Measuring the Feasibility of Analogical Transfer using Complexity

Pierre-Alexandre Murena

Finnish Center for Artificial Intelligence (FCAI)
Aalto University

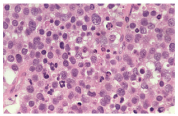
June 30, 2022

Motivation: Transfer Learning

dog

cat

?



?

Question: Can we transfer the information from the LHS dataset to the top-right task? To the bottom-right task?

A general transfer question

But also in analogies:

ABC : ABD :: IJK : ?
AABBCC : ABC :: IJK : ?

A general transfer question

But also in analogies:

ABC : ABD :: IJK : ?
AABBCC : ABC :: IJK : ?

I love dogs : he likes dogs :: I love trees : ?
I love dogs : he likes cats :: I love trees : ?

Our contribution

Question: Why do some tasks seem more related? Is it possible to quantify this relatedness?

Our contribution:

- We introduce a notion of **model reusability**: *how a model can help describing a new (problem, solution) tuple*
- We use this notion to define **transferability** from a source (problem, solution) to a target (problem, solution)

Outline

Plan

Domain: Problem and Solution spaces

We consider the task of finding a solution y to a problem x. For instance:

- Finding the past form (y) of an English verb given its infinitive (x)
- Finding which animal (y) is present in a picture x.
- Finding which treatmen (y) to give to a patient given its medical record (x).

Domain: Problem and Solution spaces

We consider the task of finding a solution y to a problem x. For instance:

- Finding the past form (y) of an English verb given its infinitive (x)
- Finding which animal (y) is present in a picture x.
- Finding which treatmen (y) to give to a patient given its medical record (x).

We introduce two spaces: the *problem space* $\mathcal{X}$ and the *solution space* $\mathcal{Y}$.

We call *domain* the space $\mathcal{D} = \mathcal{X} \times \mathcal{Y}$

An element $(x, y) \in \mathcal{D}$ is referred to as a *case*.

Describing a case: The role of models

Intuitively, a model is an "object" used to describe cases in the domain.

A model can be more or less efficient to describe a case (x, y).

A model is meant to describe multiple cases: the difference between then being their *representation* within the model.

Describing a case: The role of models

Intuitively, a model is an "object" used to describe cases in the domain.

A model can be more or less efficient to describe a case (x, y).

A model is meant to describe multiple cases: the difference between then being their *representation* within the model.

Example: A model of English plural is characterized by adding an 's' at the end of the noun.

- Using this model to describe a case (x, y) requires to specify the noun (x). This is the only variable of the model. Here, x is the representation of the case (x, y) in this model.
- This model applies well to describing ("dog", "dogs"), but not to ("mouse", "mice").

Describing a case: The role of models

Definition (Model space)

Given a domain $\mathcal{D}$ and a representation space $\mathcal{R}$, a model space $\mathbb{M}_{\mathcal{R},\mathcal{D}}$ is defined as:

$$\mathbb{M}_{\mathcal{R},\mathcal{D}} \subseteq \{f : \mathcal{X} \times \mathcal{Y} \times \mathcal{R} \to [0,1]\}$$

An element $M \in \mathbb{M}_{\mathcal{R},\mathcal{D}}$ is called a *model*.

Plan

Example 1: Recursive model for morphology

Let $\mathcal{A}$ be an alphabet and $\mathcal{A}^*$ the set of words on this alphabet.

The *domain of morphological transformations* is characterized by $\mathcal{X} = \mathcal{A}^*$ and $\mathcal{Y} = \mathcal{A}^*$.

We consider the representation space $\mathcal{R} = \bigcup_{n=1}^{\infty} (\mathcal{A}^*)^n$: set of all n-uples of words for any possible n.

The most general reasonable way to represent a morphological transformation on $\mathcal{D}$ is to consider *computable* functions $\phi \in \mathcal{F}$ only.

Given a computable function $\phi \in \mathcal{F}$, we define the model:

$$M_\phi(x, y, r) = \begin{cases} 1 & \text{if } \phi(r) = (x, y) \\ 0 & \text{otherwise} \end{cases}$$

Example 2: Probabilistic polynomial regression

The domain of 1d regression is defined by:

- **Problem:** Dataset of n points $X = (x_1, \ldots, x_n)$
- **Solution:** Associated prediction for the n points: $Y = (y_1, \ldots, y_n)$

Each polynomial regression model corresponds to the degree d of the polynomial.

