Inteligentni sistemi

- skripta za klasično polaganje ispita, školska 2013/14 (by Stepke) -

NAPOMENA: U nastavku se nalaze odgovori na pitanja za klasično polaganje ispita koji se primenjuju se počev od januarskog roka 2014. godine. Za njihovo rešavanje korišćena je odgovarajuća literatura sa predavanja i vežbi (zimski semester školske 2013/14) koju možete naći na zvaničnom sajtu predmeta.

Pitanja iz oblasti EKSPERTNI SISTEMI (ES)

1. Kako definišemo ES?

Odgovor:

Ekspertni sistem (**ES**) - je *računarski program* kojim se emulira rešavanje problema na način na koji to čini *ekspert* (*čovek*).

2. Navesti i objasniti svrhu osnovna tri modula (dela) ES-a.

3. Koja je svrha baze znanja i šta ona sadrži?

4. Koja je svrha radne memorije i šta ona sadrži?

5. Koja je svrha mehanizma za zaključivanje?

Odgovor na 2,3,4 i 5. pitanje:

Osnovna tri modula (**dela**) **ES**-a su:

1) ***Baza znanja*** - sadrži *domensko znanje* koje MORA biti formalizovano (da bi računar mogao da ga koristi),

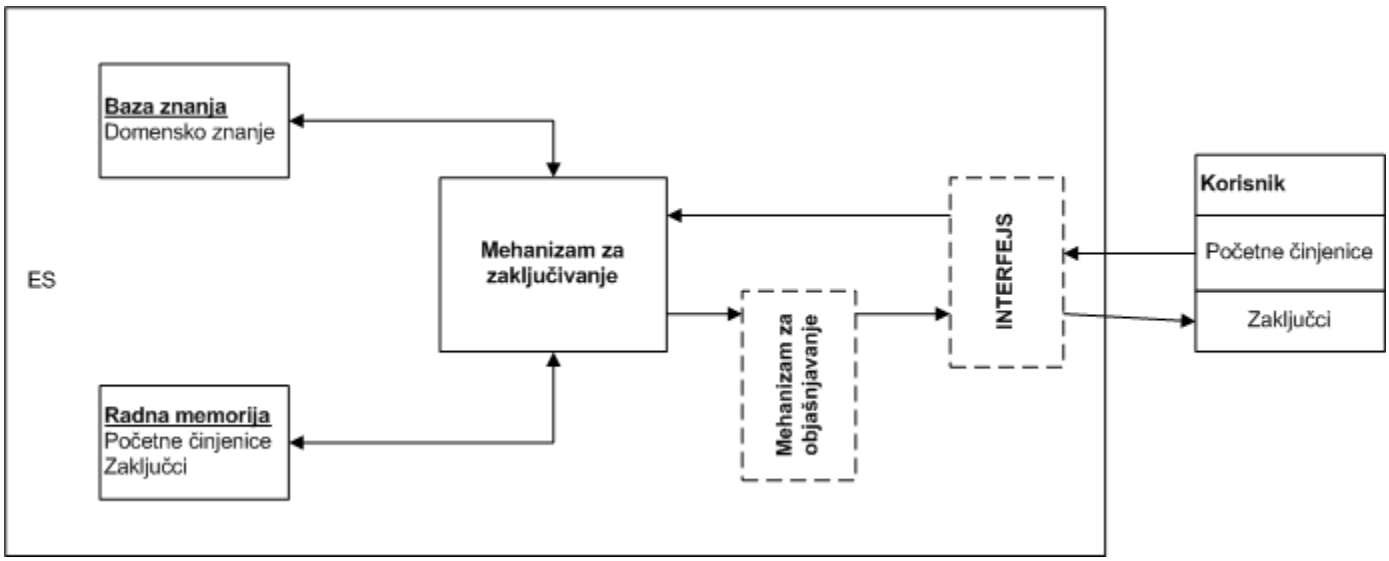
2) ***Radna memorija*** - sadrži *činjenice* i *zaključke* (zaključci predstavljaju činjenice nastale kao posledica rezonovanja).

3) ***Mehanizam za zaključivanje*** - kombinuje znanje iz *baze znanja* i činjenice iz *radne memorije* i stvara nove zaključke (tako omogućava *automatizovano rezonovanje*).

6. Nacrtati arhitekturu ES.

Odgovor:

Arhitektura ekspertnog sistema data je na sledećoj slici:



7. Koji su osnovni uslovi koje neki program mora da zadovolji da bi mogao da se nazove ES?

Odgovor:

Osnovni uslovi koje neki program mora da zadovolji da bi mogao da se nazove **ES** su da:

1. sadrži ***ekspertsko znanje*** iz neke ***oblasti***
2. omogućava ***automatizovano rezonovanje***

8. Koja su osnovna dva dela svakog pravila? Navesti i ukratko objasniti.

Odgovor:

Osnovna dva dela svakog pravila su **IF** i **THEN** deo, koji imaju ulogu da povežu ***uslov*** (*premisu*) sa ***zaključkom***, na primer:

IF

Auto neće da "upali" (*uslov - premisa*)

THEN

Kvar može da bude u električnom sistemu (*zaključak*)

Uslov (*premisa*) može da bude i složena, tj. da je čine više jednostavnih premisa povezanih logičkim operatorima ***AND***, ***OR*** i ***NOT***.

