

C7

7.1

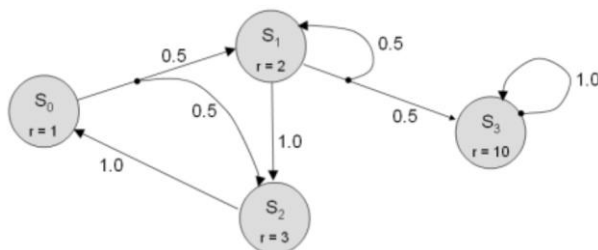
Care din următoarele afirmații sunt adevărate?

5 pc

- ☐ În cadrul învățării pasive, agentul își actualizează politica pe măsura ce învață.
- ☐ Un proces de decizie Markov poate fi rezolvat ca un sistem de ecuații liniare. $U(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \cdot U(s')$
- ☒ O valoare mai mare a parametrului ϵ din cadrul metodei ϵ -greedy indică o explorare mai mare a spațiului.
- ☒ O valoare de 0.2 a factorului de discount favorizează recompensele imediate.

7.2

Considerăm următorul proces de decizie Markov. Aplicați algoritmul Value Iteration. Factorul de discount este 1. Există o singură acțiune disponibilă pentru fiecare stare, cu excepția stării S1 care are două acțiuni. Valorile inițiale ale utilităților sunt 0. Care din afirmațiile de mai jos sunt adevărate?



- ☐ Utilitatea stării S1 după prima iterație este 2
- ☐ Utilitatea stării S1 după a doua iterație este 8
- ☐ Utilitatea stării S0 după prima iterație este 0
- ☐ Utilitatea stării S2 după prima iterație este 3

Pasi Value Iteration

Inițializează utilitățile cu valori arbitrare

Actualizează utilitatea fiecărei stări din utilitățile vecinilor

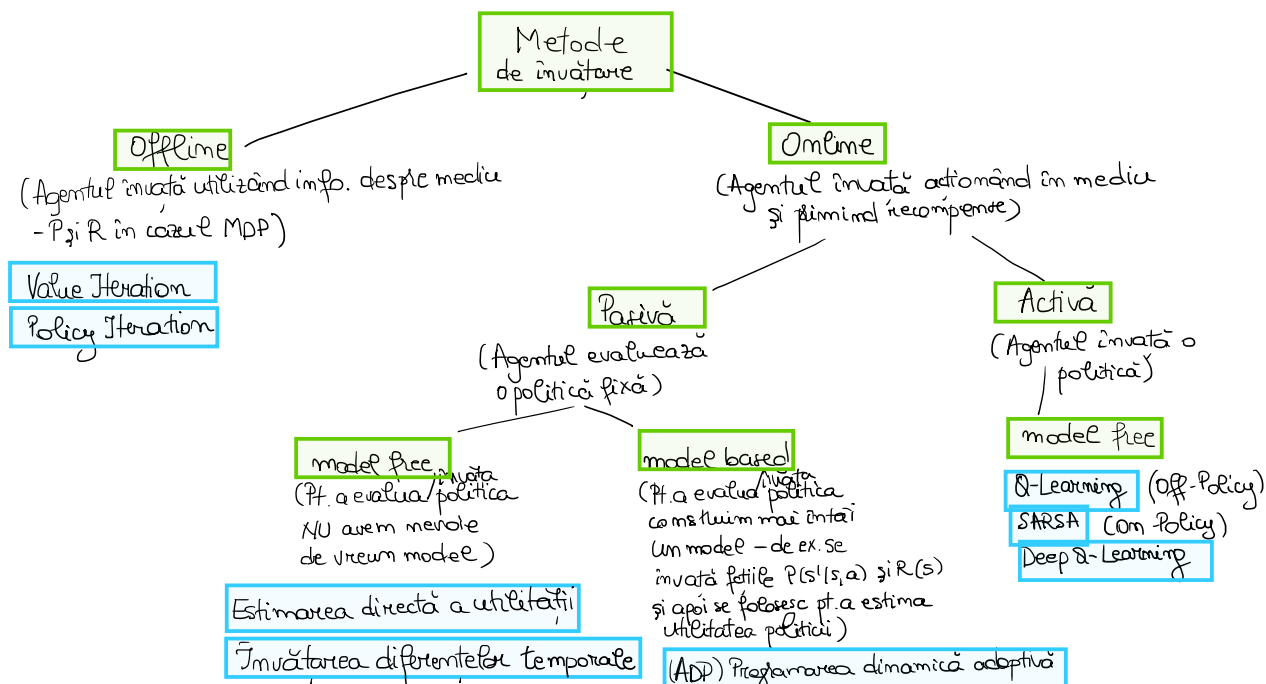
$$U_{i+1}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) U_i(s')$$