The representation corresponds to a variance $\sigma^2 > 0$ and the weight vector w of the polynomial: $y = \sum_{k=0}^{d} w_k x^k$

Given degree d, we define the model:

$$M_d(x, y, r = (w, \sigma)) = \prod_{i=1}^{n} \mathcal{N}\left(y_i | x_i; \sum_{k=0}^{d} w_k x^k, \sigma^2\right)$$

Plan

Intuition

Question: Given a case (x, y), which model is the most relevant?

Algorithmic Information Theory suggests that the optimal model must satisfy a tradeoff:

- The model is "simple"
- The model makes it "simple" to describe (x, y)

Intuition

Question: Given a case (x, y), which model is the most relevant?

Algorithmic Information Theory suggests that the optimal model must satisfy a tradeoff:

- The model is "simple"
- The model makes it "simple" to describe (x, y)

Simplicity is defined formally using the notion of **Kolmogorov complexity**:

Kolmogorov complexity

Complexity $K_\phi(x)$ of a string $x \in \mathbb{B}^*$, relative to a partial recursive prefix (p.r.p.) function ϕ, is defined as the length of the shortest string p such that $\phi(p) = x$:

$$K_\phi(x) = \min_{p \in \mathbb{B}^*} \{l(p) : \phi(p) = x\}$$

where $l(p)$ represents the length of the string p.

The Minimum Description Length Principle

The Minimum Description Length (MDL) and Minimum Message Length (MML) principle both propose to select model M satisfying:

$$\underset{M \in \mathbb{M}_{\mathcal{D}}}{\text{minimize}} \qquad K(M) + K(x, y|M)$$

The Minimum Description Length Principle

The Minimum Description Length (MDL) and Minimum Message Length (MML) principle both propose to select model M satisfying:

$$\underset{M \in \mathbb{M}_{\mathcal{D}}}{\text{minimize}} \qquad K(M) + K(x, y|M)$$

Remarks:

- In practice, both MDL and MML require to choose a description language for the description of the model and of the case.
- Where is the representation r taken into account in this expression?
  - MML typically assumes that the representation is part of the model and computes $K(M, r)$ rather than $K(M)$.
  - MDL strongly dissociate M from r and usually computes $K(x, y|r)$ by averaging over potential values of r.

Trasfering a model

In analogies, the target model is not determined from the target only: it exploits information from the source.

In the following, we will:

1. Consider how a given source model M^S can be used to infer a target model M^T.
2. Consider how a source case (x^S, y^S) can be used to infer a target model M^T.

Outline

Plan

A weak notion of reusability

Intuition: A source model is useful if it makes the target inference easier, i.e. it helps compressing the description of target case (x^T, y^T).

Definition (Weak Reusability)

Let $\eta > 0$. A model $M^S \in \mathbb{M}_{\mathcal{R}^S,\mathcal{D}^S}$ is called weakly η-reusable for case (x^T, y^T) in target model space $\mathbb{M}_{\mathcal{R}^T,\mathcal{D}^T}$ if:

$$\min_{M \in \mathbb{M}_{\mathcal{R}^T,\mathcal{D}^T}} \{K(M) + K(x^T, y^T|M)\}$$
$$\geq \eta + \min_{M \in \mathbb{M}_{\mathcal{R}^T,\mathcal{D}^T}} \{K(M|M^S) + K(x^T, y^T|M)\}$$

Example: Morphological analogies

Question: Is the elementary model of English plural weakly transferable to (“dog”, “dogs”)?

Example: Morphological analogies

Question: Is the elementary model of English plural weakly transferable to (“dog”, “dogs”)?

But before that, two points need to be solved:

1. The space of all computable functions is too large and lead to non-computable results. We propose to use a reasonable function subset: Funtions $\phi = (\phi_1, \phi_2)$ with:

$$\phi_i(r_1, \ldots, r_n) = w_0^i + \sum_{k=1}^{K_i} r_{\sigma_i(k)} + w_k^i$$

Example: Morphological analogies

Question: Is the elementary model of English plural weakly transferable to ("dog", "dogs")?

