

Transfer Learning

Eduardo Morales, Hugo Jair Escalante

INAOE

Contenido

- 1 Introducción
- 2 Transfer en Aprendizaje por Refuerzo
- 3 Transfer en Redes Bayesianas
- 4 Transfer en Árboles de Decisión
- 5 Transfer en Redes Neuronales
- 6 Temas relacionados y preguntas abiertas

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Antecedentes

- Los algoritmos de aprendizaje han logrado avances muy importantes en los últimos años
- En general suponen que los ejemplos de entrenamiento y prueba tienen los mismos atributos y provienen de la misma distribución
- Si la distribución cambia, normalmente se tienen que reconstruir los modelos
- En algunas aplicaciones es muy caro o no se pueden recolectar los ejemplos de entrenamiento para reconstruir un modelo
- Aprendizaje por transferencia o *transfer learning* puede ayudar en tales casos

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Transfer Learning

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- TL puede servir cuando los datos:
 - Son escasos
 - Están desactualizados
- TL permite que los dominios, tareas y distribuciones sean diferentes en los ejemplos de entrenamiento y prueba
- Para funcionar de alguna forma debe de existir alguna conexión entre las tareas de aprendizaje

Transfer Learning

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Definición: Dado un dominio (D_f) y tarea (T_f) fuentes, un dominio (D_o) y tarea (T_o) objetivos, aprendizaje por transferencia busca mejorar el aprendizaje de la función objetivo usando conocimiento de D_f y T_f

Transfer Learning

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

- Definición: Dadas algunas/una observación(es) del dominio fuente $m^S \in \mathbb{N}^+$ y tareas (i.e., $\{(\mathcal{D}_{S_i}, \mathcal{T}_{S_i}) | i = 1, \dots, m^S\}$), y algunas/una observación(es) del dominio objetivo $m^T \in \mathbb{N}^+$ y tareas (i.e., $\{(\mathcal{D}_{T_j}, \mathcal{T}_{T_j}) | j = 1, \dots, m^T\}$), aprendizaje por transferencia utiliza el conocimiento implicado en el(los) dominio(s) fuente para mejorar el desempeño de las funciones de decisión aprendidas $f^{T_j} (j = 1, \dots, m^T)$ en el(los) dominio(s) objetivo

Transfer Learning

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- En TL se consideran tres aspectos fundamentales:
 - ¿Qué transferir?
 - ¿Cómo transferir?
 - ¿Cuándo transferir?
- El transferir conocimiento no siempre es útil y puede dañar el desempeño de los algoritmos (*negative transfer*)
- Aunque se tenga cierta similaridad entre dominios no implica que se pueda evitar la transferencia negativa

Transferencia Negativa

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

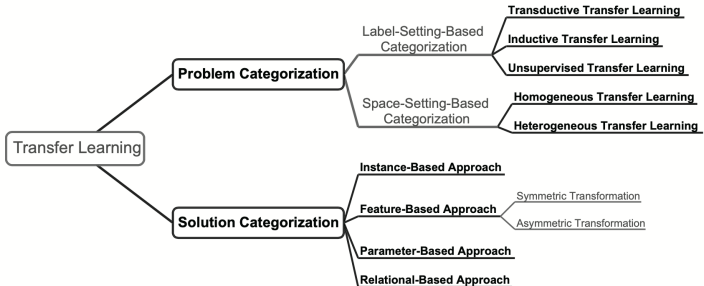
Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Ocurre cuando el dominio y tarea fuentes reducen el desempeño del aprendizaje en la tarea objetivo
- ¿Qué falta?:
 - ¿Cómo evitar transferencia negativa?
 - Estudiar “transferibilidad” entre dominios fuente y objetivo
 - ¿Se puede transferir parte del dominio?
 - Transferir entre dominios y tareas múltiples con diferentes atributos (*heterogeneous transfer learning*)
 - Escalarlo a aplicaciones más grandes

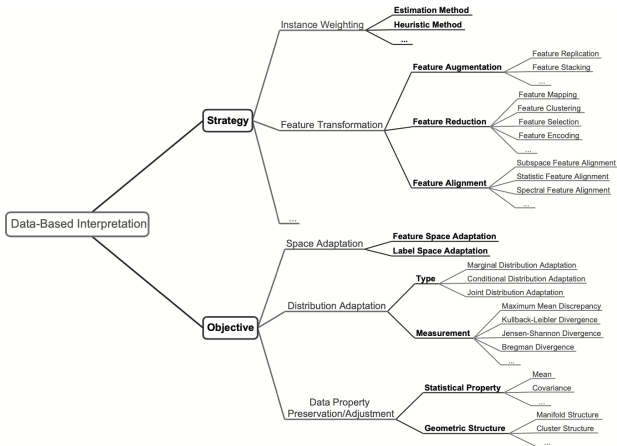
Algoritmos



Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Algoritmos



Introducción

Transfer en Aprendizaje por Refuerzo

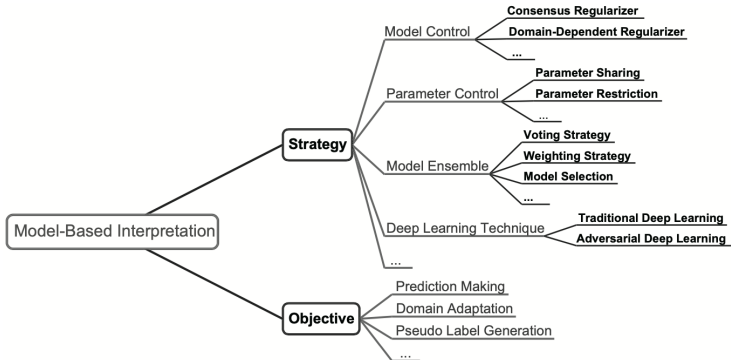
Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Algoritmos



Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Categorización de Algoritmos

Podemos catalogar a los algoritmos de TL en tres:

- *Inductive transfer learning*: La tarea fuente y objetivo son diferentes: (i) Se tienen muchos datos del dominio fuente, (ii) se tienen pocos datos o ninguno del dominio objetivo
- *Transductive transfer learning*: Las tareas de la fuente y objetivo son las mismas, pero los dominios son diferentes: (i) Los espacios de atributos son diferentes en la fuente y en el objetivo, (ii) los atributos son iguales pero las probabilidades marginales son diferentes ($P(X_f) \neq P(X_o)$)
- *Unsupervised transfer learning*: No se tienen datos etiquetados en ningún caso, las tareas son diferentes pero de alguna forma relacionadas

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Categorización de Algoritmos

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Existen muchas formas de categorizar a los algoritmos de aprendizaje, una de ellas es en:

- Homogéneos: Los dominios tienen el mismo espacio de atributos ($X^S = X^T$ y $Y^S = Y^T$)
- Heterogéneos: Los dominios tienen espacios de atributos diferentes ($X^S \neq X^T$ y/o $Y^S \neq Y^T$)

Inductive Transfer Learning

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Opciones de conocimiento a transferir:

- Transferir instancias: En este caso los espacios de atributos son iguales, aunque no necesariamente todos los ejemplos son útiles
- Transferir la representación de los atributos: Aprender una representación de baja dimensionalidad que se pueda compartir entre tareas
- Transferir parámetros: Ya sean parámetros o distribuciones de los algoritmos de aprendizaje utilizados
- Transferir conocimiento relacional: Relaciones equivalentes entre dominios

Transfer Learning basado en Datos

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- *Instance weighting*: Asignar pesos a los datos de las tareas fuentes en la función de pérdida. Se pueden usar medidas que utilizan kernels, medidas de basadas en distribuciones (e.g., KL)
- *Feature transformation*: Se puede: (i) Transformar los datos a un espacio latente, (ii) reducir la diferencia entre las distribuciones, (iii) agregar atributos, (iv) mapear entre atributos, (v) agrupar atributos, (vi) seleccionar atributos, (vii) usar *autoencoders*, (viii) alinear atributos, etc.

Transfer Learning basado en Modelos

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Se pueden adaptar modelos de diferentes fuentes
- Se pueden transferir o acotar (hiper)parámetros de las tareas fuentes
- Se pueden construir ensambles de modelos
- Usar modelos de aprendizaje profundo congelando pesos, usando autoencoders, GANs, etc.