Repetă pentru fiecare s simultan, până la atingerea unui echilibru

$$\begin{aligned} \gamma &= 1 & U_0(S_0) &= U_0(S_1) = U_0(S_2) = U_0(S_3) = 0 \\ U_1(S_0) &= 1 + 1 \cdot \max(0.5 \cdot U_0(S_1) + 0.5 \cdot U_0(S_2)) \Rightarrow U_1(S_0) = 1 \\ U_1(S_1) &= 2 + 1 \cdot \max(0.5 \cdot U_0(S_1) + 0.5 \cdot U_0(S_3), 1 \cdot U_0(S_2)) \Rightarrow U_1(S_1) = 2 \\ U_1(S_2) &= 3 + 1 \cdot \max(1.0 \cdot U_0(S_0)) \Rightarrow U_1(S_2) = 3 \\ U_1(S_3) &= 10 + 1 \cdot \max(1.0 \cdot U_0(S_3)) \Rightarrow U_1(S_3) = 10 \\ U_2(S_0) &= 1 + 1 \cdot \max(0.5 \cdot U_1(S_1) + 0.5 \cdot U_1(S_2)) \Rightarrow U_2(S_0) = 3.5 \\ U_2(S_1) &= 2 + 1 \cdot \max(0.5 \cdot U_1(S_1) + 0.5 \cdot U_1(S_3), 1 \cdot U_1(S_2)) \Rightarrow U_2(S_1) = 8 \\ U_2(S_2) &= 3 + 1 \cdot \max(1.0 \cdot U_1(S_0)) \Rightarrow U_2(S_2) = 4 \\ U_2(S_3) &= 10 + 1 \cdot \max(1.0 \cdot U_1(S_3)) \Rightarrow U_2(S_3) = 20 \\ &\dots & & \end{aligned}$$

C8

8.1



8.2

Considerăm N orașe și un vânzător care se află în orașul 1. Vânzătorul poate călători într-un oraș vecin (acțiunea East sau West) sau poate rămâne în orașul curent (acțiunea Stay). Cu probabilitatea p_i vânzătorul ajunge în orașul următor dacă se află în orașul i , iar cu probabilitatea $1-p_i$ e lovit de o furtună și nu ajunge nicaieri. Dacă vânzătorul alege acțiunea Stay, obține recompensa r_i , altfel obține recompensa 0. Considerăm următoarea secvență de stări, acțiuni, recompense (s, a, r, s') : $(s=1, a=Stay, r=6, s'=1)$, $(s=1, a=East, r=0, s'=2)$, $(s=2, a=Stay, r=4, s'=2)$, $(s=2, a=West, r=0, s'=1)$. Rata de învățare este 0.5, discountul este 1, iar valorile inițiale $Q(s, a)=0$. Actualizați valorile $Q(s, a)$ utilizând algoritmul Q-learning. Care din afirmațiile de mai jos sunt adevărate? Calculul (C2) îl atașați formularului.

- ☒ Valoarea $Q(1, East)$ este egală cu 0
- ☒ Valoarea $Q(2, Stay)$ este egală cu 2
- ☐ Valoarea $Q(2, West)$ este egală cu 2
- ☒ Valoarea $Q(2, West)$ este egală cu 1.5
- ☐ Valoarea $Q(1, Stay)$ este egală cu 0

$$Q(s, a) = Q(s, a) + \alpha (r + \gamma \max_{a'} Q(s', a') - Q(s, a))$$

$$\alpha = 0.5 \quad \gamma = 1$$

- $s=1, a=Stay, r=6, s'=1$
 $Q(1, Stay) = 0 + 0.5 (6 + 1 \cdot \max_{a'} Q(1, a') - 0) \Rightarrow Q(1, Stay) = 3$
- $s=1, a=East, r=0, s'=2$
 $Q(1, East) = 0 + 0.5 (0 + 1 \cdot \max_{a'} Q(2, a') - 0) \Rightarrow Q(1, East) = 0$
- $s=2, a=Stay, r=4, s'=2$
 $Q(2, Stay) = 0 + 0.5 (4 + 1 \cdot \max_{a'} Q(2, a') - 0) \Rightarrow Q(2, Stay) = 2$