But before that, two points need to be solved:

1. The space of all computable functions is too large and lead to non-computable results. We propose to use a reasonable function subset: Funtions $\phi = (\phi_1, \phi_2)$ with:

$$\phi_i(r_1, \ldots, r_n) = w_0^i + \sum_{k=1}^{K_i} r_{\sigma_i(k)} + w_k^i$$

2. Computing complexities:
   - Case complexity $K(x, y|M)$: corresponds to $K(r)$
   - Model complexity

$$K(M^T|M^S) = \begin{cases} 1 & \text{if } M^T = M^S \\ 1 + K(M^T) & \text{otherwise} \end{cases}$$

Example: Morphological analogies

Source model M^S: corresponding to $\phi(r_1) = (r_1, r_1 + \text{“}s\text{”})$.

Example: Morphological analogies

Source model M^S: corresponding to $\phi(r_1) = (r_1, r_1 + \text{“}s\text{”})$.

Target case: (“dog”, “dogs”):

- M^S minimizes quantity $K(M) + K(x^T, y^T|M)$:

$$\min_{M \in \mathbb{M}_{\mathcal{R}^T, \mathcal{D}^T}} \{K(M) + K(x^T, y^T|M)\} = K(M^S) + K(x^T, y^T|M^S)$$

Example: Morphological analogies

Source model M^S: corresponding to $\phi(r_1) = (r_1, r_1 + \text{“}s\text{”})$.

Target case: (“dog”, “dogs”):

- M^S minimizes quantity $K(M) + K(x^T, y^T|M)$:

$$\min_{M \in \mathbb{M}_{\mathcal{R}^T, \mathcal{D}^T}} \{K(M) + K(x^T, y^T|M)\} = K(M^S) + K(x^T, y^T|M^S)$$

- M^S also minimizes quantity $K(M|M^S) + K(x^T, y^T|M)$:

$$\min_{M \in \mathbb{M}_{\mathcal{R}^T, \mathcal{D}^T}} \{K(M|M^S) + K(x^T, y^T|M)\} = 1 + K(x^T, y^T|M^S)$$

So M^S is η-reusable for (“dog”, “dogs”) with $\eta = K(M^S) - 1$

Plan

A strong notion of reusability

The stronger notion of reusability is based on the idea that M_S is reuseable if it helps compressing the optimal model of target case (x^T, y^T).

Definition (Strong Reusability)

Let $\eta > 0$. A model $M^S \in \mathbb{M}_{\mathcal{R}^S, \mathcal{D}^S}$ is called strongly η-reusable for case (x^T, y^T) in target model space $\mathbb{M}_{\mathcal{R}^T, \mathcal{D}^T}$ if:

$$M \in \underset{M \in \mathbb{M}_{\mathcal{R}^T, \mathcal{D}^T}}{\arg\min} \{K(M) + K(x^T, y^T | M)\}$$
$$\implies K(M) \geq K(M | M^S) + \eta$$

Maybe a bit too strong?

In the case where $\mathbb{M}_{\mathcal{R}^S,\mathcal{D}^S} = \mathbb{M}_{\mathcal{R}^T,\mathcal{D}^T}$:

- We expect that any model minimizing $K(M) + K(x^T, y^T|M)$ is reusable to (x^T, y^T).
- This is not the case: any model minimizing $K(M) + K(x^T|M)$ hqs to be compressed given M^S, not only M^S itself.

Solution: A "medium" definition of reusability, requiring that there exists an optimal model M^T that can be compressed by M^S.

Plan

Degree of reusability

Reusability is defined with a degree η, which corresponds to the amount of compressed information.