9. Objasniti šta je ulančavanje pravila i napisati konkretan primer koji sadrži makar tri pravila koja se ulančavaju.

Odgovor:

Ulančavanje pravila se postiže time što ***zaključak*** jednog pravila predstavlja ***uslov*** (*premisu*) drugog pravila, na primer:

IF

Auto neće da "upali" AND Napon na akumulatoru < 12V

THEN

Akumulator je prazan (*zaključak*)

IF

Akumulator je prazan (*uslov - premisa*)

THEN

Napuni akumulator

IF

Auto neće da "upali" AND Napon na akumulatoru = 12V

THEN

Anlaser je neispravan (*zaključak*)

IF

Anlaser je neispravan (*uslov - premisa*)

THEN

Zameni anlaser

*** *NAPOMENA*: treće ulančavanje smisliti po želji

10. Od čega zavisi izbor tehnike za zaključivanje?

Odgovor:

Izbor tehnike zaključivanja zavisi od korišćene ***tehnike za predstavljanje znanja***.

Najpopularnije *tehnike za zaključivanje* su:

1. ***Ulančavanje unapred*** (*Forward chaining*)
2. ***Ulančavanje unazad*** (*Backward chaining*)

i mogu da se koriste isključivo u kombinaciji sa pravilima.

11. Navesti i ukratko objasniti osnovne korake ulančavanja unapred.

Odgovor:

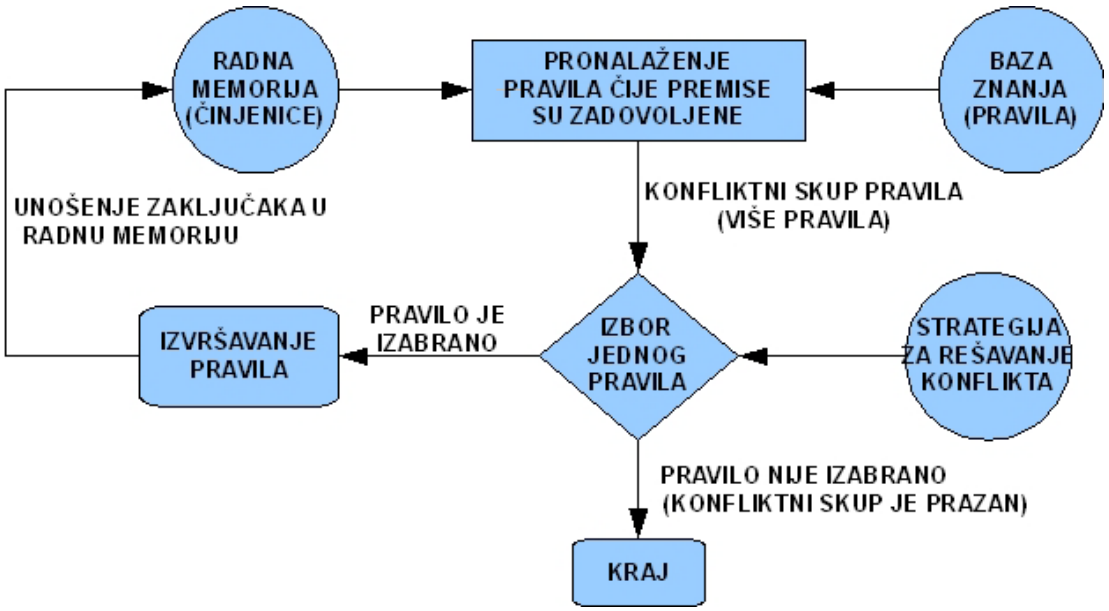
Osnovni koraci **ulančavanja unapred** su:

1. **Korak 1** - pronaći sva pravila čiji ***uslovi*** (*premise*) su zadovoljene (ova pravila čine *konfliktni skup*).
2. **Korak 2** - iz *konfliktnog skupa* izabrati samo jedno pravilo (korišćenjem *strategije za rešavanje konflikta*). Ako je *konfliktni skup* prazan, to je kraj.
3. **Korak 3** - izvršiti *izabrano pravilo* (uneti zaključke tog pravila kao činjenice u radnu memoriju) i ići na → **Korak 1**.