- $s=2, a=West, r=0, s'=1$
 $Q(2, West) = 0 + 0.5 (0 + 1 \cdot \max_{a'} Q(1, a') - 0)$
 $= 0.5 \cdot Q(1, Stay)$
 $= 1.5$

8.3

Considerăm următoarea configurație. Agentul se află în starea S. Există două stări finale: (2,3) cu recompensa +5 și (1,3) cu recompensa -5. Recompensele în stările neterminale sunt egale cu 0. Considerăm următoarele două încercări: (1,1) - (1,2) - (1,3) și (1,1) - (1,2) - (2,2) - (2,3). Factorul de discount este egal cu 1 iar rata de învățare este 0.5. Presupunem valorile inițiale ale utilităților egale cu 0. Aplicați algoritmul Temporal Differences learning.

		+5
S		-5

Check all that apply.

- ☐ După prima încercare, valorile utilităților nu se modifică
- ☒ După prima încercare, utilitatea stării (1,2) este -2.5
- ☒ După a doua încercare, utilitatea stării (1,2) este -1.25
- ☐ După a doua încercare, valorile utilităților stărilor (1,1) și (1,2) sunt egale cu -2.5

$$(1,1) \xrightarrow{0} (1,2) \xrightarrow{-5} (1,3) \quad \gamma=1 \quad \alpha=0.5 \quad U(\text{stare})=0$$

P. $\forall (s, s', r)$:

$$U(s) = U(s) + \alpha (r + \gamma \cdot U(s') - U(s))$$

$$\left[\begin{array}{l} \bullet (1,1) \xrightarrow{0} (1,2): \quad U((1,1)) = \underbrace{U((1,1))}_0 + 0.5 (0 + 1 \cdot \underbrace{U((1,2))}_0 - \underbrace{U((1,1))}_0) \Rightarrow \boxed{U((1,1)) = 0} \\ \bullet (1,2) \xrightarrow{-5} (1,3): \quad U((1,2)) = \underbrace{U((1,2))}_0 + 0.5 (-5 + 1 \cdot \underbrace{U((1,3))}_0 - \underbrace{U((1,2))}_0) \Rightarrow \boxed{U((1,2)) = -2.5} \end{array} \right.$$

$$(1,1) \xrightarrow{0} (1,2) \xrightarrow{0} (2,2) \xrightarrow{+5} (2,3)$$

$$\bullet (1,1) \xrightarrow{0} (1,2): \quad U((1,1)) = \underbrace{U((1,1))}_0 + 0.5 (0 + 1 \cdot \underbrace{U((1,2))}_{-2.5} - \underbrace{U((1,1))}_0) \Rightarrow \boxed{U((1,1)) = -1.25}$$

$$\bullet (1,2) \xrightarrow{0} (2,2): \quad U((1,2)) = \underbrace{U((1,2))}_{-2.5} + 0.5 (0 + 1 \cdot \underbrace{U((2,2))}_0 - \underbrace{U((1,2))}_{-2.5}) \Rightarrow \boxed{U((1,2)) = -1.25}$$

$$\bullet (2,2) \xrightarrow{+5} (2,3): \quad U((2,2)) = \underbrace{U((2,2))}_0 + 0.5 (5 + 1 \cdot \underbrace{U((2,3))}_0 - \underbrace{U((2,2))}_0) \Rightarrow \boxed{U((2,2)) = 2.5}$$

C9

9.1

6. (1p) Dați exemplu de relație semantică binară, reflexivă, simetrică și tranzitivă ce ar putea apare într-o ontologie pentru domeniul “Facultatea de Informatică”.

Ex:

“Coleg de grupa” (definita de la clasa Student la ea insasi)

“AreLegaturaCu” (definita de la clasa Materie la ea insasi)

9.2

Care din următoarele afirmații sunt adevărate pentru o ontologie?

- ☐ Este independentă de limbă.
- ☐ Poate fi completată cu concepte sau relații semantice preluate din oricare altă ontologie.
- ☒ Poate conține instanțe ale conceptelor.
- ☒ Poate conține lexicalizări ale conceptelor.

În curs se menționează că o taxonomie este independentă de limbă (nu cuprinde ambiguități, denumirile termenilor sunt de obicei în latină → limba statică). Ontologiile conțin lexicalizări → nu aveam aceeași independență de limbaj ca la o taxonomie.