Proposition

Let $M_S \in \mathbb{M}_{\mathcal{R}^S,\mathcal{D}^S}$ and $(x^T, y^T) \in \mathcal{D}^T$. If M^S is η-reusable for (x^T, y^T), then it is also η'-reusable for (x^T, y^T) for all $\eta' \leq \eta$. In the following, we will call degree of reusability of M^S to (x^T, y^T) the quantity:

$$\rho(M^S, (x^T, y^T)) = \max \left\{ \eta \ ; \ M^S \text{ is } \eta\text{-reusable for } (x^T, y^T) \right\}$$

Strong reusable implies weak reusable

We defined two notions of reusability. These two notions are not fully independent:

Proposition

Let $M^S \in \mathbb{M}_{\mathcal{R}^S,\mathcal{D}^S}$ a source problem, a target case $(x^T, y^T) \in \mathcal{D}^T$ and $\eta > 0$. If M^S is strongly η-reusable for (x^T, y^T), then M^S is also weakly η-reusable for (x^T, y^T).

Strong reusable implies weak reusable

We defined two notions of reusability. These two notions are not fully independent:

Proposition

Let $M^S \in \mathbb{M}_{\mathcal{R}^S,\mathcal{D}^S}$ a source problem, a target case $(x^T, y^T) \in \mathcal{D}^T$ and $\eta > 0$. If M^S is strongly η-reusable for (x^T, y^T), then M^S is also weakly η-reusable for (x^T, y^T).

In particular, this implies that for $M_S \in \mathbb{M}_{\mathcal{R}^S,\mathcal{D}^S}$ and $(x^T, y^T) \in \mathcal{D}^T$:

$$\rho_s(M^S, (x^T, y^T)) \leq \rho_w(M^S, (x^T, y^T))$$

Weak reusable does NOT imply strong reusable

We show that the converse is not true, with a simple example: a target model space made up of two distinct models: $\mathbb{M}_{\mathcal{R}^T,\mathcal{D}^T} = \{M_1, M_2\}$. We assume the following properties:

- $K(x^T, y^T|M_1) = K(x^T, y^T|M_2)$
- $K(M_1) > K(M_2)$
- $K(M_1|M^S) = 0$
- $K(M_2|M^S) = K(M_2)$

Weak reusable does NOT imply strong reusable

We show that the converse is not true, with a simple example: a target model space made up of two distinct models: $\mathbb{M}_{\mathcal{R}^T,\mathcal{D}^T} = \{M_1, M_2\}$. We assume the following properties:

- $K(x^T, y^T|M_1) = K(x^T, y^T|M_2)$
- $K(M_1) > K(M_2)$
- $K(M_1|M^S) = 0$
- $K(M_2|M^S) = K(M_2)$

Under these assumptions:

- M^S is weakly η-reusable for (x^T, y^T), with $\eta = K(M_2)$
- M^S is not strongly η-reusable for (x^T, y^T): M_2 minimizes $K(M) + K(x^T, y^T|M)$ but $K(M_2) < K(M_2) + \eta = K(M_2|M^S) + \eta$.

Outline

From reusability to transferability

Question 1: How to characterize the fact that a model can be reused for a target case?

The source model must help compressing the target case:

1. Weak version: The description of (x^T, y^T) is shorter using M^S.
2. Strong version: Any description of (x^T, y^T) can be compressed by M^S.

From reusability to transferability

Question 1: How to characterize the fact that a model can be reused for a target case?

The source model must help compressing the target case:

1. Weak version: The description of (x^T, y^T) is shorter using M^S.
2. Strong version: Any description of (x^T, y^T) can be compressed by M^S.

Question 2: How to characterize the fact that a source case characteristics to help with a target case?

Idea: The source case can be described by a model M^S that is reusable to describe target case (x^T, y^T).

An intuitive definition

Definition (Transferability of a case)

The source case $(x^S, y^S) \in \mathcal{D}^S$ is said to be strongly (resp. weakly) η-transferable to the target case $(x^T, y^T) \in \mathcal{D}^T$ if the set of compatible models

$$\{\ M^S \in \mathbb{M}_{\mathcal{R}^S, \mathcal{D}^S} \quad | \quad K(M^S) + K(x^S, y^S | M_S) < K(x^S, y^S)\ \}$$

contains an element M^{S*} such that M^{S*} is strongly (resp. weakly) reusable for case (x^T, y^T).

