows the range of target sample sizes in which metaphors dominated all other methods with 95% significance.

We would like to point out a few observations. First of all, metaphors performed significantly better than every other method, when target data was scarce. Nevertheless, with sufficient amounts of target data, metaphors will eventually fail to provide better classification than the target-only method, as can be seen in figures 5.3(a), 5.3(d), and 5.3(g). It was also observed among the other datasets when the target sample size was significantly larger than 20. This phenomenon coincides with human cognition; when learning some concept for the first time, we will attempt to project it onto another concept, but eventually, we will become experts from studying the relevant data alone.

Another interesting observation is that state-of-the-art methods did not perform better than baseline methods. While it has been demonstrated in previous work that these methods perform well, this claim does not hold when the data does not meet the methods’ underlying assumptions.

Finally, we would like to bring the reader’s attention to the results regarding 1D Intervals in figure 5.3(a). This setting was given as a very convincing example in our introduction, yet the empirical results seem somewhat disappointing; one would expect metaphors to have a more significant edge over the other methods. The explanation is twofold. First, $M_{pol(2)}$ has three degrees of freedom, which makes it prone to over-fitting. Secondly, the source classifier given in the introduction was an interval classifier; i.e. the hypothesis space consisted of single intervals. The experimental setting, however, used the Nearest Neighbor algorithm. Had the more restrictive interval classifier been used, the remaining methods would not have been able to achieve low error classification, even with an abundance of target instances.
Figure 5.3: Comparison of transfer learning methods as function of target sample size
### Table 5.2: Range of target sample sizes where metaphors dominate all other methods

<table>
<thead>
<tr>
<th>Transfer Learning Task</th>
<th>Significance Range (Sample Size)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1D Intervals</td>
<td>$1 - 5, 7 - 8$</td>
</tr>
<tr>
<td>2D Circles</td>
<td>$1 - 20$</td>
</tr>
<tr>
<td>Digits: Negative Image</td>
<td>$3 - 20$</td>
</tr>
<tr>
<td>Digits: Recognize 9, Recognize 6</td>
<td>$1 - 20$</td>
</tr>
<tr>
<td>Digits: Higher Resolution</td>
<td>$1 - 20$</td>
</tr>
<tr>
<td>Latin and Cyrillic</td>
<td>$3 - 18$</td>
</tr>
<tr>
<td>Wine</td>
<td>$2 - 16$</td>
</tr>
</tbody>
</table>

Qualitative Analysis

A closer look at the actual metaphors that were found should provide a broader understanding of how metaphors work. An arbitrary target instance was selected from each optical domain (Negative Image; Recognize 9, Recognize 6; Higher Resolution; Latin and Cyrillic). Upon that instance, we invoked actual metaphors that were found for different target sample sizes. Table 5.3 shows the input (target instance) and the outputs (translated instances).

It can be noticed that even a few target instances are enough to learn a good metaphor. With only two examples, the outputs already resemble the source data, and five are sufficient for nearly perfect classification. The learning curve is most visible with the Negative Image task, where one target example creates an unintelligible mixture of black and white, two are enough to form the general shape of 8, and five (and ten) examples are enough to remove any shadow of doubt regarding the digit’s class.

Performance Across Base Classifiers

All of the results presented till now used the Nearest Neighbor algorithm as the base classifier. To ensure that the results are not biased towards one base classifier in particular, we repeated the experiments across three other base classifiers as well: C4.5, Naive Bayes, and Linear SVM. We used WEKA’s implementation and default parameters for all of these algorithms.