De obicei nu putem prelua direct concepte și relații din orice ontologie și să le includem în alta (este necesar să facem *ontology mapping* mai întâi). Pe de altă parte, ontologiile sunt suficient de expresive încât să ne permită să definim orice concept/instanță/relație semantică (chiar și cu riscul de apariție al inconsistențelor).

Lexicalizarea unui concept = exprimare în cuvinte a conceptului (sinonim, expresie similară, abreviere, traducerea în alta limbă). Pot fi exprimate în Protégé prin [annotation properties](#).

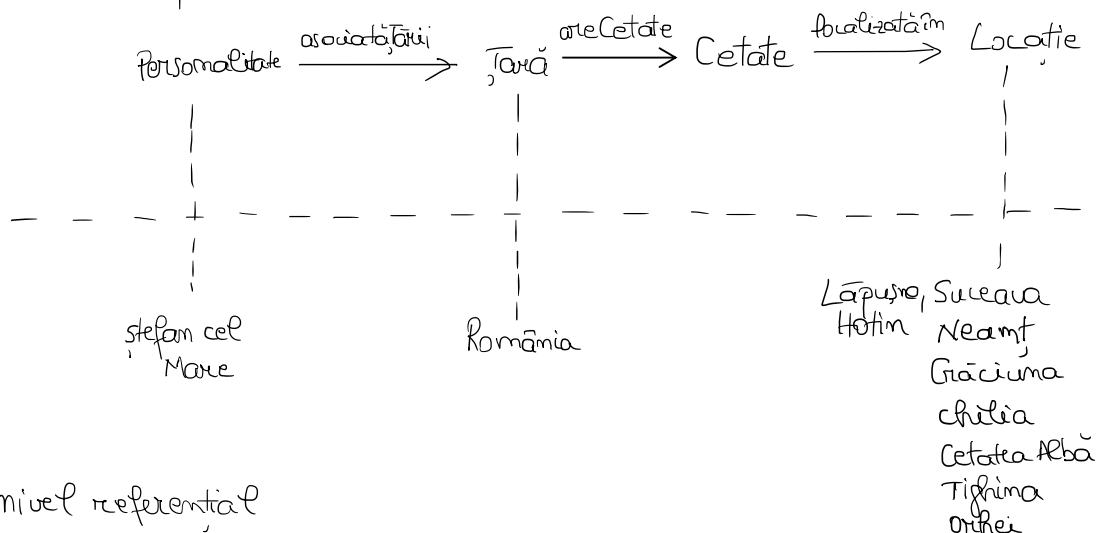
9.3

Care este numărul minim de noduri dintr-o rețea semantică, pe nivelul conceptual și pe cel referențial, în care este reprezentată cunoașterea din textul: "Ștefan cel Mare este considerat o personalitate marcantă a istoriei României. El a creat un sistem de fortificații permanent la granițele țării - în timpul său construindu-se sau dezvoltându-se rețeaua de cetăți ce cuprindea cetățile de la Suceava, Neamț, Crăciuna, Chilia, Cetatea Albă, Tighina, Orhei, Lăpușna și Hotin."

Check all that apply.

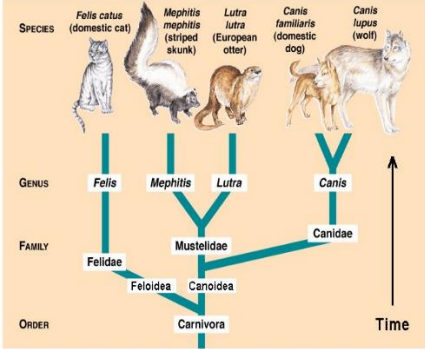
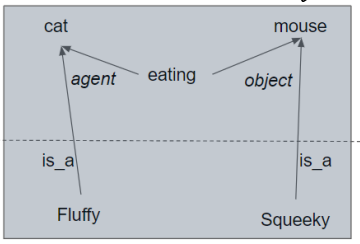
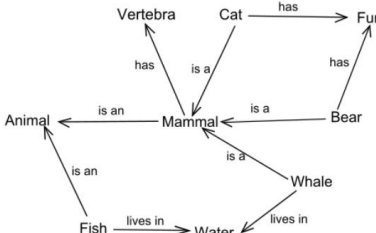
- ☐ 8 noduri pe nivelul conceptual, unul pe cel referențial
- ☒ 4 noduri pe nivelul conceptual, 11 pe cel referențial
- ☐ 8 noduri pe nivelul conceptual, 8 pe cel referențial
- ☐ 12 noduri pe nivelul conceptual, 12 pe cel referențial

nivel conceptual



nivel referențial

Diferente si asemanari intre taxonomii, retele semantice, frame-uri semantice si ontologii

