Description Logics for Commonsense Concept Combination

Valentina Gliozzi, Antonio Lieto, Gian Luca Pozzato[1], Gioele Tallone, Alberto Valese

[1]Dipartimento di Informatica, Università di Torino, Italy

CILC 2024

DLs with Typicality and Probabilities

Outline

- Introduction to Description Logics of Typicality
- Extensions with Probabilities
  - Logic of concept combination
- Original contributions:
  - Dynamic generation of knowledge
  - Learning inclusions and probabilities

DLs with Typicality and Probabilities

Outline

- Introduction to Description Logics of Typicality
- Extensions with Probabilities
  - Logic of concept combination
- Original contributions:
  - Dynamic generation of knowledge
  - Learning inclusions and probabilities

DLs with Typicality and Probabilities

Outline

- Introduction to Description Logics of Typicality
- Extensions with Probabilities
  - Logic of concept combination
- Original contributions:
  - Dynamic generation of knowledge
  - Learning inclusions and probabilities

DLs with Typicality and Probabilities

Outline

- Introduction to Description Logics of Typicality
- Extensions with Probabilities
  - Logic of concept combination
- Original contributions:
  - Dynamic generation of knowledge
  - Learning inclusions and probabilities

DLs with Typicality and Probabilities

Outline

- Introduction to Description Logics of Typicality
- Extensions with Probabilities
  - Logic of concept combination
- Original contributions:
  - Dynamic generation of knowledge
  - Learning inclusions and probabilities

DLs with Typicality and Probabilities

Outline

- Introduction to Description Logics of Typicality
- Extensions with Probabilities
  - Logic of concept combination
- Original contributions:
  - Dynamic generation of knowledge
  - Learning inclusions and probabilities

DLs with typicality

What are they?

- Non-monotonic extensions of Description Logics for reasoning about prototypical properties and inheritance with exceptions
  - Basic idea: to extend DLs with a typicality operator $\mathbf{T}$
  - $\mathbf{T}(C)$ singles out the "most normal" instances of the concept C
  - semantics of $\mathbf{T}$ defined by a set of postulates that are a restatement of Lehmann-Magidor axioms of rational logic $\mathbf{R}$

Basic notions

- A KB comprises assertions $\mathbf{T}(C) \sqsubseteq D$
- $\mathbf{T}(\mathit{TeenAger}) \sqsubseteq \mathit{InstagramUsers}$ means "normally, teen-agers use Instagram"
- $\mathbf{T}$ is nonmonotonic
  - $C \sqsubseteq D$ does not imply $\mathbf{T}(C) \sqsubseteq \mathbf{T}(D)$

DLs with typicality

What are they?

- Non-monotonic extensions of Description Logics for reasoning about prototypical properties and inheritance with exceptions
  - Basic idea: to extend DLs with a typicality operator $\mathbf{T}$
  - $\mathbf{T}(C)$ singles out the "most normal" instances of the concept C
  - semantics of $\mathbf{T}$ defined by a set of postulates that are a restatement of Lehmann-Magidor axioms of rational logic $\mathbf{R}$

Basic notions

- A KB comprises assertions $\mathbf{T}(C) \sqsubseteq D$
- $\mathbf{T}(\mathit{TeenAger}) \sqsubseteq \mathit{InstagramUsers}$ means "normally, teen-agers use Instagram"
- $\mathbf{T}$ is nonmonotonic
  - $C \sqsubseteq D$ does not imply $\mathbf{T}(C) \sqsubseteq \mathbf{T}(D)$

The logic $\mathbf{ALC} + \mathbf{T}_{min}$

Example

$\mathbf{T}(TopPlayer) \sqsubseteq FairPlayer$

$\mathbf{T}(TopPlayer \sqcap InterPlayer) \sqsubseteq \neg FairPlayer$

Reasoning

- ABox:
  - $TopPlayer(paolino)$
  - $TopPlayer(cristiano)$
- Expected conclusions:
  - $FairPlayer(paolino)$
  - $FairPlayer(cristiano)$

The logic $\mathbf{ALC} + \mathbf{T}_{min}$

Example

$$\mathbf{T}(\mathit{TopPlayer}) \sqsubseteq \mathit{FairPlayer}$$
$$\mathbf{T}(\mathit{TopPlayer} \sqcap \mathit{InterPlayer}) \sqsubseteq \neg \mathit{FairPlayer}$$