33
<table>
<thead>
<tr>
<th>Transfer Learning Task</th>
<th>Target Instance</th>
<th>Target Sample Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digits: Negative Image</td>
<td><img src="image1" alt="Image" /></td>
<td><img src="image2" alt="Image" /></td>
</tr>
<tr>
<td>Digits: Recognize 9, Recognize 6</td>
<td><img src="image3" alt="Image" /></td>
<td><img src="image4" alt="Image" /></td>
</tr>
<tr>
<td>Digits: Higher Resolution</td>
<td><img src="image5" alt="Image" /></td>
<td><img src="image6" alt="Image" /></td>
</tr>
<tr>
<td>Latin and Cyrillic</td>
<td><img src="image7" alt="Image" /></td>
<td><img src="image8" alt="Image" /></td>
</tr>
</tbody>
</table>

Table 5.3: Metaphor invocation across target sample sizes
Changing the source classifier did not have a significant effect on the metaphor learner’s performance. Metaphors are indifferent to the base classifier’s type because the metaphor heuristic is independent of the base classifier; the same metaphors will be selected by the algorithm, regardless of the base classifier. As long as each one of the base classifier’s inductive bias is general enough to capture the source concept with small error, classification by metaphors will display the same performance across base classifiers. In Recognize 9, Recognize 6, Naive Bayes seems to hinder the metaphor learner’s performance, but this is due Naive Bayes’s inability to properly classify the source concept, even with a thousand examples.

5.3 Performance with Automatic Selection of Metaphor Spaces

After evaluating the metaphor framework under the assumption of a single metaphor space, we shall proceed to generalize this setting by testing the Metaphor Space Selection Algorithm presented in the previous section. In this series of experiments, we selected three optical recognition tasks (Negative Image; Recognize 9, Recognize 6; Latin and Cyrillic) and four metaphor spaces that were applicable ($M_{geo}$; $M_{ord}$; $M_{lin}$; $M_{pol(2)}$). The metaphor spaces were given in order of complexity, beginning from structured geometric transformations, through reordering, orthogonal linear transformations ($2n$ degrees of freedom), and finally orthogonal quadratic transformations ($3n$ degrees of freedom). The algorithm was given a significance threshold of $\alpha = 90\%$ (similar thresholds yielded similar results).

In Recognize 9, Recognize 6 and in Latin and Cyrillic, $M_{geo}$ was selected at every fold. Not only did the space of geometric manipulations enjoy the algorithm’s bias, it was also the best metaphor space for the task at hand. The Negative Image task was not as simple, since geometric manipulations do not describe the negative image relation. The space of orthogonal linear transformations ($M_{lin}$), however, proves as a significantly better metaphor space as the number of target examples grows. Note that while three examples or less do not provide 90% significance in McNemar’s test, five target examples are enough to convince the algorithm to select the $M_{lin}$ more than half of the time. Table 5.4 shows the portion of folds in which each metaphor space was selected, by target sample size, in the Negative Image task.
Table 5.4: Distribution of metaphor space selection (*Digits: Negative Image*)

These results show quick convergence into the best metaphor space, even when the algorithm is strongly biased towards simple spaces. The Metaphor Space Selection Algorithm’s performance in classification also suggests that the algorithm converges into the right metaphor space within a small amount of target examples. Figure 5.4 presents this result alongside the original classification performance of metaphors on *Negative Image*.

### 5.4 Performance with Multiple Source Datasets

Another generalization of the original metaphor framework is that of multiple sources. We evaluated the Greedy Multiple-Source Metaphor Algorithm
Figure 5.4: Performance of the Metaphor Space Selection Algorithm

(Digits: Negative Image)
on two multiple-class tasks (Recognize 9, Recognize 6; Latin and Cyrillic). Each task’s source dataset was split into several binary sources: Recognize 9, Recognize 6 to all the digits (excluding 6), and Latin and Cyrillic to all 26 Latin characters. This means that sixes do not necessarily need to be related to nines - perhaps a better metaphor to some other digit exists.

It turns out that this is not the case. Even with two target examples, 9 is selected more than two-thirds of the time as the most fitting source. At four target examples, 9 is selected in more than 90% of the folds. A similar process occurs with the Latin and Cyrillic task; while there are 26 available sources, R is repeatedly selected as the best source. The remaining sources are selected an insignificant amount of times, apart from H and B, which arrive at distant second and third places. This is reasonable, since both H and B share resemblance to R, in terms of bitmaps. Table 5.5 shows the portion of folds in which the original hand-picked source dataset were selected, by target sample size.