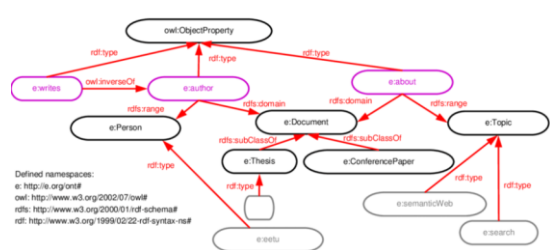
Taxonomie: • independenta de limba • neambigua • expresivitate redusa (doar relatii semantice de tip IS-A, conceptele sunt clasificate intr-un arbore) • neadaptabila la adaugarea de noi informatii (de exemplu, daca am adauga in taxonomia din dreapta o specie noua ar trebui sa remodelam intreg arborele)																									
Retele semantice: • dependente de limba • expresivitate mai ridicata ○ folosim un graf orientat – nu un arbore (si reprezentam graful prin triplete (concept1, relatie, concept2)) ○ putem include orice tip de relatie semantica – nu numai de tip IS A ○ putem include orice tip de concept ○ putem defini un nivel conceptual si unul referential (deci in plus fata de taxonomii, putem defini instante asociate conceptelor)	Conceptual network  Referential network 																								
Frameuri semantice: • dependente de limba • un frame descrie o actiune/o referinta (instanta) • expresivitate ridicata ○ un frame contine unul / mai multe sloturi (fiecare slot continand o relatie si o valoare asociata / o relatie catre un alt frame). • Putem trece de la reprezentarea prin retele semantice la cea prin frameuri si invers – de obicei fara sa pierdem informatii. • La fel ca la retelele semantice, putem realiza inferente folosind semantice frames	<table border="1" data-bbox="901 1333 1177 1648"> <thead> <tr> <th colspan="2">"to drive" frame</th> </tr> </thead> <tbody> <tr> <td>Agent</td> <td>John</td> </tr> <tr> <td>Object</td> <td>a car</td> </tr> <tr> <td>Starting point</td> <td>?</td> </tr> <tr> <td>Ending point</td> <td>?</td> </tr> <tr> <td>Reason</td> <td>?</td> </tr> <tr> <td>Beneficiary</td> <td>?</td> </tr> <tr> <td>Modal</td> <td>?</td> </tr> </tbody> </table> <table border="1" data-bbox="1201 1333 1404 1438"> <thead> <tr> <th colspan="2">"Cat" frame</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>Fluffy</td> </tr> <tr> <td>Age</td> <td>?</td> </tr> <tr> <td>Weight</td> <td>?</td> </tr> </tbody> </table> relative valoare / referinta alt frame	"to drive" frame		Agent	John	Object	a car	Starting point	?	Ending point	?	Reason	?	Beneficiary	?	Modal	?	"Cat" frame		Name	Fluffy	Age	?	Weight	?
"to drive" frame																									
Agent	John																								
Object	a car																								
Starting point	?																								
Ending point	?																								
Reason	?																								
Beneficiary	?																								
Modal	?																								
"Cat" frame																									
Name	Fluffy																								
Age	?																								
Weight	?																								

Ontologii

- La fel ca o retea semantică:
 - cuprinde un set de concepte și relații
 - este dependentă de limbă
 - poate fi văzută ca un hipergraf
 - o putem folosi pentru a realiza inferențe
- Fata de o rețea semantică, sunt mai formalizate și expresive, cuprinzând informații adiționale, precum:
 - C is a set of concepts c , where c is defined as a tuple (i, d, l, r, p, r) , where:
 - i is an unique identifier;
 - d is a list of definitions;
 - l is a list of lexicalisations;
 - r is a list of references to other related sources;
 - p is a list of property identifiers (relations with one member);
 - r is a list of relation identifiers with c as a member.

Handwritten notes: "Equiv. Property (URI)" next to the tuple definition; "annotation properties" with a bracket pointing to p and r ; "object properties" with an arrow pointing to r .
 - R is a set of semantic relations r , each defined as a triple (i, m, b) , where:
 - i is an unique identifier;
 - m is the number of members of that relation;
 - b indicates whether the relation is bidirectional.

Handwritten note: "property network" with an arrow pointing to the triple definition.
 - I is a set of inferences (rules) over the ontology performing changes to it.