Reasoning

- ABox:
  - $\mathit{TopPlayer}(\mathit{paolino})$
  - $\mathit{TopPlayer}(\mathit{cristiano})$
- Expected conclusions:
  - $\mathit{FairPlayer}(\mathit{paolino})$
  - $\mathit{FairPlayer}(\mathit{cristiano})$

The logic $\mathbf{ALC} + \mathbf{T}_{min}$

Example

$\mathbf{T}(TopPlayer) \sqsubseteq FairPlayer$

$\mathbf{T}(TopPlayer \sqcap InterPlayer) \sqsubseteq \neg FairPlayer$

Reasoning

- ABox:
  - $TopPlayer(paolino)$
  - $TopPlayer(cristiano)$
- Expected conclusions:
  - $FairPlayer(paolino)$
  - $FairPlayer(cristiano)$

The logic $\mathbf{ALC} + \mathbf{T}_{min}$

Example

$\mathbf{T}(TopPlayer) \sqsubseteq FairPlayer$

$\mathbf{T}(TopPlayer \sqcap InterPlayer) \sqsubseteq \neg FairPlayer$

Reasoning

- ABox:
  - $TopPlayer(nicolo), InterPlayer(nicolo)$
  - $TopPlayer(lautaro), InterPlayer(lautaro)$
- Expected conclusions:
  - $\neg FairPlayer(nicolo)$
  - $\neg FairPlayer(lautaro)$

The logic $\mathbf{ALC} + \mathbf{T}_{min}$

Example

$$\mathbf{T}(TopPlayer) \sqsubseteq FairPlayer$$
$$\mathbf{T}(TopPlayer \sqcap InterPlayer) \sqsubseteq \neg FairPlayer$$

Reasoning

- ABox:
  - $TopPlayer(nicolo), InterPlayer(nicolo)$
  - $TopPlayer(lautaro), InterPlayer(lautaro)$
- Expected conclusions:
  - $\neg FairPlayer(nicolo)$
  - $\neg FairPlayer(lautaro)$

The logic $\mathbf{ALC} + \mathbf{T}_{min}$

Example

$\mathbf{T}(TopPlayer) \sqsubseteq FairPlayer$

$\mathbf{T}(TopPlayer \sqcap InterPlayer) \sqsubseteq \neg FairPlayer$

Reasoning

- ABox:
  - $TopPlayer(nicolo), InterPlayer(nicolo)$
  - $TopPlayer(lautaro), InterPlayer(lautaro)$
- Expected conclusions:
  - $\neg FairPlayer(nicolo)$
  - $\neg FairPlayer(lautaro)$

The logic $\mathcal{ALC} + \mathbf{T}$

Semantics

- $\mathcal{M} = \langle \Delta^{\mathcal{I}}, <, .^{\mathcal{I}} \rangle$
  - additional ingredient: preference relation among domain elements
  - $<$ is an irreflexive, transitive, modular and well-founded relation over $\Delta^{\mathcal{I}}$:
    - for all $S \subseteq \Delta^{\mathcal{I}}$, for all $x \in S$, either $x \in Min_<(S)$ or $\exists y \in Min_<(S)$ such that $y < x$
    - $Min_<(S) = \{u : u \in S \text{ and } \nexists z \in S \text{ s.t. } z < u\}$
  - Semantics of the $\mathbf{T}$ operator: $(\mathbf{T}(C))^{\mathcal{I}} = Min_<(C^{\mathcal{I}})$

Weakness of monotonic semantics

Logic $\mathcal{ALC} + \mathbf{T}$

- The operator **T** is nonmonotonic, but...
- The logic is monotonic
  - If KB $\models F$, then KB' $\models F$ for all KB' $\supseteq$ KB

Example

- in the KB of the previous slides:
  - if *TopPlayer(leao)* $\in$ ABox, we are not able to

Weakness of monotonic semantics

Logic $\mathcal{ALC} + \mathbf{T}$

- The operator **T** is nonmonotonic, but...
- The logic is monotonic
  - If KB $\models F$, then KB' $\models F$ for all KB' $\supseteq$ KB

Example

- in the KB of the previous slides:
  - if $TopPlayer(leao) \in$ ABox, we are not able to:
    - assume that $\mathbf{T}(TopPlayer)(leao)$
    - infer that $FairPlayer(leao)$