The quick convergence into the best source can also be observed in the classifier’s performance. Figure 5.5 compares the algorithm’s performance with that of the single-source metaphor, and further demonstrates convergence into quality classification with very few examples.
<table>
<thead>
<tr>
<th>Sample Size</th>
<th>Digits: 9</th>
<th>Latin: R</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>42.7%</td>
<td>25%</td>
</tr>
<tr>
<td>2</td>
<td>67%</td>
<td>48%</td>
</tr>
<tr>
<td>3</td>
<td>83.5%</td>
<td>65.6%</td>
</tr>
<tr>
<td>4</td>
<td>90.7%</td>
<td>66.7%</td>
</tr>
<tr>
<td>5</td>
<td>93.3%</td>
<td>75%</td>
</tr>
<tr>
<td>6</td>
<td>96%</td>
<td>75%</td>
</tr>
<tr>
<td>7</td>
<td>97.6%</td>
<td>78.6%</td>
</tr>
<tr>
<td>8</td>
<td>98.6%</td>
<td>83.3%</td>
</tr>
<tr>
<td>9</td>
<td>100%</td>
<td>70%</td>
</tr>
<tr>
<td>10</td>
<td>100%</td>
<td>80%</td>
</tr>
<tr>
<td>11</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>12</td>
<td>100%</td>
<td>87.5%</td>
</tr>
<tr>
<td>13</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>14</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>15</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>16</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>17</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>18</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>19</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>20</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 5.5: Distribution of source dataset selection
Figure 5.5: Performance of the Greedy Multiple-Source Metaphor Algorithm as function of target sample size.
Chapter 6

Related Work

This section presents the literary background preceding our study. Work on learning with few examples will be presented at first, followed by a broader survey and comparison of existing transfer learning methods. We will finally present previous studies on metaphors and analogies in both cognitive and computer sciences.

6.1 Learning from Few Examples

The problem of learning from few examples has been studied extensively throughout the years. Three main approaches have been suggested: explanation-based learning, semi-supervised learning, and transfer learning. Common to all of these approaches is the fact that they add additional information beyond the original examples; however, they differ by the type of that information.

Explanation-based learning \[22, 38\] claims that by relying on known rules (axioms), one can logically deduce a hypothesis that explains an observation. Thus, a learner equipped with enough axioms can grasp entire concepts from a single observation.

Semi-supervised learning \[9, 8, 14\] assumes that in addition to a few labeled examples, the learner is given many unlabeled examples. Understanding some underlying qualities of the data from the abundance unlabeled examples may assist the learner in forming a better hypothesis.

The transfer learning setting incorporates prior knowledge in the form of...
an additional dataset of a related concept. Metaphors fall into this category.

6.2 Transfer Learning

Transfer learning's origins can be traced to Pratt’s early work with backpropagation neural networks [42], where she showed that training a classifier on a different (related) task before attempting to learn the target task, can improve learning speed. Thrun and Mitchell [52] provided a more formal definition of transfer learning. Their setting included multiple source tasks and a single target task, all from the same feature space, but not of the same distribution or concept. Meanwhile, Caruana studied a similar setting [13], where all given learning tasks were target tasks; hence, the name Multi-Task Learning. A new framework for transfer learning called Domain Adaptation has recently been introduced [18, 5]. Domain adaptation assumes that both source and target tasks are essentially the same concept, but their corresponding sample sets have different underlying distributions. While many setups have been presented, even more solutions have been suggested. We divide these solutions into three major approaches, as presented in our introduction:

**Common Inductive Bias** An interesting conclusion of Pratt’s work was that the same inductive bias performed well on related tasks. This paved way to additional methods [52, 2, 23, 48] that used the parameters of a source classifier as an inductive bias for learning the target classifier. Baxter [4] formalized the notion of Bias Learning (or "learning to learn").