Transfer Learning

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- En esta clase sólo vamos a ver algunos aspectos (y ejemplos particulares) de *Inductive Transfer Learning*
- Nos vamos a enfocar en transferencia en cuatro áreas:
 - Aprendizaje por Refuerzo
 - Redes Bayesianas
 - Árboles de decisión
 - Redes neuronales

Transfer en Aprendizaje por Refuerzo

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

RL:

- No requiere de un modelo del ambiente
- El agente aprende solo
- Converge a la política óptima

Pero:

- El aprendizaje es lento
- Pocos desarrollos en ambientes complejos con variables continuas
- Poco desarrollo en reutilizar políticas

Transfer en Aprendizaje por Refuerzo

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- En Aprendizaje por Refuerzo (RL), TL usa conocimiento de una o más tareas fuente para aprender una o más tareas objetivo
- Los pasos a seguir son:
 - Dada una tarea objetivo, seleccionar la(s) tarea(s) fuente(s) de las cuales se va a transferir
 - Aprender cómo se relacionan las tareas fuente y objetivo
 - Transferir de forma efectiva conocimiento de la(s) fuente(s) a la(s) tarea(s) objetivo

Métricas de Desempeño

Se pueden usar diferentes métricas para evaluar los beneficios de TL:

- *Jumpstart*: El desempeño inicial de un agente en la tarea objetivo
- Desempeño asintótico: El desempeño final del agente en la tarea objetivo
- Recompensa total: La recompensa total acumulada
- Razón de transferencia: La división entre la recompensa total acumulada con y sin transferencia
- Tiempo para alcanzar el umbral: El tiempo requerido para alcanzar un nivel de desempeño pre-establecido

Introducción

Transfer en Aprendizaje por Refuerzo

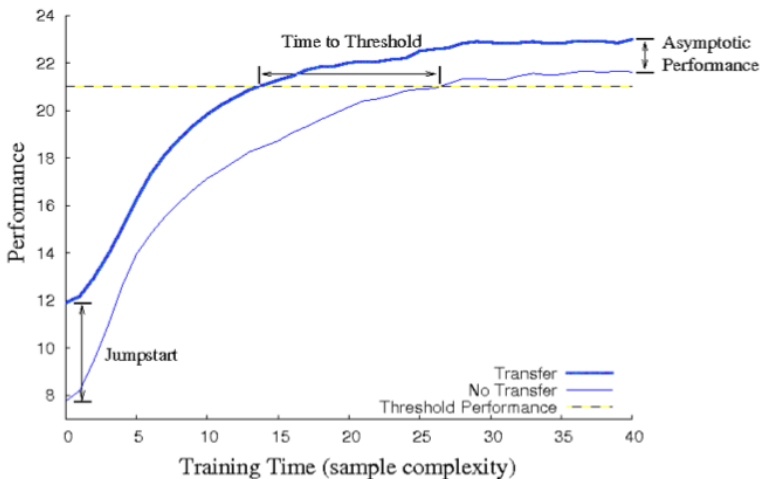
Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Efectos de TL en RL



Selección de Diferentes Fuentes

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- ¿Qué suposiciones se hacen en cuanto a las posibles diferencias entre el fuente y el objetivo?
- ¿Cómo garantizar no transferir de tareas irrelevantes?
- ¿Se hace selección de diferentes fuentes?
- ¿Se hace un mapeo entre tareas? ¿Está dado o es automático? (*Inter-task mapping*)
- En general, en la mayoría de los algoritmos el mapeo lo da el usuario!

Conocimiento a Transferir

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Puede ser conocimiento de bajo nivel, e.g., tuplas $\langle s, a, r, s' \rangle$, funciones de valor (V, Q), la política (π), el modelo de transición ($p(s' | s, a)$)
- Pueden ser de más alto nivel, e.g., qué acciones usar en determinado momento, políticas parciales u *options* (macros), distribuciones *a priori*, atributos relevantes para aprender, *reward shaping*, definición de subtareas, modelos causales, ...

Conocimiento a Transferir

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Se puede transferir entre tareas que tengan diferentes: funciones de transición ($p(s' | s, a)$), espacio/variables de estados (S), estados iniciales/metras (s_0/s_f), conjuntos de acciones (A), funciones de recompensa (R), ...
- Relacionado: *Multi-Task Learning* suponen que todos los problemas son de la misma distribución, e.g., aprender varios péndulos invertidos.

Aplicaciones

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Vamos a describir aplicaciones en donde transferimos:

- Hiper-parámetros
- Tuplas transformadas

En un contexto de Procesos Gaussianos

Procesos Gaussianos

Introducción

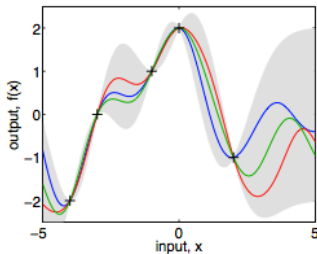
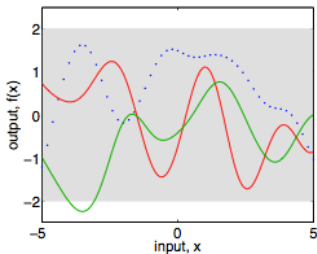
Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

- Distribución Gaussiana multivariada

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu)\right)$$

- Un proceso gaussiano es una generalización a un número infinito de variables: $\mathcal{GP}(m(\cdot), k(\cdot, \cdot))$

Motivación



Aunque parece peor trabajar con dimensionalidad infinita, lo que se calcula se hace en dimensiones finitas

Procesos Gaussianos

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

- En particular, se debe cumplir que:

$$p\left(\begin{bmatrix} \vec{h} \\ \vec{h}_* \end{bmatrix} \mid X, X_*\right) \sim \mathcal{N}\left(\vec{0}, \begin{bmatrix} k(X, X) & k(X, X_*) \\ k(X_*, X) & k(X_*, X_*) \end{bmatrix}\right)$$

- Podemos considerar ruido Gaussiano:

$$p\left(\begin{bmatrix} \vec{\epsilon} \\ \vec{\epsilon}_* \end{bmatrix}\right) \sim \mathcal{N}\left(\vec{0}, \begin{bmatrix} \sigma^2 I & \vec{0} \\ \vec{0}^T & \sigma^2 I \end{bmatrix}\right)$$

Procesos Gaussianos

- Suponemos que son independientes, por lo que su suma también lo es:

$$\begin{bmatrix} \vec{y} \\ \vec{y}_* \end{bmatrix} | X, X_* = \begin{bmatrix} \vec{h} \\ \vec{h}_* \end{bmatrix} + \begin{bmatrix} \vec{\epsilon} \\ \vec{\epsilon}_* \end{bmatrix} \sim \mathcal{N} \left(\vec{0}, \begin{bmatrix} k(X, X) + \sigma^2 I & k(X, X_*) \\ k(X_*, X) & k(X_*, X_*) + \sigma^2 I \end{bmatrix} \right)$$

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Procesos Gaussianos

- Usando las reglas de condicionamiento Gaussiano, se obtiene:

$$\vec{y}_* | \vec{y}, X, X_* \sim \mathcal{N}(\mu^*, \Sigma^*)$$

donde:

$$\mu^* = K(X_*, X)(K(X, X) + \sigma^2 I)^{-1} \vec{y}$$

$$\Sigma^* = K(X_*, X_*) + \sigma^2 I - K(X_*, X)(K(X, X) + \sigma^2 I)^{-1} K(X, X_*)$$

- Su solución implica invertir una matriz simétrica del tamaño de los ejemplos

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

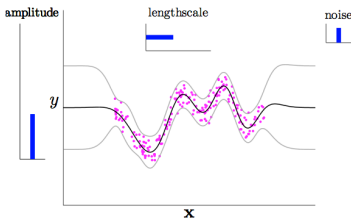
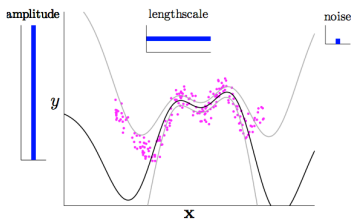
Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Kernel e Hiperparámetros

$$k(x, x') = \alpha_0^2 \exp\left(-\frac{1}{2} \left(\frac{x - x'}{\lambda}\right)^2\right)$$



Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Cálculo de los Hiperparámetros

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

- Recordando la definición de una distribución gaussiana multivariable:

$$p(x|\mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu)\right)$$

- $P(y|x, \theta)$ sigue una distribución gaussiana multivariable con media cero y covarianza de $K + \sigma_n^2 I$ (si consideramos ruido)

$$\log p(y|x, \theta) = -\frac{1}{2} y^T (K + \sigma_n^2 I)^{-1} y - \frac{1}{2} \log |K + \sigma_n^2 I| - \frac{n}{2} \log 2\pi$$

Hiperparámetros

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Los hiperparámetros nos determinan las posibles distribuciones
- Para obtener los hiperparámetros podemos derivar con respecto a θ
- Se sigue un proceso basado en gradiente (es un problema de optimización no-convexo), por ejemplo, basado en gradiente conjugado o quasi-Newton
- Se puede caer en mínimos locales

PILCO

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- PILCO (Probabilistic Inference for Learning Control)
- $P(s_t | s_{t-i}, a_{t-i}) = \mathcal{GP}(m, k)$
- PILCO usa diferencias como predictor: $\Delta_t = s_t - s_{t-i}$ y un kernel exponencial cuadrado

PILCO

Interact with environment, apply π to obtain tuples

REPEAT

Infer transition function distribution f from
tuples and hyper-parameters θ

REPEAT

Evaluate policy π over f

Get V^π

Improve π (Updating parameters ψ)

UNTIL convergence

$\pi \leftarrow \pi(\psi)$

Interact with environment, apply π to obtain more tuples

Learn θ_{ρ_i} from all tuples

UNTIL task learned

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Motivación

VIDEO DE PILCO

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

QTL

- La idea es transferir hiperparámetros (sesgo sobre la distribución de posibles funciones de transición)
- Hay que definir cómo hacer la transferencia
- Ésta se hace de forma gradual:
 - Usando un factor de olvido

$$\Theta_0 = \Theta_s$$

$$\Theta_i = \gamma\Theta_{i-1} + (1 - \gamma)\Theta_{p_i}, i > 0$$

- Actualización Bayesiana

$$p(\Theta_{p_k}) \sim \mathcal{N}(\mu_p, \sigma_p^2) \quad \sigma_k^2 = \frac{\sigma_p^2 \sigma_{k-1}^2}{\sigma_p^2 + \sigma_{k+1}^2}$$

$$p(\Theta | \Theta_{p_k}) \sim \mathcal{N}(\mu_k, \sigma_k^2) \quad \sigma_{k=0}^2 = \frac{1}{n_{source}}$$

$$\mu_k = \sigma_k^2 \left(\frac{\mu_{k-1}}{\sigma_{k-1}^2} + \frac{\mu_p}{\sigma_p^2} \right) \quad \sigma_p^2 = \frac{1}{n_{target}}$$

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

SST

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Síntesis de Tuplas:

- Aprender una función de transición en la tarea objetivo
- Aprender/usar la función de transición de la tarea original
- Aprender una función de diferencias
- En espacios poco explorados, generar ejemplos artificiales usando los ejemplos de la tarea original y la función de diferencias

Dónde y Cuántas Tuplas

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

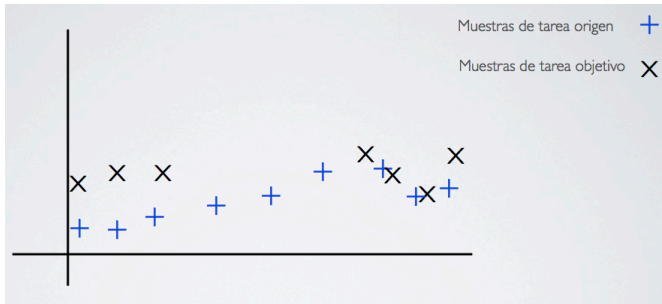
Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- En lugares desconocidos (alejados en $\langle s, a \rangle$): Si ya tengo ejemplos, no necesito generar
- Generar hasta completar el número de ejemplos usados en la tarea original
- Mantener ese número fijo \Rightarrow ir reduciendo el número de ejemplos de la tarea original conforme se explora la tarea objetivo

Proceso



Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Proceso

Introducción

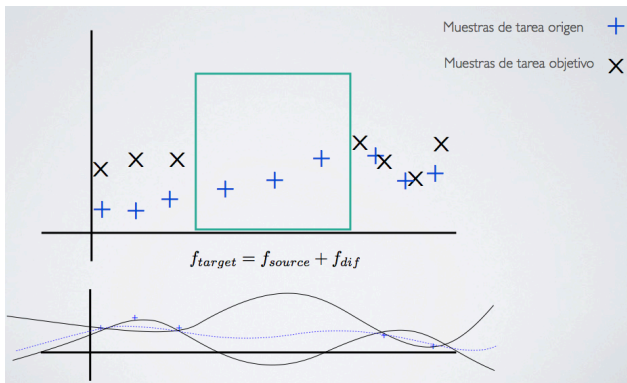
Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas



Proceso

Introducción

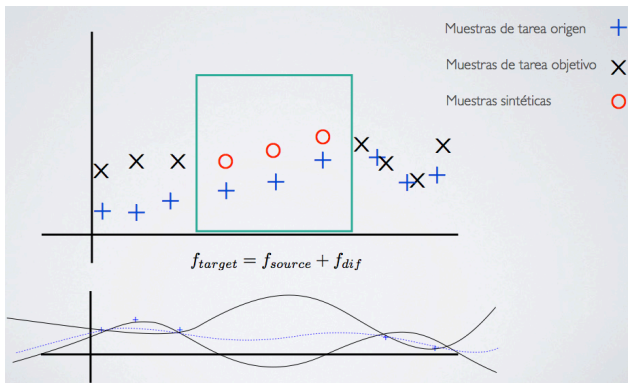
Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas



Proceso

Introducción

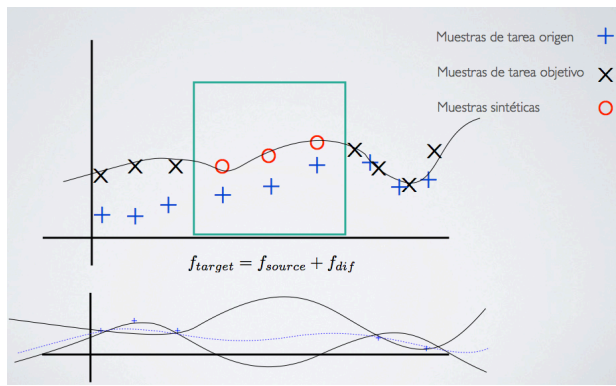
Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

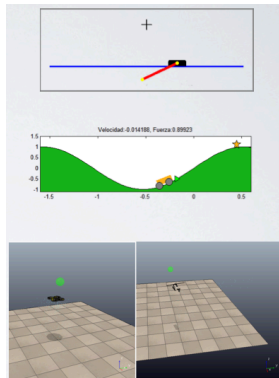
Temas relacionados y preguntas abiertas



Experimentos

Se probó en 3 dominios:

- 1 Péndulo invertido (clásico)
- 2 Auto en la montaña (transferencia negativa)
- 3 De cuadricóptero a helicóptero



Experimentos QTL

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Probar:

- Transferencia desde diferentes variantes
- Usar hiperparámetros de la tarea original
- Usar la política de la tarea original
- Diferentes valores de γ ($\gamma = 0 \Rightarrow$ PILCO)
- Enfoque Bayesiano

Resultados

Introducción

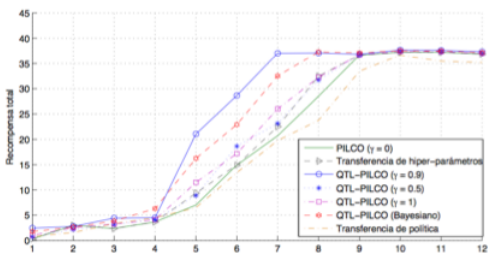
Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

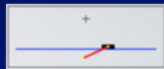
Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