Weakness of monotonic semantics

Logic $\mathcal{ALC} + \mathbf{T}$

- The operator **T** is nonmonotonic, but...
- The logic is monotonic
  - If KB $\models F$, then KB' $\models F$ for all KB' $\supseteq$ KB

Example

- in the KB of the previous slides:
  - if $TopPlayer(leao) \in$ ABox, we are not able to:
    - assume that $\mathbf{T}(TopPlayer)(leao)$
    - infer that $FairPlayer(leao)$

The nonmonotonic logic ALC + $\mathbf{T}_{min}$

Rational closure

- Preference relation among models of a KB
  - $\mathcal{M}_1 < \mathcal{M}_2$ if $\mathcal{M}_1$ contains less exceptional (not minimal) elements
  - $\mathcal{M}$ minimal model of KB if there is no $\mathcal{M}'$ model of KB such that $\mathcal{M}' < \mathcal{M}$
- Minimal entailment
  - KB $\models_{min} F$ if F holds in all *minimal* models of KB
- Nonmonotonic logic
  - KB $\models_{min} F$ does not imply KB' $\models_{min} F$ with KB' $\supset$ KB
- Corresponds to a notion of rational closure of KB

The nonmonotonic logic $\mathbf{ALC} + \mathbf{T}_{min}$

Rational closure

- Preference relation among models of a KB
  - $\mathcal{M}_1 < \mathcal{M}_2$ if $\mathcal{M}_1$ contains less exceptional (not minimal) elements
  - $\mathcal{M}$ minimal model of KB if there is no $\mathcal{M}'$ model of KB such that $\mathcal{M}' < \mathcal{M}$
- Minimal entailment
  - KB $\models_{min} F$ if F holds in all *minimal* models of KB
- Nonmonotonic logic
  - KB $\models_{min} F$ does not imply KB' $\models_{min} F$ with KB' $\supset$ KB
- Corresponds to a notion of rational closure of KB

The nonmonotonic logic $\mathbf{ALC} + \mathbf{T}_{min}$

Rational closure

- Preference relation among models of a KB
  - $\mathcal{M}_1 < \mathcal{M}_2$ if $\mathcal{M}_1$ contains less exceptional (not minimal) elements
  - $\mathcal{M}$ minimal model of KB if there is no $\mathcal{M}'$ model of KB such that $\mathcal{M}' < \mathcal{M}$
- Minimal entailment
  - KB $\models_{min} F$ if F holds in all *minimal* models of KB
- Nonmonotonic logic
  - KB $\models_{min} F$ does not imply KB' $\models_{min} F$ with KB' $\supset$ KB
- Corresponds to a notion of rational closure of KB

The Description Logic with Typicality and Probabilities

The logic $\mathbf{T}^{\text{CL}}$

- extension of $\mathcal{ALC}$ by inclusions

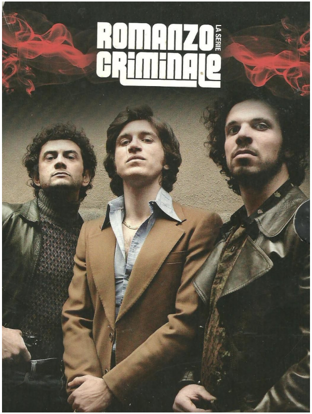
$$p \ :: \ \mathbf{T}(C) \sqsubseteq D$$

- $p \in (0.5, 1) \subseteq \mathbb{R}$ is probability of the typicality inclusion
  - epistemic interpretation: we believe p in the fact that typical Cs are Ds
  - probabilistic interpretation: all typical properties, but with a different percentage of exceptions

The Description Logic with Typicality and Probabilities

Example

- $Criminal \sqsubseteq \exists hasCommited.Crime$
- $Criminal \sqsubseteq Convicted$
- $0.8 \ :: \ \mathbf{T}(Criminal) \sqsubseteq \exists uses.Weapon$
- $0.9 \ :: \ \mathbf{T}(Criminal) \sqsubseteq \exists uses.Gun$
- $0.75 \ :: \ \mathbf{T}(Criminal) \sqsubseteq WellDressed$
- $0.7 \ :: \ \mathbf{T}(Criminal) \sqsubseteq Murderer$
- $0.8 \ :: \ \mathbf{T}(Criminal) \sqsubseteq \neg NicePerson$
- $0.85 \ :: \ \mathbf{T}(Criminal) \sqsubseteq Rich$
- $0.95 \ :: \ \mathbf{T}(Criminal) \sqsubseteq \neg BaglionisFan$