**Common Instances** This approach assumes that certain instances of the source data can be used as examples in the target. Wu et al [57] demonstrated the notion on SVMs, by weighing source instances according to their relevance in the target task. Daume and Marcu [18] presented a Bayesian method for learning under the common instances assumption. In recent years, a series of boosting algorithms [44, 17, 41, 58] have proven to be effective when this assumption holds.

**Common Features** This approach harnesses discriminating features that are common to all tasks [13, 19, 3, 43, 39]. These methods attempt to
identify the "lowest common denominators" among all learning tasks, and amplify them with feature re-weighting or similar techniques. Obviously, they require all tasks to be of the same feature space, making them popular in multi-task learning and domain adaptation.

Metaphors do not assume that the source and target have anything in common, but rather that a transformation function from one to another exists. In this sense, metaphors differ dramatically from previous methods.

Other approaches to transfer learning exist, but are not applicable to our setting. Manifold alignment, for example, [55] has also shown interesting results, but requires actual examples of this alignment (i.e. pairs of source-target instances). This kind of information is not available to the learner under the terms of our problem’s definition. Relational learning has also been studied in the context of knowledge transfer [37, 21], and a number of methods for reinforcement learning have also been proposed, which attempt to map information from the source to the target [33, 51, 50].

Adapting the last approach to concept learning would mean mapping source instances to the target feature space under some transformation. However, evaluating such a source-to-target transformation would require knowledge of the target concept - knowledge that is absent, given the sparsity of target examples. No ability to evaluate means no ability to learn a proper mapping from source to target. Metaphors are fundamentally different; while mapping source to target instances transfers knowledge, metaphors transfer the problem to where the knowledge is.

Metaphors also describe the relation of one concept to another. The issue of concept relatedness has been studied since the beginning of transfer learning [53, 49]. Throughout the years, additional definitions and measures of relatedness have been presented [46, 23, 6, 34]. Metaphors provide a different approach for evaluating one concept’s relation with another, in the sense that they describe the manner and not the amount of the relation.

NLP has greatly benefited from the onset of transfer learning. Since human tagging of texts is a resource-consuming business, development of NLP systems could be downsized if related examples were to be used as well. The task of sentiment analysis, for example, has become the unofficial test-bed for domain adaptation, and has spiked NLP-specific methods for transfer in return [15, 7].
6.3 Metaphors and Analogies

The notion of metaphors and analogies in human cognition had been studied by Gentner throughout the 80’s [26]. At the same time, Analogical Reasoning (also called Computational Metaphors) was developed by Carbonell and others [12, 11, 20, 30, 24]. Their prime focus was to use prior knowledge to assist reasoning tasks (such as logical deduction and planning) in new domains. In inductive learning, an instance-based method that incorporates analogical reasoning was recently introduced [36].

While there are some similarities between metaphors and previous work in Case-Based Reasoning (CBR) [45, 1], it is important to notice the fundamental differences. While CBR maps target problems to previously observed source problems, metaphors may translate target instances into never-before-seen instances in the source feature space. Metaphors are not necessarily similarity-based (as in CBR), and do not even require the source and target feature spaces to be identical. Another disparity is that CBR retrieves a different set of source problems for each given target problem, while metaphors translate the entire target feature space.
Chapter 7

Discussion

We embarked on a mission to solve one of the greatest drawbacks in machine learning - the problem of learning with few examples. Inspired by human cognition, we presented a novel transfer learning approach: metaphors. By relating a new learning problem to one that has already been solved, the learner is able to acquire new concepts after observing only a handful of examples.

To render this intuitive idea into a reality, we defined and built the foundations of the metaphor framework. The Metaphor Theorem shows that if two concepts are related by metaphor, the new (target) concept can be classified as accurately as the original (source) concept. Metaphor spaces and their automatic selection, alongside the metaphor heuristic and our efficient search methods, provide a robust toolbox for learning metaphors. These tools were tried and tested in a real transfer learning setting, and performed better than state-of-the-art transfer learning methods.