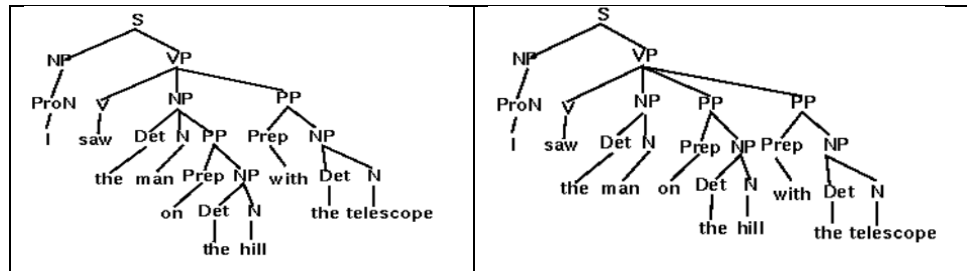
Semantic Web vs normal Web: conectam concepte, nu documente

C10

10.1

Taskuri specifice NLP:

- **Analiza text (text -> intelegere mesaj, extragere info)**
(ex: Analiza sentimentelor, Named Entity Recognition, Clasificarea textelor, Topic Modelling)
(Atentie! mov=diferente fata de curs!)
 - **Analiza lexicala**
 - Tokenizare (We | ' | re | going | there | . |])
 - Identificare cuvinte
 - Maparea cuvintelor cu forma din dictionar
 - Convertirea cuvintelor in forma din dictionar (Lematizare) (vorbisem ->(a) vorbi)
 - **Analiza morfologica** (analiza “formeii” cuvintelor)
 - Descompunerea cuvintelor in morfeme (ex: ne-intelegere)
 - Identificarea partilor de vorbire (POS-tagging)
 - Identificare trasaturi gramaticale (precum gen, persoana, numar, timp, caz)
(eu) am intrebat – verb la perfect compus, persoana a I-a, sg.)
 - Ambiguitate morfologica (ex: acelasi cuvant cu parti diferite de vorbire)
 - **Analiza sintactica** (analiza relatiilor dintre cuvinte la nivel de propozitie)
 - Identificarea partilor de propozitie (subiect, predicat, etc.)
 - Analiza relatiilor sintactice prin construirea de arbori sintactici
 - Ambiguitate sintactica -> mai multi arbori sintactici pentru aceeasi propozitie



- **Analiza semantica**
 - Identificare sensului cuvintelor
 - Ambiguitatea semantica -> mai multe sensuri pentru aceeași propoziție (ex: Ancheta politistului a fost oprită -> Politistul conduce ancheta sau este anchetat?)
- **Analiza la nivel de discurs**
 - Analizăm relațiile dintre unități de discurs
 - Ex: identificarea referințelor anaforice într-un text – task specific și analizei sintactice (ex: Popescu Ion Studentul...)
- **Analiza la nivel pragmatic**
 - “Pragmatica studiază relația dintre un limbaj și utilizarea sa într-un context.”
 - Analiza elementelor pragmatice (contextul poate fi dedus din ton, intonație, cui se adresează mesajul, punctuație).
- **Generare text (info/mesaj -> generare text)**
(ex: Story Generation, Summarization, Question Answering etc.)
- **Aplicații multimodale** (ex: Text-to-speech, speech-to-text, Image-to-text, SignLanguageToSpeech, etc.)

10.2

6. (1p) Cum putem identifica autorul unui text ca fiind una din două persoane dacă avem la dispoziție exemple numeroase a unor texte scrise de cele două persoane? Indicați și tehnologiile necesare specifice Lingvisticii Computaționale.

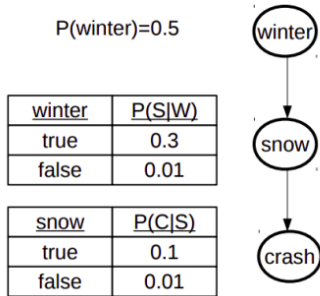
- Un mod de identificare ar putea include o analiză a textelor celor 2 persoane la nivel:
 - Lexical (statistici privind lungimea propozițiilor, frecvența cuvintelor utilizate, a punctuației, a unor simboluri precum emoji)
 - Morfologic (frecvența partilor de vorbire utilizate + analiză unor elemente precum: timpurile verbale utilizate, persoana și numărul la care se vorbește)
 - Sintactic (frecvența partilor de propoziție, analiză anaforelor)
 - Semantic (analiză sentimentelor)
 - De discurs (identificarea centrilor de discurs, identificarea tematicii sau a unor cuvinte cheie care să rezume ideea principală a mesajului)
 - ...