Concept Combination

Basic ideas

Concept Combination

- Inventing novel concepts by combining the typical knowledge of pre-existing ones
- Important human creative ability
- prototypical concepts are not compositional
  - Example: Pet Fish
    - typical pet: furry and warm
    - typical fish: grayish
    - typical pet fish: neither furry and warm nor grayish (typically, it is red)

The Logic of Concept Combination

Concept Combination

- method inspired by cognitive semantics for the identification of a dominance effect between the concepts to be combined
  - HEAD: stronger element of the combination
  - MODIFIER
- definition of a revised knowledge base, enriched by typical properties of the combined concept
- Description logic $\mathbf{T}^{\mathsf{CL}}$: semantics inspired by DISPONTE for considering only *some* scenarios

The Logic of Concept Combination

The logic $\mathbf{T}^{\text{CL}}$

- extension of $\mathcal{ALC}$ by inclusions

$$p \ :: \ \mathbf{T}(C) \sqsubseteq D$$

- $p \in (0.5, 1) \subseteq \mathbb{R}$ is probability of the typicality inclusion
- typical properties $\mathbf{T}(C_H \sqcap C_M)$ from scenarios obtained by considering only *some* typicality properties
- combined concept: properties holding in scenarios:
  - consistent with respect to KB;
  - not trivial, e.g. those ascribing *all* properties of the HEAD are discarded;
  - giving preference to C_H w.r.t. C_M with the highest probability

Example

Pet Fish

- $Fish \sqsubseteq \forall livesIn.Water$
- $0.9 :: \mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Affectionate$
- $0.7 :: \mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Warm$
- $0.6 :: \mathbf{T}(Fish) \sqsubseteq Greyish$
- $0.9 :: \mathbf{T}(Fish) \sqsubseteq Scaly$
- $0.8 :: \mathbf{T}(Fish) \sqsubseteq \neg Warm$

Example

Pet Fish - Different scenarios

- $Fish \sqsubseteq \forall livesIn.Water$
- $0.9 :: \mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Affectionate$
- $0.7 :: \mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Warm$
- $0.6 :: \mathbf{T}(Fish) \sqsubseteq Greyish$
- $0.9 :: \mathbf{T}(Fish) \sqsubseteq Scaly$
- $0.8 :: \mathbf{T}(Fish) \sqsubseteq \neg Warm$

Example

Pet Fish - Different scenarios

- $Fish \sqsubseteq \forall livesIn.Water$
- $0.9 :: \mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Affectionate$
- $0.7 :: \mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Warm$
- $0.6 :: \mathbf{T}(Fish) \sqsubseteq Greyish$
- $0.9 :: \mathbf{T}(Fish) \sqsubseteq Scaly$
- $0.8 :: \mathbf{T}(Fish) \sqsubseteq \neg Warm$

Example

Pet Fish - Different scenarios

- $Fish \sqsubseteq \forall livesIn.Water$
- $0.9 :: \mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Affectionate$
- $0.7 :: \mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Warm$
- $0.6 :: \mathbf{T}(Fish) \sqsubseteq Greyish$
- $0.9 :: \mathbf{T}(Fish) \sqsubseteq Scaly$
- $0.8 :: \mathbf{T}(Fish) \sqsubseteq \neg Warm$

Example

Pet Fish - Different scenarios

- $Fish \sqsubseteq \forall livesIn.Water$
- $0.9 :: \mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Affectionate$
- $0.7 :: \mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Warm$
- $0.6 :: \mathbf{T}(Fish) \sqsubseteq Greyish$
- $0.9 :: \mathbf{T}(Fish) \sqsubseteq Scaly$
- $0.8 :: \mathbf{T}(Fish) \sqsubseteq \neg Warm$

Example

Pet Fish - Inconsistent scenario

- $Fish \sqsubseteq \forall livesIn.Water$
- $0.9 :: \mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Affectionate$
- $0.7 :: \mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Warm$
- $0.6 :: \mathbf{T}(Fish) \sqsubseteq Greyish$
- $0.9 :: \mathbf{T}(Fish) \sqsubseteq Scaly$
- $0.8 :: \mathbf{T}(Fish) \sqsubseteq \neg Warm$
- Probability: -