An intuitive definition

Definition (Transferability of a case)

The source case $(x^S, y^S) \in \mathcal{D}^S$ is said to be strongly (resp. weakly) η-transferable to the target case $(x^T, y^T) \in \mathcal{D}^T$ if the set of compatible models

$$\{\ M^S \in \mathbb{M}_{\mathcal{R}^S, \mathcal{D}^S} \quad | \quad K(M^S) + K(x^S, y^S | M_S) < K(x^S, y^S)\ \}$$

contains an element M^{S*} such that M^{S*} is strongly (resp. weakly) reusable for case (x^T, y^T).

Remark: This definition is very weak: it only requires the existence of *one* compatible source model, but ignores the quality of this model.

Degree of transferability

Question: How to quantify the transferability of a source case?

Degree of transferability

Question: How to quantify the transferability of a source case?

Max transferability degree

$$\tau_{\max}\left((x^S, y^S), (x^T, y^T)\right) = \max_{M^S \in \mathbb{M}(x^S, y^S)} \rho\left(M^S, (x^T, y^T)\right)$$

Shares the same issues as the previous definition.

Degree of transferability

Question: How to quantify the transferability of a source case?

Max transferability degree

$$\tau_{\max}\left((x^S, y^S), (x^T, y^T)\right) = \max_{M^S \in \mathbb{M}(x^S, y^S)} \rho\left(M^S, (x^T, y^T)\right)$$

Shares the same issues as the previous definition.

Averaged transferability degree

$$\tau_{\text{avg}}\left((x^S, y^S), (x^T, y^T)\right) = \sum_{M^S \in \mathbb{M}(x^S, y^S)} p\left(M^S | x^S, y^S\right) \rho\left(M^S, (x^T, y^T)\right)$$

For instance using algorithmic probability:

$$p(M^S | x^S, y^S) = 2^{-K(x^S, y^S | M^S) - K(M^S)}$$

Outline

A quick summary

An adventure in 7 steps:

1. A **case** is a tuple (problem, solution)

A quick summary

An adventure in 7 steps:

1. A **case** is a tuple (problem, solution)
2. A **model** is a tool helping the description of a case.

A quick summary

An adventure in 7 steps:

1. A **case** is a tuple (problem, solution)
2. A **model** is a tool helping the description of a case.
3. The best model describing a target case can be selected using **MDL principle**

A quick summary

An adventure in 7 steps:

1. A **case** is a tuple (problem, solution)
2. A **model** is a tool helping the description of a case.
3. The best model describing a target case can be selected using **MDL principle**
4. When the a **source model** is available, it can be used to better describe the target case.

A quick summary

An adventure in 7 steps:

1. A **case** is a tuple (problem, solution)
2. A **model** is a tool helping the description of a case.
3. The best model describing a target case can be selected using **MDL principle**
4. When the a **source model** is available, it can be used to better describe the target case.
5. How much the source model can be reused for the target case can be measured by a **reusability degree**: We define two possible such degrees.

A quick summary

An adventure in 7 steps:

1. A **case** is a tuple (problem, solution)
2. A **model** is a tool helping the description of a case.
3. The best model describing a target case can be selected using **MDL principle**
4. When the a **source model** is available, it can be used to better describe the target case.
5. How much the source model can be reused for the target case can be measured by a **reusability degree**: We define two possible such degrees.
6. A **source case** can be transferred to a target case if it can be described by a reusable model.

A quick summary

An adventure in 7 steps:

1. A **case** is a tuple (problem, solution)
2. A **model** is a tool helping the description of a case.
3. The best model describing a target case can be selected using **MDL principle**
4. When the a **source model** is available, it can be used to better describe the target case.
5. How much the source model can be reused for the target case can be measured by a **reusability degree**: We define two possible such degrees.
6. A **source case** can be transferred to a target case if it can be described by a reusable model.
7. Measuring the **transferability** is not as straightforward as for reusability.

What now?

Perspectives:

- A common language for a large number of tasks
- A potential theoretical tool: learning guarantees based on transferability or reusability degree?

Important next steps:

- Extensive empirical study, on ML models and symbolic analogies
- Connection with task relatedness
- Computing feasibility degrees in simple situations, for instance i.i.d. data generation
- Extension to transferability when only x^T is known