Eventual pe baza acestor trasături extrase s-ar putea antrena unul sau mai mulți clasificatori pentru a determina autorul textului.

C11

11.1

7. (1.5p) Fie rețeaua bayesiană de mai jos. Calculați probabilitatea marginală $P(C)$.



Marginalizare $P(x) = \sum_y P(x, y=y)$

Condiționare $P(x) = \sum_y P(x|y=y) \cdot P(y=y)$

$$P(c) = \sum_{w \in \{T, F\}} \sum_{s \in \{T, F\}} P(c, w=w, s=s) \stackrel{R.B.}{=} \sum_{s \in \{T, F\}} P(c, s=s) = \sum_{s \in \{T, F\}} P(c|s=s) \cdot P(s=s)$$

$$P(s=s) = \sum_w \sum_c P(s=s, w=w, c=c) \stackrel{R.B.}{=} \sum_{w \in \{T, F\}} P(s=s, w=w) = \sum_{w \in \{T, F\}} P(s=s|w=w) \cdot P(w=w)$$

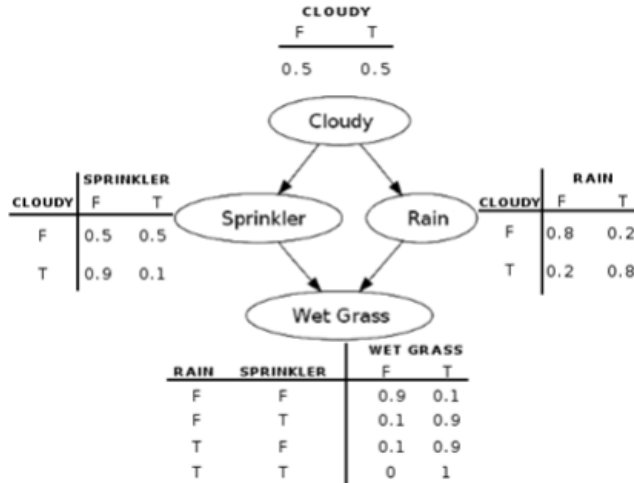
$$P(s=T) = \underbrace{P(s=T|w=T)}_{0.3} \cdot \underbrace{P(w=T)}_{0.5} + \underbrace{P(s=T|w=F)}_{0.01} \cdot \underbrace{P(w=F)}_{0.5} \Rightarrow \boxed{P(s=T) = 0.155}$$

$$P(s=F) = \underbrace{P(s=F|w=T)}_{1-0.3} \cdot \underbrace{P(w=T)}_{0.5} + \underbrace{P(s=F|w=F)}_{1-0.01} \cdot \underbrace{P(w=F)}_{0.5} \Rightarrow \boxed{P(s=F) \approx 0.845}$$

$$P(c) = \underbrace{P(c|s=T)}_{0.1} \cdot \underbrace{P(s=T)}_{0.155} + \underbrace{P(c|s=F)}_{0.01} \cdot \underbrace{P(s=F)}_{0.845} \Rightarrow \boxed{P(c) \approx 0.024}$$

11.2

Fie rețeaua bayesiană din figură. Folosind metoda inferenței prin enumerare, calculați probabilitatea de a ploua (rain = T) dacă cerul este înnorat (cloudy = T) și iarba nu este udă (wet grass = F).



- Alg. de inferență prin enumerare
- $P(x|e) = \frac{P(x,e)}{P(e)} = \alpha \cdot P(x,e) = \alpha \cdot \sum_y P(x,e,y)$
 - $\rightarrow$ variabile neobservate
 - $\rightarrow$ variabile observate
 - Pt. calcule $P(x_i, y)$ folosim:

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^n P(x_i | \text{Parents}(x_i))$$
 - Pt. a det α : $\sum_x P(x|e) = 1$

$$P(R|C, TW) = \alpha \cdot \sum_S P(R, C, TW, S) = \alpha \cdot \sum_S P(C) \cdot P(S|C) \cdot P(R|C) \cdot P(TW|R, S)$$

$$P(R|C, TW) = \alpha \cdot P(C) \cdot P(R|C) \cdot \sum_S P(S|C) \cdot P(TW|R, S)$$

$$P(R|C, TW) = \alpha \cdot \underbrace{P(C)}_{0.5} \cdot \underbrace{P(R|C)}_{0.8} \cdot (\underbrace{P(S|C)}_{0.1} \cdot \underbrace{P(TW|R, S)}_0 + \underbrace{P(TS|C)}_{0.9} \cdot \underbrace{P(TW|R, TS)}_{0.1}) = \alpha \cdot 0.036$$