Example

Pet Fish - Trivial scenario

- $Fish \sqsubseteq \forall livesIn.Water$
- $0.9 :: \mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Affectionate$
- $0.7 :: \mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Warm$
- $0.6 :: \mathbf{T}(Fish) \sqsubseteq Greyish$
- $0.9 :: \mathbf{T}(Fish) \sqsubseteq Scaly$
- $0.8 :: \mathbf{T}(Fish) \sqsubseteq \neg Warm$
- Probability:
  $(1-0.9) \times (1-0.8) \times 0.7 \times \cdots \times 0.8 = 0.1\%$

Example

Pet Fish - MODIFIER preferred to the HEAD

- $Fish \sqsubseteq \forall livesIn.Water$
- $0.9 :: \mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Affectionate$
- $0.7 :: \mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- $0.8 :: \mathbf{T}(Pet) \sqsubseteq Warm$
- $0.6 :: \mathbf{T}(Fish) \sqsubseteq Greyish$
- $0.9 :: \mathbf{T}(Fish) \sqsubseteq Scaly$
- $0.8 :: \mathbf{T}(Fish) \sqsubseteq \neg Warm$
- Probability: 0.05%

Example

Pet Fish - Selected scenario

- $Fish \sqsubseteq \forall livesIn.Water$
- 0.9 :: $\mathbf{T}(Pet) \sqsubseteq \forall livesIn.(\neg Water)$
- 0.8 :: $\mathbf{T}(Pet) \sqsubseteq Affectionate$
- 0.7 :: $\mathbf{T}(Fish) \sqsubseteq \neg Affectionate$
- 0.8 :: $\mathbf{T}(Pet) \sqsubseteq Warm$
- 0.6 :: $\mathbf{T}(Fish) \sqsubseteq Greyish$
- 0.9 :: $\mathbf{T}(Fish) \sqsubseteq Scaly$
- 0.8 :: $\mathbf{T}(Fish) \sqsubseteq \neg Warm$
- Probability: 0.092%

Formal definitions

Atomic choice

- Given KB$= \langle \mathcal{R}, \mathcal{T}, \mathcal{A} \rangle$
- $\mathcal{T} = \{E_1 = q_1 \ :: \mathbf{T}(C_1) \sqsubseteq D_1, \ldots, E_n = q_n \ :: \mathbf{T}(C_n) \sqsubseteq D_n\}$
- $(E_i,\ k_i)$ is an *atomic choice*, where $k_i \in \{0, 1\}$

Selection

- set of atomic choices ν
- ν is a *selection* if, for each E_i, one decision is taken
  - either $(E_i,\ 0) \in \nu$ and $(E_i,\ 1) \not\in \nu$ or
  - $(E_i,\ 1) \in \nu$ and $(E_i,\ 0) \not\in \nu$ for $i = 1, 2, \ldots, n$
- probability of ν

$$P(\nu) = \prod_{(E_i,1)\in\nu} q_i \prod_{(E_i,0)\in\nu} (1 - q_i)$$

Formal definitions

Scenario

- given a selection σ, *scenario* $w_\sigma = \langle \mathcal{R}, \{E_i \mid (E_i, 1) \in \sigma\}, \mathcal{A} \rangle$
- $P(w_\sigma) = P(\sigma)$
- a scenario is *consistent* when it admits a model in $\mathbf{T}^{\mathsf{CL}}$

C-revised knowledge base

- Output of combining C_H and C_M into the compound C

$$\mathcal{K}_C = \langle \mathcal{R}, \mathcal{T} \cup \{p \, : \, \mathbf{T}(C) \sqsubseteq D\}, \mathcal{A} \rangle$$

- for all D such that $\mathbf{T}(C) \sqsubseteq D$ is entailed in w_σ

Complexity results

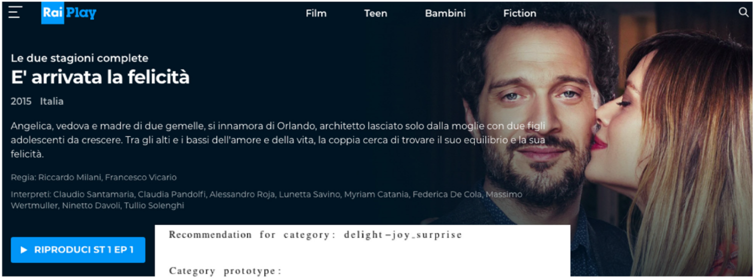
- Entailment restricted to *C-revised* knowledge base
- Reasoning in $\mathbf{T}^{\mathsf{CL}}$ is $\textsc{ExpTime}$-complete.