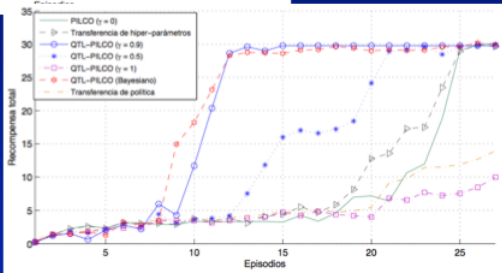


QTL



1.8 x masa original

4 x masa original



Resultados

Introducción

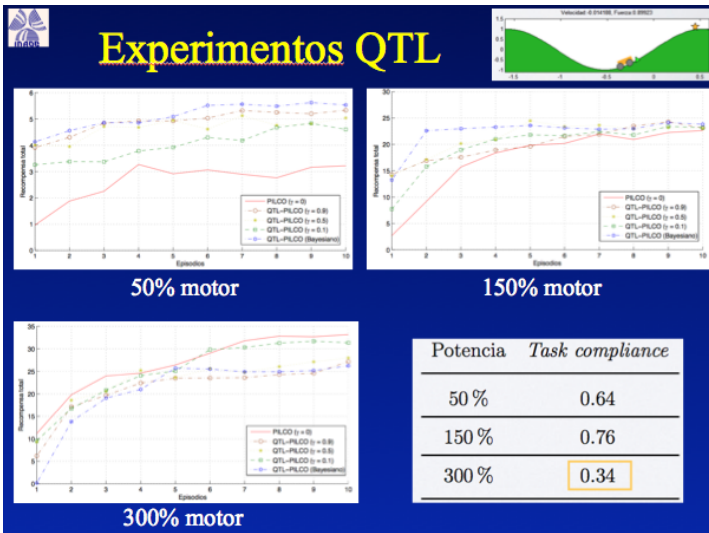
Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

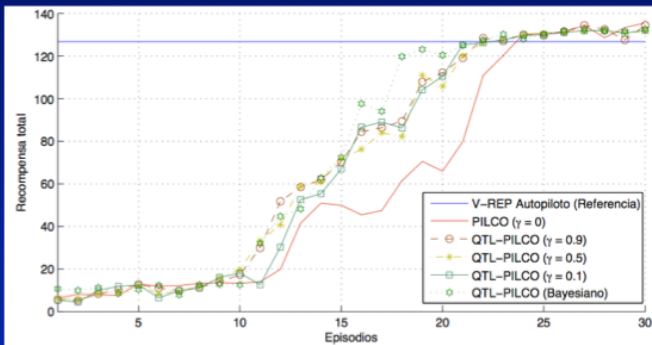
Temas relacionados y preguntas abiertas



Resultados



Experimentos QTL



Experimentos SST

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- PILCO
- Transferir todas la tuplas
- Transferir usando un filtro simple
- Transferir usando el filtro de Lazaric
- Todas las tuplas + SST
- Filtro simple + SST
- Filtro Lazaric + SST

Resultados SST

Introducción

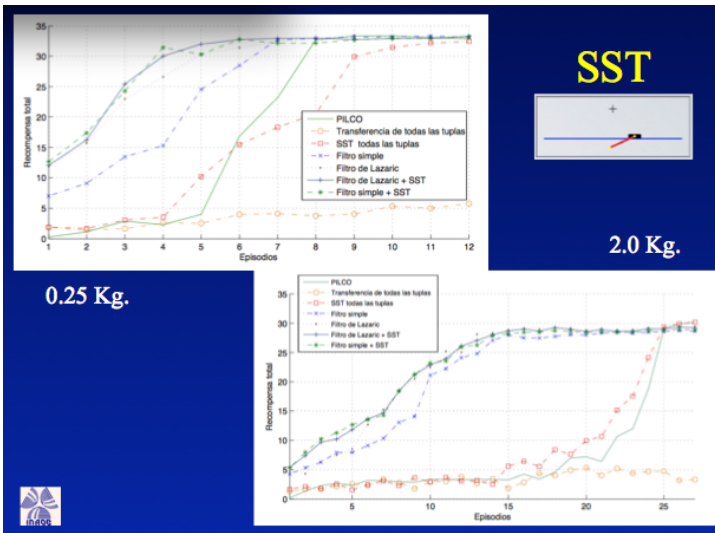
Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

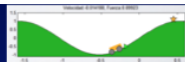
Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas



Resultados SST

Experimentos SST

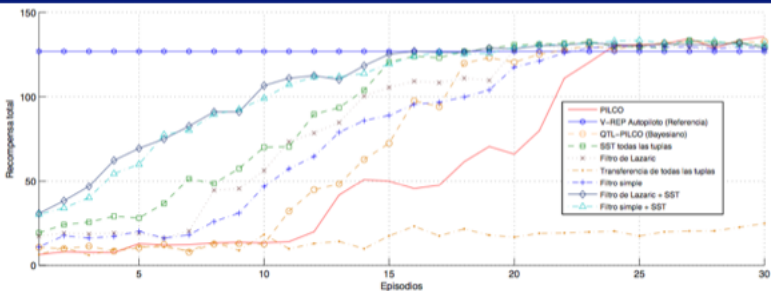


Algoritmo	50 %	150 %	300 %
PILCO (sin transferencia)	26.39	173.72	265.48
QTL (bayesiano)	51.21	222.60	206.60
Transferencia de todas las tuplas	17.56	52.90	64.05
Filtro de Lazaric	46.78	191.28	223.68
Síntesis de tuplas	50.57	232.47	252.73
Filtro simple	45.81	186.32	202.70
Filtro de Lazaric + SST	54.70	289.75	276.39
Filtro simple + SST	51.34	249.45	268.78



Resultados SST

Experimentos SST



Helicóptero a Cuadróptero

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

VIDEO

Transfer en Redes Bayesianas

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Las Redes Bayesianas han probado ser un formalismo adecuado para tratar información con incertidumbre
- Para aprender una Red Bayesiana necesitamos determinar la estructura de la red y las tablas de dependencia condicional
- Con pocos datos lo que se aprende es poco confiable
- Vamos a ver un mecanismo de transferencia usando como base el algoritmo PC

Transfer en Redes Bayesianas

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- PC aprende un esqueleto (grafo no dirigido) y después determina la dirección de las ligas
- Para determinar el esqueleto empieza con una red no dirigida completamente conectada y determina la independencia condicional de cada par de variables dados subconjuntos de variables
- TL en BNs: Algoritmo para aprendizaje de red bayesiana incorporando información de bases de datos auxiliares

TL en BN

- Extensión del algoritmo PC, para el caso en donde tenemos pocos datos para la tarea objetivo y tenemos muchos datos de tareas fuentes
- Las diferencias con PC son en la forma en que se evalúan las pruebas de independencia
- Para cada par de variables X, Y la medida de independencia es una combinación lineal de la estimación de la tarea objetivo con la tarea del dominio auxiliar más cercana
- La combinación lineal es pesada por factores que determinan una medida de confianza

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Algoritmo de PC

Require: Set of variables \mathbf{X} , Independence test I

- 1: Initialize a complete undirected graph G'
- 2: $i=0$
- 3: **repeat**
- 4: **for** $X \in \mathbf{X}$ **do**
- 5: **for** $Y \in ADJ(X)$ **do**
- 6: **for** $S \subseteq ADJ(X) - \{Y\}, |S| = i$ **do**
- 7: **if** $I(X, Y | S)$ **then**
- 8: Remove the edge $X - Y$ from G'
- 9: **end if**
- 10: **end for**
- 11: **end for**
- 12: **end for**
- 13: $i=i + 1$
- 14: **until** $|ADJ(X)| \leq i, \forall X$
- 15: Orient edges in G'

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Algoritmo con TL

Initialize a complete undirected graph G' , Set $i = 0$

repeat

for $X \in \mathbf{X}$ **do**

for $Y \in \text{ADJ}(X)$ **do**

for $S \subseteq \text{ADJ}(X) - \{Y\}$, $|S| = i$ **do**

Find the most similar auxiliary domain, k , and its similarity measure Sk_{XY}

Determine the confidence measures $\alpha(X, Y|S)$ for target and auxiliary domains

Obtain the combined independence measure $I_F(X, Y | S)$

if $I_F(X, Y | S)$ **then**

Remove the edge $X - Y$ from G'

end if

end for

end for

end for

$i = i + 1$

until $|\text{ADJ}(X)| \leq i, \forall X$

Orient edges in G'