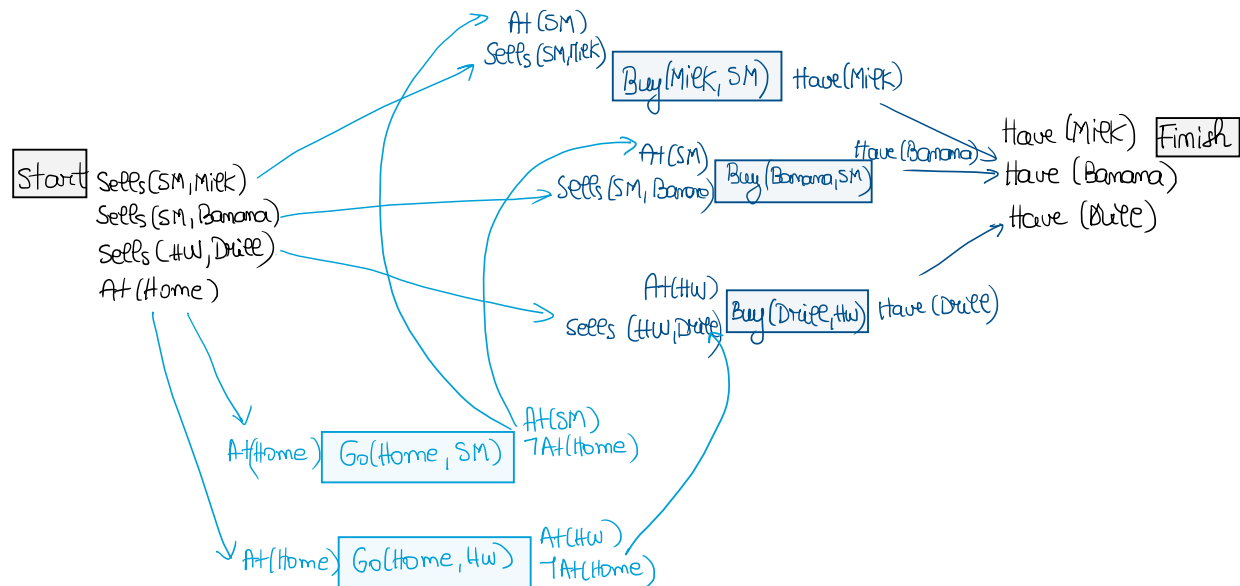
$$P(TR|C, TW) = \alpha \cdot \underbrace{P(C)}_{0.5} \cdot \underbrace{P(TR|C)}_{0.2} \cdot (\underbrace{P(S|C)}_{0.1} \cdot \underbrace{P(TW|TR, S)}_{0.1} + \underbrace{P(TS|C)}_{0.9} \cdot \underbrace{P(TW|TR, TS)}_{0.9}) = \alpha \cdot 0.082$$

$$P(R|C, TW) + P(TR|C, TW) = 1 \Rightarrow \alpha = \frac{1}{0.036 + 0.082} \approx 8.47 \Rightarrow P(R|C, TW) \approx 0.304 \approx 0.31$$

C12

12.1 Aplicați algoritmul de planificare cu ordine parțială cunoscând următoarele:

Actions:
Buy(x, store)
 PRECOND: At(store), Sells(store, x)
 EFFECT: Have(x)
Go(x, y)
 PRECOND: At(x)
 EFFECT: At(y), $\neg$ At(x)
Init: At(Home) and Sells(SM, Milk) and Sells(SM, Banana) and Sells(HW, Drill)
Goal: Have(Milk) and Have(Banana) and Have(Drill)



12.2

7. (2p) Considerăm următoarea problemă de planificare în care acțiunile posibile sunt:

Go(there)
 PRECOND: At(here)
 EFFECT: At(there) $\wedge$ $\neg$ At(here)

și

Buy(x)
 PRECOND: At(store) $\wedge$ Sells(store, x)
 EFFECT: Have(x)

Starea inițială și obiectivul sunt:

Init: At(home) $\wedge$ Sells(TS, tea) $\wedge$ Sells(BS, book)

Goal: At(home) $\wedge$ Have(tea) $\wedge$ Have(book)

Aplicați algoritmul de planificare cu ordine parțială pentru a construi un plan.

O explicație bună se găsește [aici](#) (sl 21-32)