The logic $\mathbf{T}^{\text{CL}}$

Results

- able to capture some well known and paradigmatic examples of concept combination from the cognitive science literature (e.g. conjunction fallacy problem)
- can be iteratively applied to combine prototypical concepts already resulting from the combination of prototypes
- several applications with generation of novel concepts as the combination of two (or more) prototypes
  - ...

Intelligent Recommender Systems

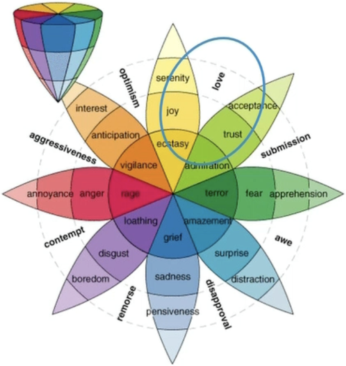



```
Recommendation for category: delight-joy_surprise

Category prototype:

[ ('happiness', 0.98, True),
 ('beatitude', 0.97, True),
 ('party', 0.97, True),
 ('jubilant', 0.97, True),
 ('ecstatic', 0.95, True)],
 ('surprise', 0.93, False),
['not holocaust', 'not anticipation']

Recommended items:

E_ arrivata la felicit  - S1E15 -
        \-> Reason:
        instance's description has the following word(s) in common
```

Creation of stories

DEGARI

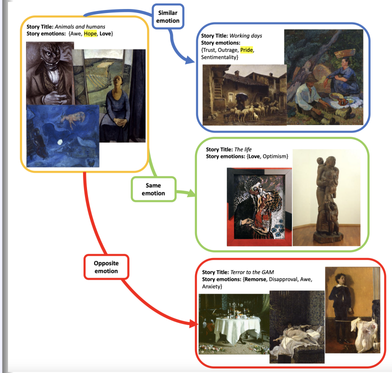
- affective-based sensemaking system for grouping and suggesting stories created by the users about the cultural artefacts in a museum
- classification and suggestion of stories encompassing cultural items able to evoke not only the very same emotions of already experienced or preferred museum objects but also novel items sharing different emotional stances
  - break the filter bubble effect
  - open the users view towards more inclusive and empathy-based interpretations of cultural content
- tested with deaf people on the collection of the Gallery of Modern Art (GAM) in Turin



Creation of stories

DEGARI

- affective-based sensemaking system for grouping and suggesting stories created by the users about the cultural artefacts in a museum
- classification and suggestion of stories encompassing cultural items able to evoke not only the very same emotions of already experienced or preferred museum objects but also novel items sharing different emotional stances
    - break the filter bubble effect
    - open the users view towards more inclusive and empathy-based interpretations of cultural content
- tested with deaf people on the collection of the Gallery of Modern Art (GAM) in Turin



CILC paper #1

Antonio Lieto, Gian Luca Pozzato, Gioele Tallone

A Description Logics Based Cognitively Inspired Tool for Knowledge Generation Via Concept Combination

Dynamic Knowledge Generation

Basic ideas

- dynamic generation of novel knowledge by exploiting $\mathbf{T}^{\text{CL}}$
- Given a goal expressed as a set of properties:
  - if one cannot find a concept able to fulfill all these properties, exploit $\mathbf{T}^{\text{CL}}$ in order to find two concepts whose combination satisfies the goal
- typicality properties obtained from the combination are added to the initial knowledge
- application in cognitive architectures
  - overcome *impasse* in SOAR by extending the possible options of its subgoaling procedures

How does it work?