While we have stressed the importance of the target concept being related to the source concept, the reader may naturally ask what were to happen if the concepts were not related by metaphor? Obviously, metaphors would not work in this case; neither would they perform well in a scenario where the concepts are too distant for a simple metaphor to describe their relation. If translating from target to source requires a sophisticated metaphor, we might as well learn the target without using the source at all. Even humans are sometimes required to learn entirely foreign concepts, and in these particular situations, tabula rasa is the only way to go. We can therefore conclude that if a clear relation between target and source does not exist, the problem at
hand does not fit the definition of a transfer learning task.

However, when such a relation exists, metaphors double our profit. Not only are we rewarded with better classification, we are also provided with an explanation as to how the new concept relates to the old. Metaphors provide a unique assessment of task relatedness - a qualitative difference rather than a numerical measure.

Since relatedness between concepts may take on many forms, the selection of a suitable metaphor space is critical. Selecting an appropriate metaphor space is not a trivial choice, and one may question the amount of engineering involved in this process. For this precise reason, we have designed and tested the automatic Metaphor Space Selection Algorithm. As demonstrated, the algorithm is able to select the best metaphor space from a variety of spaces after observing a very small amount of examples. Nevertheless, we must re-ask the question at a higher level: how does one go about selecting an adequate kernel function, hypothesis space, or even feature space? Much research has been devoted to answering these questions, and their answers should easily be applicable to metaphor spaces without prejudice.

We would finally like to point out that metaphors are not limited to a single source concept, and can rely on a collection of known concepts. Together with the ability to rapidly learn new concepts from few examples, metaphors may realize the dream of never-ending learning by enabling the learner to constantly form new knowledge based on recently acquired concepts. A machine that builds layer upon layer of knowledge from pebbles of experience - that is the vision that metaphors will realize.
Bibliography


כיצד ללמד מחשב לلزم באמנעות

משל

עמר לו
לפיו למד מחשב למדים באומנות

משול

חובור על מחבר

לשמם מיילי חלקי של תחום מחשב התchanger
מגיסטר למדעי מחשב

עומר לי

הוגשת התוכנית למכון טכנולוגי לישראל
 одно ל臬טוף ליו"ט
נובמבר 2012

Technion - Computer Science Department - M.Sc. Thesis  MSC-2012-08 - 2012
המחקר נערך בנהנית פורום של דוקטור המחקר, פרופ’ שאול מרקוביץ’ פקдолיה, למדעי המחשב

אנירוץ להודות לשלואל, שניצחה על התחרות של מילר, מתמטכת למחר. הוא דוגן
מאורעות שונים ו ייסויים חוסם קפוא את העבורה עב שואל לתחנוג של מדע.
אנירוץ—heregation הליגה לולם עלי התמקדות ההברירה והמשובים הערכיים
אני מודע להחריר, על התמקדות והויזיות הבהלהיליסיס במשוך האקדيمي של
העמקות על החרות האינטנסיבית.
לבסוף, אני רצה להודות ליעל, שמיעלה חום על שפתני מידי יום.
אני מתוד הלאיני על התמקדות הבהלהיליסיס במשוך האקדימי

Technion - Computer Science Department - M.Sc. Thesis MSC-2012-08 - 2012
In this document, there is a discussion about the impact of computer-based learning on the 21st century. The applications of computer-based learning are already prevalent in various fields such as the internet, medical examination, the automotive industry, and agriculture. Despite its success in commerce, computer-based learning is not able to compete with human learning in cases where the need arises to learn new concepts from a few examples. A particular algorithm, computer-based learning, can be used to obtain a large amount of information on how to learn a new concept from various examples and its use in known facts.

In a sense, the human mind asks the question: "How do humans do that?" Among the many theories that try to explain how humans learn new concepts, one of them is of particular interest. This theory asserts that humans learn new concepts by referring to similar concepts they are familiar with, and using known facts from their area.

Numerous studies have been conducted to improve computer-based learning, and it is suggested that algorithms that utilize existing knowledge (source) can be used to cope with a new learning problem (target). It is possible to divide these methods into three main approaches, which differ in their assumptions: Broader generalizations, common objects, and shared characteristics.