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

TL en BN

- La confianza en la prueba de independencia de PC entre X y Y , dado S depende de la cantidad de datos:

$$\alpha(X, Y|S) = 1 - \frac{\log N}{2N} \times T$$

donde $T = |X| \times |Y| \times |S|$

- Para medir la similaridad entre tareas se usa una medida de similaridad global (Sg_{Dj}) y una local ($Sl_{Dj}(X, Y)$)

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Similitud entre Tareas

- Medida global: $Sg_{Dj} = dep_j + ind_j$
 - dep_j = número de dependencias condicionales comunes entre la tarea objetivo y la tarea auxiliar j
 - ind_j = número de independencias condicionales comunes entre todos los pares de variables en la tarea objetivo y la tarea auxiliar j
- Medida local:

$$Sl_{Dj}(X, Y) = \begin{cases} 1,0 & \text{If } I_0(X, Y|S) = I_{Dj}(X, Y|S) \\ 0,5 & \text{If } I_0(X, Y|S) \neq I_{Dj}(X, Y|S) \end{cases}$$

Donde: $I_0(X, Y|S)$ = prueba de independencia en tarea objetivo y $I_{Dj}(X, Y|S)$ = prueba en la tarea auxiliar j

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Similitud entre Tareas

- Se combinan estas dos medidas como:

$$Sk_{XY}^* = Sg_{Dk} \times Sl_{Dk}(X, Y)$$

- La medida combinada de independencia se calcula como una combinación lineal pesada de las medidas de independencia de las tareas objetivo y fuentes:

$$I_F(X, Y|S) = \alpha_0(X, Y|S) \times \text{sgn}(I_0(X, Y|S)) + Sk_{XY}^* (\alpha_{D_{XY}}(X, Y|S) \times \text{sgn}(I_{D_{XY}}(X, Y|S)))$$

- $\text{sgn}(I) = +1$ si la prueba de independencia es positiva y -1 de otra forma
- $\alpha_0(X, Y|S)$ es la medida de confianza en el dominio objetivo
- $\alpha_{D_{XY}}(X, Y|S)$ es la medida de confianza en la tarea auxiliar más similar para $\{X, Y\}$ condicionado en S

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Aprendizaje de Parámetros

- Combinar las CPTs (tablas de probabilidad condicional)
- Unificar variables:
 - Mismas variables: Fácil
 - Más padres en las estructuras auxiliares: Aplicar marginalización (sumar sobre todos los valores de las variables extra)
 - Menos padres en las estructuras auxiliares: Duplicar los valores de las CPTs para todos los valores de las variables extra
 - Una combinación de las 2 anteriores: Primero marginalizar y luego duplicar

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Tablas de Probabilidad Condicional

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

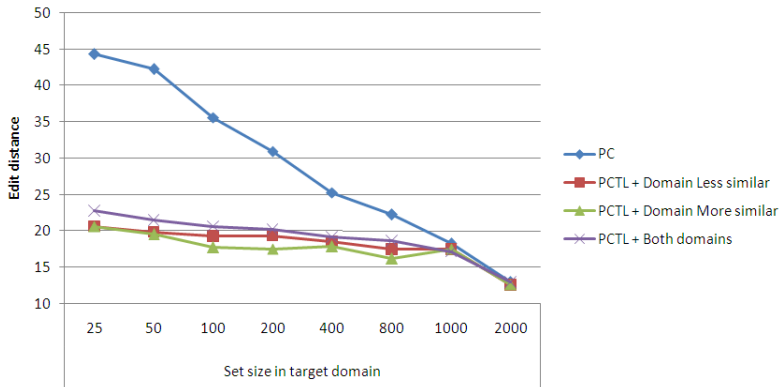
Temas relacionados y preguntas abiertas

Ya que se tienen las mismas variables existen varias formas de combinar los valores de las CPTs:

- Lineal: $P(X) = k \times \sum_{i=1}^n w_i P_i(X)$
- Logarítmica: $P(X) = k \times \prod_{i=1}^n P_i(X)^{w_i}$
- Distance Based Lineal Pool: Obtiene las probabilidades promedio de todas las BDs descontadas por su nivel de confianza (f_i): $\bar{p} = k \sum_{i=1}^n (f_i \times p_i)$
- Local Linear Pool: Usa sólo las más parecidas a las locales y las pesa de acuerdo a su confianza:

$$p'_{target} = f_{target} \times p_{target} + (1 - f_{target}) \times \bar{p}_{local}$$

Resultados cambiando la cantidad de datos



Introducción

Transfer en Aprendizaje por Refuerzo

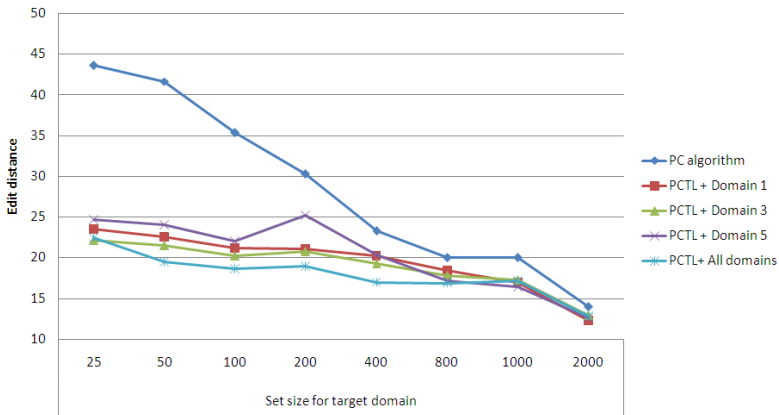
Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Resultados cambiando datos y número de fuentes



Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

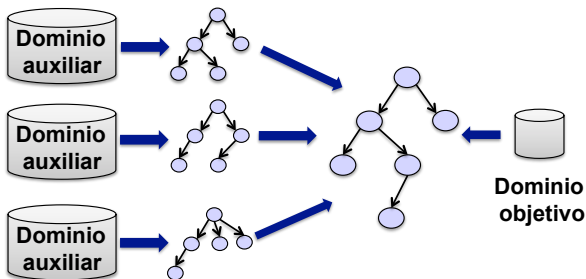
Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

TL en Redes Bayesianas de Nodos Temporales

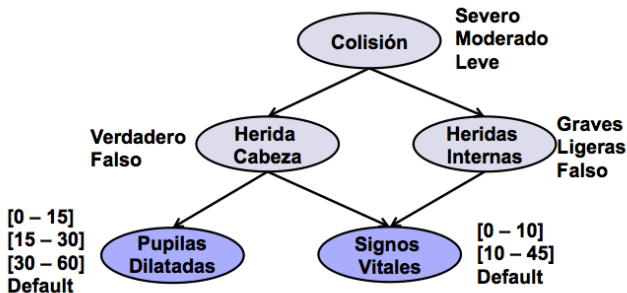
- Una red bayesiana de nodos temporales sirve para modelar procesos dinámicos que están caracterizados por cambios irreversibles



Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Esquema General



- Nodos Instantáneos
- Nodos Temporales

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

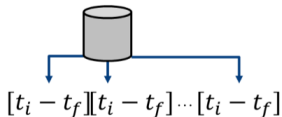
Transfer en Árboles de Decisión

Transfer en Redes Neuronales

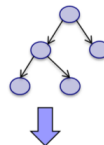
Temas relacionados y preguntas abiertas

Determinación de Intervalos

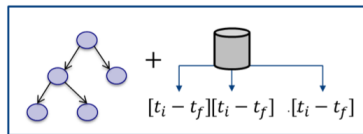
1. Aprendizaje de Intervalos



2. Aprendizaje Estructural



3. Refinamiento de Intervalos



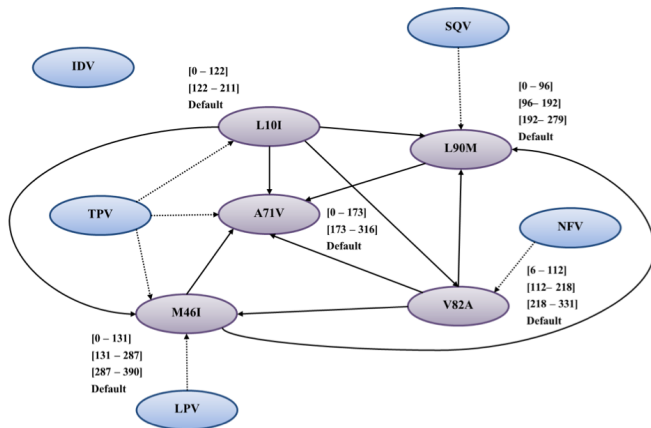
4. Aprendizaje Paramétrico

$P(\mathbf{RBNT})$

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Resultados en HIV



Transferencia Usando Árboles de Decisión

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Se tienen datos relacionados a estrés y sus modelos (árboles) para varios personas
- Se tiene un usuario nuevo, con pocos datos, y se quiere dar una estimación aceptable de su nivel de estrés
- Idea: Transferir datos/modelos de otros usuarios para mejorar la predicción del nuevo