The problem

- Knowledge base (ontology in $\mathbf{T}^{\text{CL}}$) $\mathcal{K}$
- Set of goals $\mathcal{G} = \{D_1, D_2, \ldots, D_n\}$
- *Solution* for the goal = concept C such that, for all D_i, either $\mathcal{K} \models C \sqsubseteq D_i$ or $\mathcal{K} \models \mathbf{T}(C) \sqsubseteq D_i$ in $\mathcal{ALC} + \mathbf{T}_{min}$
- If there is no solution, try to generate a *new concept* by combining two existing ones C_1 and C_2 by means of $\mathbf{T}^{\text{CL}}$
  - $(C_1 \sqcap C_2)$ is a solution if it is a solution with respect to the revised knowledge base $\mathcal{K}_C$, i.e. either $\mathcal{K}_C \models C \sqsubseteq D_i$ or $\mathcal{K}_C \models \mathbf{T}(C) \sqsubseteq D_i$ in $\mathcal{ALC} + \mathbf{T}_{min}$

Objective

Problem

- Generation of novel knowledge obtained through a process of commonsense reasoning
- Given an intelligent agent and a set of *goals*, if it is not able to achieve them from an initial knowledge base, then it tries to dynamically generate new knowledge by *combining* available information
- Novel information will be then used to extend the initial knowledge base

Example

Objective

Example

- normally, coffee contains caffeine and is a hot beverage
- the chocolate with cream is normally sweet and has a taste of milk
- Limoncello is not a hot beverage
- Both coffee and Limoncello are after meal drinks
- June in Turin suggests to have
  - a hot after-meal drink
  - sweet
  - having taste of milk
- None of the concepts are able to achieve the goal on their own
- however, the combination between coffee and chocolate with cream provides a solution
  - famous Turin drink known as *Bicerín* (coffee, chocolate and cream)

How does it work?

$\mathcal{G} = \{AfterMealDrink, HotBeverage, Sweet, TasteOfMilk\}$

$0.9 :: \mathbf{T}(Coffee) \sqsubseteq AfterMealDrink$
$0.8 :: \mathbf{T}(Coffee) \sqsubseteq WithCaffeine$
$0.85 :: \mathbf{T}(Coffee) \sqsubseteq HotBeverage$
$Limoncello \sqsubseteq AfterMealDrink$
$0.9 :: \mathbf{T}(Limoncello) \sqsubseteq \neg HotBeverage$
$0.65 :: \mathbf{T}(ChocolateWithCream) \sqsubseteq Sweet$
$0.95 :: \mathbf{T}(ChocolateWithCream) \sqsubseteq TasteOfMilk$

Solution: combination *Coffee* and *ChocolateWithCream*

$0.9 :: \mathbf{T}(Coffee \sqcap ChocolateWithCream) \sqsubseteq AfterMealDrink$
$0.85 :: \mathbf{T}(Coffee \sqcap ChocolateWithCream) \sqsubseteq HotBeverage$
$0.65 :: \mathbf{T}(Coffee \sqcap ChocolateWithCream) \sqsubseteq Sweet$
$0.95 :: \mathbf{T}(Coffee \sqcap ChocolateWithCream) \sqsubseteq TasteOfMilk$

Contribution

What's new?

- EDIFICA (ExtenDIble & FlexIble concept Combination Architecture) tries to tackle the main criticisms of existing tool GOCCIOLA:
  - EDIFICA: goal directed - GOCCIOLA: randomly selects concepts to be combined
  - number of concepts to be combined: GOCCIOLA: 2, EDIFICA: ≥ 2 (no limitations)
  - generation of scenarios: GOCCIOLA: brute force, EDIFICA: smart (for instance by discarding inconsistent ones)
  - EDIFICA implements a more sophisticated mechanism for choosing the list of concepts to be combined among all the candidates

EDIFICA

Further details

- implemented in Pyhton
- exploits the translation of an ALC $+ \mathbf{T}_{min}$ knowledge base into standard $\mathcal{ALC}$
- exploits the system CoCoS for generating scenarios and choosing the selected one(s) according to the logic $\mathbf{T}^{\text{CL}}$
- exploits WordNet sysnsets in order to extend its search space in case of a failure

CILC paper #2

Valentina Gliozzi, Gian Luca Pozzato, Alberto Valese

Learning Typicality Inclusions in a Probabilistic Description Logic for Concept Combination

Objectives

- Extract logical rules using $\mathbf{T}^{\text{CL}}$ from diverse datasets (tabular data)
- Starting point: CN2 algorithm
  - typically employed for classification tasks
  - adapted to our case for learning both structure and probabilities
- well-known datasets: iris, zoo, GTZAN
  - efficacy in generating typicality inclusions across different data domains