The scientific literature shows a wide variety of methods for improving computer-based learning, which have been used, and we find that their assumptions are not always consistent. Is there a correct approach? In fact, all the approaches mentioned require strong conditions, and they cannot be applied to the relationships between concepts and their sources. In this dissertation, we refer to various computer-based tools, each with a different resolution. If we view the images taken in this manner, we can get a different description for each camera; that is, the resolution is different. If we look at the resolution of the images in this manner, we can get a different description for each camera.
נצלם את אותם עצמים בשתי המצלמות, ברור כי היה קש חק בור ביניהם הת-Headers.

המצולמות, אם כי שמידת בירור עד שלמרות הת-Headers.

ניקול לודגון את ביעת הסיווג הבאה. נdiği את מרחבי הת-Headers לכל ציר הממשי.

(פיזור את). מושג המקור היה הקטוע [4, 9] ואילו מושג המטריה היה [2, 3].

ניתן לתллер את التركي ביצuerdo למטריה במלאכת ההתחפשותoyer, לכל נקודה פסיק.

יכモושיגי אלו ק aprox.航海 בצמר ארז אופי CASEY. הניה בניה את אופי הת-Headers הש>()) לש כל גישה לחיה וזו.

мотрבח השתרעות משוחק: מושג ינית לחרת את מושג המקור בחרת, כיוון וזו התסה.

למען הקטוע תחית הקטוע עם מושג המטריה. מארב שקטוע כוי חינ קיימ, והיוותיקከשל.

ניתש וא@RunWith את מושג המקור והמטרה כלכליתупитьואני טור.

ענני מושיגים: נ النواب כי כאל קימיים תיפה הגאומטריה בין מושג המקור והמטרה.

אוצר את מרחבי המקור, אל נוחיש שמקי מוסים, מושג הקטוע תקע ממעש המטריה.

המטרה המקור של מרחבי המטריה יוליג כחרת לחרת העדключение עם מושג המטריה.

לחלקה לחלקה הלמידה. הניה ו蚍וחיב את המקור והמטרה כלכליתupo תור.

 kuruluş מושגית: מושג יונון עם קימיי המקור החיה, הניה וייתר כל

 формирова. רוחב את הת-Headers על מתן לדימないように ולהיות החיה תקיעה. תפורט.

ממסת ענני פיזור (נקחייה יונון תק Penal בוסיון) באופי גורילה, גורילה או מרחבי הת-Headers.

. \( y \in [-3, -2] \cup [2, 3] \) מושג המטריה \( x \in [4, 9] \) כמיושם המטריה.

המשרה בין מקוד מחוזיר ביצuerdo במלאכת ההתחפשותoyer. ניונה לחרת ואחרת

הometown יזarmacy לע ספק המקור וה删除成功ה בחרתアウト לע המטריה נובול לייגון עלייה-

 unfoldית.

_fhית פ讳ית.

השאלה עודנה נושל: כיווןות תקも多いין קימי המקור החיה, הניה ו〒יתר כל

ופגישה. נרחב את הת-Headers על מתן לדימないように ולהיות החיה תקיעה. תפורט.

אמני 컴퓨터 מחוורארה, גורילה או מרחבי הת-Headers.

התקיים 미תיי שמייתו מלקד ידחבר ביני

מקוד מחוזיר ביצuerdo במלאכת ההתחפשותoyer. ניונה לחרת ואחרת

הometown יזarmacy לע ספק המקור וה'LBLיתחת בחרתアウト לע המטריה נובול לייגון עלייה-

 unfoldית.

fhית פ讳ית.

מאטשאלו ועיר ארית: כיווןות תקも多いין קימי המקור החיה, הניה ו〒יתר כל

ואחת מחושן סלון אתי בחימה ומזרחית פשיטה. אממי קימי מחוורארה, גורילה או מרחבי הת-Headers.