Transferencia Usando Árboles de Decisión

Se probaron 4 esquemas de transferencia:

- 1 Naïve: Encuentra el modelo más parecido, transfiere los datos (los junta con los del nuevo usuario) e induce un nuevo árbol
- 2 Umbral: Encuentra los modelos más parecidos (arriba de cierto umbral), transfiere los datos de los usuarios parecidos e induce un nuevo árbol
- 3 Muestreo: Encuentra los k modelos más parecidos y muestrea sus datos proporcionalmente a su medida de similaridad
- 4 Ensamble: Usa los k modelos más parecidos, junto con el modelo del usuario y construye un ensamble pesado por la similaridad

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Medida de Similaridad entre Árboles

- Sean D_i y D_j dos árboles con H y K hojas que clasifican n ejemplos.
- Formamos la matriz:
 $M = [m_{hk}]$, $h = 1, \dots, H$ y $k = 1, \dots, K$ donde m_{hk} es el número de ejemplos que pertenecen a la h -ésima hoja de D_i y a la k -ésima hoja de D_j y $m_{h0} = \sum_{k=1}^K m_{hk}$,
 $m_{0k} = \sum_{h=1}^H m_{hk}$.
- La medida de disimilaridad se define como:

$$d(D_i, D_j) = \sum_{h=1}^H \alpha_h (1 - s_h) \frac{m_{h0}}{n} + \sum_{k=1}^K \alpha_k (1 - s_k) \frac{m_{0k}}{n}$$

donde las m miden la similaridad predictiva y las α y s miden la similaridad estructural

Medida de Similaridad entre Árboles

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

- s_{hk} mide la similaridad entre hojas tomando en cuenta las clases y ejemplos:

$$s_{hk} = \frac{m_{hk}c_{hk}}{\sqrt{m_{h0}m_{0k}}} \quad k = 1, \dots, K$$

donde $c_{hk} = 1$ si la h -ésima hoja de D_i tiene la misma clase que la k -ésima hoja de D_j , y $c_{hk} = 0$ si no.

- $s_h = \text{máx}\{s_{hk}, k = 1, \dots, K\}$

Medida de Similaridad entre Árboles

- $\alpha_h = q - p + 1$ mide la disimilaridad entre hojas, dependiendo de la longitud de los caminos diferentes (p), y la profundidad en donde difieren (q)
- El valor máximo de $d(D_i, D_j)$ se alcanza cuando la diferencia entre los árboles es máxima y su similaridad de predicción es cero
- El factor de normalización es:

$$\text{máx } d(D_i, D_j) = \sum_{h=1}^H \alpha_h \frac{m_{h0}}{n} + \sum_{k=1}^K \alpha_k \frac{m_{0k}}{n}$$

donde α_h es la longitud del nodo raíz a la h -ésima hoja

- La medida de disimilaridad normalizada es entonces:

$$d_n = \frac{d(D_i, D_j)}{\text{máx } d(D_i, D_j)}$$

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Resultados

Subject ID	No transfer	Transfer learning approaches				All data
		Naive	Threshold	Sampling weighted	Ensemble weighted	
S01	72.86	61.43	72.86	64.28	87.14	78.57
S04	81.25	71.88	81.25	65.62	73.44	84.37
S05	70.69	65.52	65.52	75.86	68.97	86.20
S08	57.41	50.00	57.41	57.40	85.19	55.55
S09	57.69	73.08	73.08	65.38	38.46	76.92
S10	44.89	51.02	51.02	55.10	63.27	71.42
S11	65.45	74.55	65.45	76.36	65.45	72.72
S12	55.93	54.24	54.24	55.93	62.71	62.71
S14	51.56	48.44	51.56	53.12	90.00	82.81
S16	61.11	62.96	61.11	62.96	90.74	74.07
S18	70.27	62.16	62.16	70.27	81.08	75.67
S19	60.00	53.33	60.00	70.00	90.00	90.00
S23	53.33	50.00	53.33	38.33	38.33	58.33
S24	67.14	67.14	67.14	70.00	70.00	71.42
S25	85.71	83.67	85.71	83.67	85.71	89.79
S28	57.35	63.24	63.24	60.29	95.59	77.94
S29	62.07	44.83	62.07	67.24	36.21	79.31
S30	42.86	53.57	53.57	48.21	66.07	78.57
Avg.±Std.dev.	62.09±11.0	60.61±10.4	63.37±9.5	63.33±10.6	71.58±18.2	75.91±9.4

Transferencia en Redes Neuronales Profundas

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

- Recientemente las redes neuronales profundas han acaparado la atención de los investigadores por sus resultados exitosos, principalmente en tareas de visión y lenguaje natural
- Su entrenamiento, sin embargo, requiere de una gran cantidad de datos y de *hardware* especializado
- Transfer puede ayudar a aprender modelos con menos datos y menos tiempo de entrenamiento

Transferencia en Redes Neuronales Profundas

- La forma más utilizada, principalmente en imágenes, es utilizar una red pre-entrenada en ImageNet
- Se han propuesto varias opciones:
 - Hacer *fine tuning* en toda la red
 - Congelar la mayoría de los pesos y sólo cambiar los de las últimas capas
 - Cambiar pesos de acuerdo a su profundidad (pocos cambios al principio y muchos al final)
 - *Knowledge Distillation* para construir una red más pequeña
 - Transferir parte de la red

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

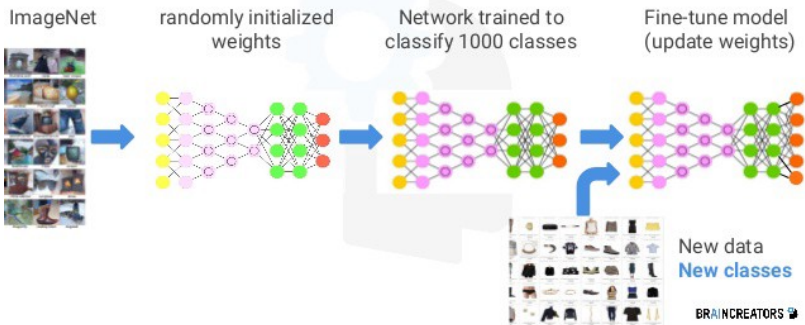
Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Fine Tuning en toda la red

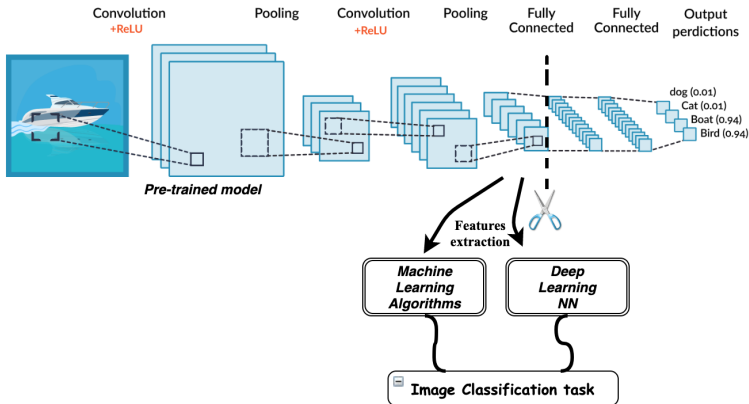
Transfer Learning



Introducción

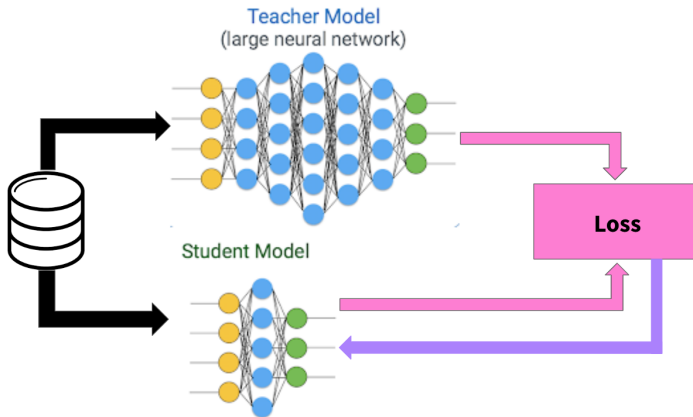
Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Sólo reentrenar la parte final de la red