The CN2 algorithm

Introduction

- induction algorithm used to automatically generate rules for classifying new data
- developed by Clark and Niblett in 1989 to handle noisy data (with uncertainties, errors, inconsistencies)
- combines efficiency and noise-handling capabilities of the ID3 algorithm with the flexible search strategy of the AQ family (if-then rule form)
- non-overcomplicated nature
  - adaptability through modifications tailored to our needs
  - simplicity

Our Algorithm

Basic ideas

- CN2 stops at the first rule sufficient for classification
- E.g. zoo dataset:
  - IF *Milk* $\rightarrow$ type=*Mammal*
  - 17 different attributes
  - continuous values converted into discrete intervals
  - we can obtain $Mammal \sqsubseteq Milk$
  - we also want to obtain $\mathbf{T}(Mammal) \sqsubseteq FourLegs$
- we select a target class
- re-run CN2 separately for each target class

Species	Hair	Legs	...	Family
Dog	Yes	4	...	Mammal
Goldfish	No	0	...	FISH

Our Algorithm

Modified CN2

- we search for single-attribute rules for a given and fixed class
- we use AUC-ROC as the evaluation function to balance informativeness and coverage
- the best rule is further evaluated by frequency
  - standard inclusion if all examples are covered
  - typicality inclusion (probability = percentage of coverage) otherwise
- we do not remove examples covered by the rules to ensure rules are probable across the entire set

Our Algorithm

Example

- Zoo example, prototype of a Mammal
- CN2 only finds IF *Milk* $\rightarrow$ type=*Mammal*
- With our modified algorithm:
  - re-run CN2 by evaluating the other attributes (eggs==true, eggs==false, hair==true, hair==false, ...)

1	$Mammal \sqsubseteq Milk$	-
2	$\mathbf{T}(Mammal) \sqsubseteq \neg Eggs$	0.976
3	$\mathbf{T}(Mammal) \sqsubseteq Hair$	0.951
4	$\mathbf{T}(Mammal) \sqsubseteq FourLegs$	0.756
5	$\mathbf{T}(Mammal) \sqsubseteq Toothed$	0.976
6	$\mathbf{T}(Mammal) \sqsubseteq Catsize$	0.78
7	$\mathbf{T}(Mammal) \sqsubseteq \neg Aquatic$	0.854
8	$Mammal \sqsubseteq Breathes$	-
9	$Mammal \sqsubseteq \neg Feathers$	-
10	$\mathbf{T}(Mammal) \sqsubseteq \neg Airborne$	0.951

Contributions

Dynamic Knowledge Generation

- EDIFICA: tool exploiting $\mathbf{T}^{\textsf{CL}}$ for
  - solving a goal with concept combination
  - dynamic generation of knowledge

Learning Ontologies

- modified CN2 algorithm for automated learning:
  - inclusions (both rigid and typical)
  - probabilities

Future works

Dynamic Knowledge Generation

- partial solutions, satisfying a proper subset of the initial goals
- evaluation of EDIFICA by suitable experiments involving humans

Learning ontologies

- only a first step
- apply the algorithm to more complex datasets
  - domain of music
  - refine the concepts presented in AIxIA 2022, automating rule extraction from more intricate and precise datasets (e.g. GTZAN)

References

- F. Baader and B. Hollunder (1995), Embedding defaults into terminological knowledge representation formalisms. *JAR*, 14(1):149–180.
- L. Giordano, V. Gliozzi, N. Olivetti, and G.L. Pozzato (2013), A NonMonotonic Description Logic for Reasoning About Typicality. *Artificial Intelligence*, 195:165 – 202.
- L. Giordano, V. Gliozzi, N. Olivetti, and G.L. Pozzato (2015), Semantic characterization of Rational Closure: from Propositional Logic to Description Logics. *Artificial Intelligence*, 226:1–33.
- A. Lieto, G.L. Pozzato, A. Valese, M. Zito (2022). A logic-based tool for dynamic generation and classification of musical content. In *Proc. of the International Conference of the Italian Association for Artificial Intelligence*, pp. 313326. Springer.
- F. Riguzzi, E. Bellodi, E. Lamma, and R. Zese (2015), Probabilistic description logics under the distribution semantics. *Semantic Web*, 6(5): 477–501.

Any question?