התקיים 미תיי שמייתו מלקד ידחבר ביני

מקוד מחוזיר ביצuerdo במלאכת ההתחפсходיה. ניונה לחרת ואחרת

הometown יזarmacy לע ספק המקור והLBLיתחת בחרתアウト לע המטריה נובול לייגון עלייה-

 unfoldית.
ללמוד ולהורות על defensivelyحة מופעים مدريد הצפה. דמיינו ילד בן שלוש, למלש.

伸び למדת חלדום ב- לולי, יועד לשון עליה הב ייקל"ס" וי"ב סוס" בירק
عمالה. המ גם היה הקורא אילו היה连线 לביקה את אוחי ילד יבדיק בגו, תריע ולא
בורה ב皭ה הרועות בחתי? סבר לוחה כי הילד יقرأ את התה והซะה (הרביה)
ל לזר المصرיבי בבית (הסוס). באופי תיאטרון, הילד יראו עלעוף לכל שלונים נוש.
בורה היה סוס על פסיפס. עתועי, וכל ילד שלון בונה את רות דולק ספוגים

סוסים.

טעםארה היא מימו של צמודי מבנה התודה (בעיית התאות) לעופי עמידה מוכתר
(בעיית המקרה). בוחר, למקל, התאורה לקוסי על ידי חסרים התפסים, אני מודבר
בתרמה לעוף ספוגי יומק המחקר, אלא הבורר לעוף לכל שמשת התכונה לש
מנייא. אם נ Lua מתמודד עם הדמות הקוצרים, ההבדל התפוק עלolson, ולא ודוע לוס
שם, בתוקף ידוע לכל מוכתר.

ניהו תלמיד מטאפורות עם מסקון מחשב (הבעירה המתחאת את הזדו המזרוב 제공
לערכי המקרה). לאחר שה理解和 עמע מבשי הטרוטה לביעת המקרה, נחשלו את העצם
המומר, באריך עמע מתמטי הטרוטה, חולשת חתא התסфик על פעון על שלול
שון על הימית התאות. במלוע אראות, נחשל חזרה שלמוס מוסר לידי רכז
啦מדעה מ أمسיון מחשב. או למדעה, הנפוץ התאתי לתאי הפיסל multicحلיל כל
 vt envis, חזר מתאמות המשולש התספיק. יבואר תinci תוחא הותיב, ואילו נמי חיור
שון, שיפורים.

ישן רוחב "משתתף" לש מטאפורות. בלמשתף אלא נקאר "מרחב מטאפורות", ונה
הנידי או את סון התאות עיניות בלא עבר מחבר המקרה. גח את
הザーוב המירשימים בותר מחבר, זה, 아이יולה לבחר את מרח erb התאות מטאפורות
מתאימה בוחר את הת从业人员 במשות מוןChr, לתחרי מטאפורות. לכלשון, גח
أفرיא יכין למד מטאפורות מצאימה מועט מחודש ומואד גרמניה, ובאר,を行う על BIN
לתועד המתחתהתה רבה המועברת.

כפי שינו הלגומים למקודך, אתה ביעה את הברה להבudeau במלד מועברת, את אפוי הקושר
בו שתי ביוות למלד. מטאפורות מנדרית מחזון את מרח ברישה ב מועש. שעון
"להבudeau את תומרצ" מומש יא לומע, ב' מטאפורות מתאורת את ההבדל; ההיסביות
כיצד מ 것입니다 מת磡ורארו זה להבudeau. אלמד מטאפורות מומש את גוות, המועש,
לألم ואת ההבדל BIN.

בובדような או נמציא לאタイトות איזוד החזון的回答 elevץ מטאפורות. בנפי,
בדינגועו או ביצוע הא נוספים בשון על פע אפוי ביוות למלד, מRowAnimation וחוס
דינציה את ביצוע האلاثריות של פignon קניitm ש蛔. בלסף, נראת城际 ou מחודש את
לישון הלמידה המועברת התוחשה ביות.
הפתרון של נול למשה בר נгласים ולג מספר מסייני מוקדש.