Se puede usar cualquier clasificador que tome el vector de la red (antes de la capa densa) como atributos (e.g., SVM)

Knowledge Distillation: Mismas clases



Introducción

Transfer en
Aprendizaje
por Refuerzo

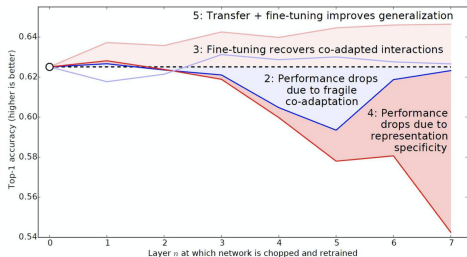
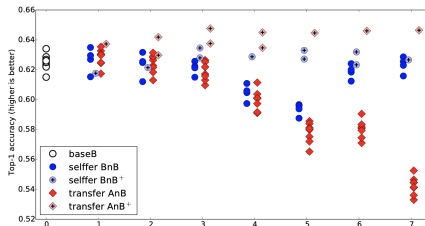
Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

¿Qué tanto se puede transferir?



- 2) BnB: Primeras n capas se congelan
- 3) BnB+: Igual a 2) pero se hace finetuning en toda la red
- 4) AnB: Primeras n capas se congelan entrenadas en A
- 5) AnB+: Igual a 4) pero se hace finetuning en toda la red

Domain Adaptation

Opciones:

- 1 (i) Suponer que haciendo *fine tuning* se puede resolver, usando la información de las clases (lo más común), (ii) tratando de alinear las distribuciones (e.g., con Kullback-Liebler), (iii) ajustando la arquitectura de la red
- 2 Usando una estrategia de adversarios, (i) usando una GAN con los ejemplos fuente para generar ejemplos parecidos a los del concepto objetivo o (ii) aprender una representación abstracta de los ejemplos fuente y mover los ejemplos objetivo a esa representación
- 3 Usando métodos basados en reconstrucción, (i) usando auto-encoders apilados (SAE) o (ii) basados en errores en la reconstrucción

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Ejemplos recientes de Domain Adaptation

Introducción

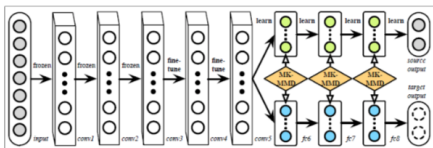
Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

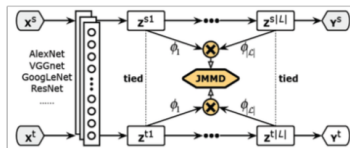
Transfer en Árboles de Decisión

Transfer en Redes Neuronales

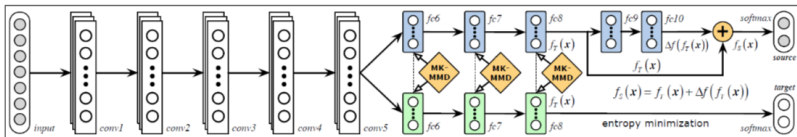
Temas relacionados y preguntas abiertas



(a) The Deep Adaptation Network (DAN) architecture



(b) The Joint Adaptation Network (JAN) architecture



(c) The Residual Transfer Network (RTN) architecture

¿Cómo seleccionar de dónde transferir?

- Si tenemos varios modelos ya entrenados, ¿cómo saber cuál es relevante para la tarea que queremos resolver?
- Ideas:
 - 1 Transfiere varios modelos fuente en diferentes tareas objetivo y evalúa sus resultados
 - 2 Extrae características del efecto de instancias de la tarea objetivo en los modelos fuente (e.g., entropía)
 - 3 Construye un regresor, que dadas las características de un modelo con instancias objetivo, prediga su desempeño al transferir

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

¿Cómo seleccionar de dónde transferir?

Introducción

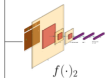
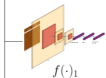
Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

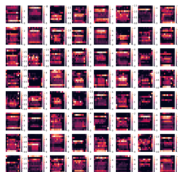
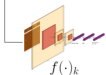
Transfer en Árboles de Decisión

Transfer en Redes Neuronales

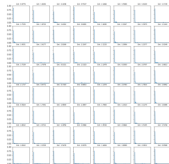
Temas relacionados y preguntas abiertas



...

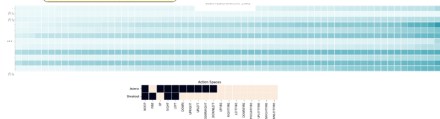


a) Evaluate a sample of the target game in each pre-trained model

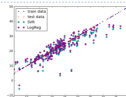


b) Discretize each convolutional kernel

For each model



c) Use the entropies of each convolutional kernel and the action spaces in one-hot encoding as features of the regression model



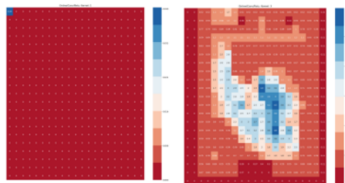
d) Evaluate the features of c) in the regressor

Source	Prediction
$f(\cdot)_1$	1.23
$f(\cdot)_2$	1.10
...	...
$f(\cdot)_k$	0.20

e) Select the pre-trained model with the maximum predicted performance

¿Qué transferir?

- No necesariamente todo el modelo es útil para transferir
- La idea es sólo transferir los kernels convolucionales que aportan la mayor cantidad de información (medida en términos de la entropía en los mapas de atributos)



(a) Uniform output

(b) Output with diverse values

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Resultados: (i) Regresor, (ii) Transferir todo, (iii) Transferencia selectiva

Introducción

Transfer en Aprendizaje por Refuerzo

Transfer en Redes Bayesianas

Transfer en Árboles de Decisión

Transfer en Redes Neuronales

Temas relacionados y preguntas abiertas

Table 5 Predicted performance by the regressor and real performance, the task that outperform train form crash (higher than 1) are presented in bolds, those that obtain neutral performance are marked in italics.

Target	Predicted	Performance	Error
Crazy Climber	1.0978	1.0029	-0.0948
Boxing	1.0449	1.0289	-0.0160
Freeway	1.4855	1.4816	-0.0039
Pong	1.0338	1.0061	-0.0277
Robotank	1.0318	1.0348	0.0030
Up & Down	1.0593	<i>0.8818</i>	-0.1775
Atlantis	1.2189	5.2701	4.0512
Breakout	1.1610	1.2153	0.0543
Gopher	1.1790	<i>0.9359</i>	-0.2431
Space Invaders	1.1204	0.9413	-0.1791
Video Pinball	1.6838	1.4186	-0.2652
Star Gunner	1.2596	1.1168	-0.1428
Assault	0.9664	0.6229	-0.3435
Beam Rider	1.1842	0.9235	-0.2607
Fishing Derby	1.0577	1.0080	-0.0497
Hero	1.0896	1.1281	0.0385
Kangaroo	1.0586	1.1914	0.1328
Name This Game	1.0339	0.9048	-0.1291
Road Runner	1.1326	1.0024	-0.1302
Tutankham	1.3395	1.1152	-0.2243
Asterix	1.1096	0.8833	-0.2263
Demon Attack	1.0260	0.8055	-0.2205
Ice Hockey	1.0833	<i>0.9595</i>	-0.1238
Krull	1.0840	1.4111	0.3271
Tennis	0.9880	<i>1.0600</i>	0.0120
Enduro	1.0563	1.0236	-0.0327
Jamesbond	1.1033	1.1386	0.0386
Kung-Fu Master	1.1915	<i>0.9806</i>	-0.2109
Qbert	1.1279	<i>0.9628</i>	-0.1651
Time Pilot	1.0586	1.1382	0.0796

Table 6 Results of training with transfer, we train three independent experiments in each of the thirty games (normalized), the used metrics are: Jumpstart (JS), Maximum score (MAX), Transfer Ratio (TR) and Final Score (FS).

Target	JS	MAX	TR	FS
Crazy Climber	0.4718	1.0029	-0.0216	0.9138
Boxing	1.9513	1.0289	0.0305	0.9847
Freeway	0.0084	1.4816	0.4440	1.4663
Pong	8.3750	1.0061	0.1657	0.9818
Robotank	0.7720	1.0374	-0.0892	0.5722
Up & Down	0.2920	0.8818	-0.0744	0.7405
Atlantis	1.3628	5.2701	2.4858	4.6987
Breakout	1.0072	1.2153	0.1887	1.2176
Gopher	3.2273	0.9359	-0.0291	0.8407
Space Invaders	0.4572	0.9413	0.0164	0.9955
Video Pinball	1.2340	1.4186	0.0447	1.1866
Star Gunner	0.9125	1.1168	-0.0608	1.0980
Assault	0.6535	0.6229	-0.3435	0.5827
Beam Rider	0.8025	0.9235	-0.0492	0.9650
Fishing Derby	1.0785	1.0080	-0.0245	1.1337
Hero	0.0099	1.1281	0.1967	1.1025
Kangaroo	0.7236	1.1914	-0.3019	1.8442
Name This Game	1.3211	0.9524	-0.0387	0.8811
Road Runner	2.1365	1.0024	0.0609	0.9704
Tutankham	0.5900	1.1996	0.0083	0.9359
Asterix	0.8008	0.8833	-0.1930	0.7036
Demon Attack	0.4454	0.8055	-0.2136	0.6977
Ice Hockey	0.5327	0.9595	0.0687	1.4186
Krull	1.1481	1.4111	-0.0499	0.8583
Tennis	3.7172	1.0000	-0.5247	0.5862
Enduro	99.2887	1.0236	0.1043	0.9355
Jamesbond	2.1415	1.1386	0.5333	1.0689
Kung-Fu Master	370.9185	0.9806	-0.0894	0.6554
Qbert	1.4574	0.9628	-0.0667	1.0513
Time Pilot	2.0570	1.1382	0.0322	0.9310
Mean	16.9964	1.1872	0.0736	1.0988

Table 9 Results of training with selective transfer, we train three independent experiments in each of the thirty games (normalized), the used metrics are: Jumpstart (JS), Maximum score (MAX), Transfer Ratio (TR) and Final Score (FS).

Target	JS	MAX	TR	FS
Crazy Climber	0.4718	0.9660	-0.0134	0.9180
Boxing	1.9513	0.9920	0.1602	1.0106
Freeway	0.0084	1.4694	-0.3894	0.9873
Pong	8.3750	0.9116	-0.1208	1.7556
Robotank	0.7720	0.9581	-0.1940	1.1356
Up & Down	0.2920	0.9292	-0.0730	1.7052
Atlantis	1.3628	1.5248	-0.5833	0.0215
Breakout	1.0072	1.5135	0.0047	0.7876
Gopher	3.2273	0.7222	-0.2104	0.9231
Space Invaders	0.4572	0.8774	-0.0562	0.8556
Video Pinball	1.2340	1.2914	-0.1804	1.1239
Star Gunner	0.9125	0.7853	-0.2408	0.7354
Assault	0.6535	0.5759	-0.3734	0.7750
Beam Rider	0.8025	0.8421	-0.1479	0.8488
Fishing Derby	1.0785	1.0133	-0.0688	0.9854
Hero	0.0099	0.7722	-0.2594	0.5546
Kangaroo	0.7236	1.0342	-0.2669	0.8007
Name This Game	1.3211	0.8798	-0.1163	1.1441
Road Runner	2.1365	1.0944	0.0656	1.1102
Tutankham	0.5900	1.1083	-0.1970	0.7249
Asterix	0.8008	0.7431	-0.2031	0.8806
Demon Attack	0.4454	0.8200	-0.2372	0.5905
Ice Hockey	1.6852	0.9500	-0.2573	0.4331
Krull	1.1481	1.3316	-0.1109	1.1897
Tennis	3.7172	1.0000	-0.5306	1.0759
Enduro	99.2887	0.9703	-0.0332	0.9274
Jamesbond	2.1415	0.9542	0.0993	0.6968
Kung-Fu Master	370.9185	0.8639	-0.1595	0.4515
Qbert	2.0570	0.6980	-0.2928	0.6246
Time Pilot	1.4574	1.1172	-0.0763	1.1535
Mean	11.7588	0.9903	-0.1524	0.8613

Algunas Técnicas Relacionadas

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

- Lifelong learning/Continual learning/Curriculum learning
- Concept Drift
- Imitation learning
- Human advice/Shaping
- Multi-view learning
- Multi-task learning

Preguntas Abiertas

- Si queremos hacerlo incremental ¿cómo evitar el *catastrophic forgetting*?
- Si se tiene un modelo de una tarea fuente, ¿cómo modificarlo para la tarea objetivo?
- ¿Se pueden modificar las tareas fuentes automáticamente para mejorar el aprendizaje?
- ¿Se pueden tomar ideas de *theory revision/refinement* para hacer un mapeo entre tareas?
- ¿La transferencia se podría utilizar para mejorar el mecanismo de exploración del agente en la tarea objetivo?
- Transferencia negativa: No hay trabajo que defina de forma confiable cuándo una transferencia va a ser negativa

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Referencias

- S.J. Pan and Q. Yang (2010). A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359.
- Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, Qing He (2020). A Comprehensive Survey on Transfer Learning. [arXiv:1911.02685](https://arxiv.org/abs/1911.02685)
- Mei Wang, Weihong Deng (2018). Deep Visual Domain Adaptation: A Survey. [arXiv:1802.03601](https://arxiv.org/abs/1802.03601)
- Jason Yosinski, Jeff Clune, Yoshua Bengio, Hod Lipson (2014). How transferable are features in deep neural networks? [arXiv:1411.1792](https://arxiv.org/abs/1411.1792)
- C.E. Rasmussen and C.K.I. Williams (2005). *Gaussian Processes for Machine Learning*. MIT Press.

Introducción

Transfer en
Aprendizaje
por RefuerzoTransfer en
Redes
BayesianasTransfer en
Árboles de
DecisiónTransfer en
Redes
NeuronalesTemas
relacionados y
preguntas
abiertas

Referencias

- E. Omar, E. Muñoz de Cote, E.F. Morales (2014). Transfer Learning for Continuous State and Action Spaces. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)* 28(7).
- E. Munoz de Cote, E.O. Garcia, E.F. Morales (2016). Transfer Learning by Prototype Generation in Continuous Spaces. *Adaptive Behavior* 24(6): 464-478.
- R. Luis-Velázquez, L.E. Sucar, E.F. Morales (2010). Inductive Transfer for Learning Bayesian Networks. *Machine Learning* 79 (1-2): 227–255.
- L.J. Fiedler, L.E. Sucar, E.F. Morales (2015). Transfer Learning for Temporal Nodes Bayesian Networks. *Applied Intelligence* 43 (3): 578-597.

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas

Referencias

- A. Maxhuni, P. Hernandez-Leal, L.E. Sucar, V. Osmani, E.F. Morales, O. Mayora (2016). Stress Modelling and Prediction in Presence of Scarce Data. *Journal of Biomedical Informatics* 63: 344-356.
- J. García-Ramírez, E.F. Morales, H.J. Escalante (2021). Source tasks selection for transfer deep reinforcement learning: a case of study on Atari games. *Neural Computing and Applications* pp. 1-13

Introducción

Transfer en
Aprendizaje
por Refuerzo

Transfer en
Redes
Bayesianas

Transfer en
Árboles de
Decisión

Transfer en
Redes
Neuronales

Temas
relacionados y
preguntas
abiertas