12. Nacrtati algoritam za ulančavanje unapred.

Odgovor:

Algoritam za ulančavanje unapred dat je na sledećoj slici:



13. Navesti bar tri različite strategije za rešavanje konflikta.

Odgovor:

Strategije za rešavanje konflikta kod *ulančavanja unapred* su:

1. Izbor *prvog pravila*
2. Izbor *pravila sa najvišim prioritetom*
3. Izbor *najspecifičnijeg pravila* (sa najsloženijim uslovom - premisom)
4. Izbor *pravila* koje se odnosi na *najskorije dodate činjenice*
5. Svako *pravilo* može *samo jednom da se izvrši*

*** Najčešće se koristi više strategija odjednom.

14. Koja je uloga strategije za rešavanje konflikta u okviru algoritma za ulančavanje unapred?

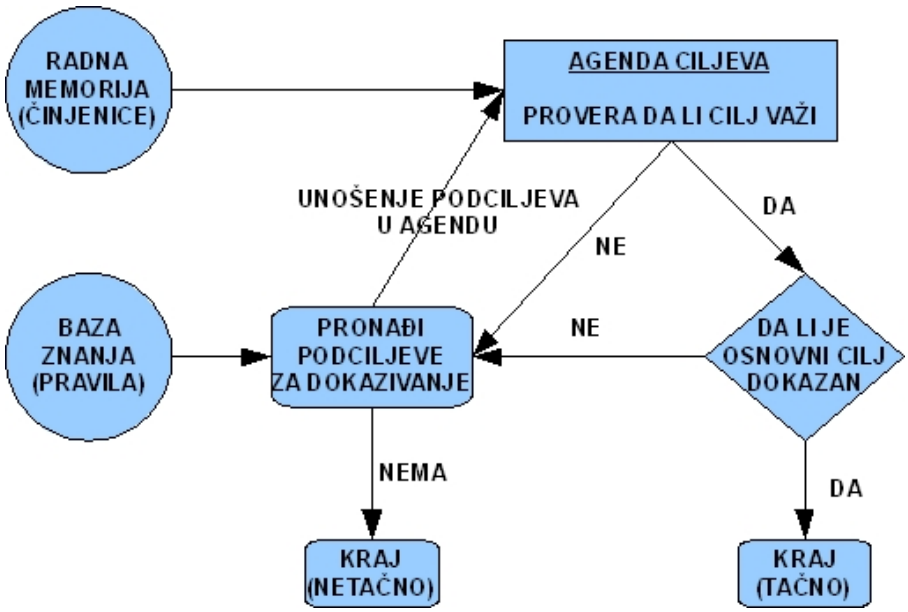
Odgovor:

Uloga ***strategije za rešavanje konflikta*** u okviru algoritma za **ulančavanje unapred** jeste da se iz *konfliktnog skupa* (sva pravila čiji *uslovi - premise* su zadovoljene) izabere samo jedno pravilo.

15. Nacrtati algoritam ulančavanja unazad.

Odgovor:

Algoritam za ulančavanje unazad dat je na sledećoj slici:



16. Koje vrste objašnjenja može da pruži mehanizam za objašnjavanje? Ukratko opisati svaku vrstu objašnjenja.

Odgovor:

Vrste objašnjenja koje *mehanizam za objašnjavanje* može da pruži su:

1. **ZAŠTO** - objašnjenje o tome zašto ES postavlja određeno pitanje.
2. **KAKO** - objašnjenje o tome kako je ES stigao do rešenja.

17. Navesti bar 4 oblasti primene ES.

Odgovor:

Ekspertni sistemi se primenjuju u sledećim ***oblastima*** [1] :

1) Upravljanje industrijskim procesima
2) Praćenje rada medicinskih uređaja
3) Autonomno kretanje vozila (na zemlji i vodi)
4) Automatski piloti
5) Upravljanje satelitima
6) Nadgledanje instalacija
7) Operativno i taktičko upravljanje vojnim operacijama na bojnom polju
8) Analize složenih finansijskih transakcija itd.

[1] "Veštačka inteligencija", Velibor Ilić (novembar 1999) , http://solair.eunet.rs/~ilicv/AI_index.htm (21.1.2014. 20:15)

18. Navesti osnovne uloge u razvoju ES.
19. Koja je uloga eksperta u razvoju ES?
20. Koja je uloga inženjera znanja u razvoju ES?
21. Koja je uloga korisnika u razvoju ES?

Odgovor na 18,19,20 i 21. pitanje:

Osnovne uloge u **razvoju ES** su:

1. ***Ekspert*** - koji *daje ("pozajmljuje") svoje znanje* i pomaže pri proveri (testiranju) znanja

2. ***Inženjer znanja*** - koji:
   a) Vodi intervju sa ekspertom i iz njega "izvlači" znanje
   b) Vrši izbor:
      - Odgovarajućih tehnika za predstavljanje znanja
      - Odgovarajućih tehnika za zaključivanje
      - Razvojnog alata
   c) Formalizuje, formuliše i "sređuje" ekspertovo znanje
   d) Testira ES

3. ***Korisnik*** - koji:
   a) Koristi gotov ES
   b) Učestvuje u formiranju zahteva
   c) Može da učestvuje u testiranju i pisanju dokumentacije za ES

22. Nacrtati proces razvoja ES.

Odgovor:

Pitanja iz oblasti MAŠINSKO UČENjE

1. Kako definišemo mašinsko učenje?

Odgovor:

Mašinsko učenje se definiše kao sposobnost *softverskog sistema* da:

- *generalizuje* na osnovu prethodnog iskustva (podataka), i
- koristi generalizacije kako bi pružio odgovore na pitanja koja se odnose na podatke koje pre nije sretao

2. U kojim slučajevima (tj. za koje vrste problema) je mašinsko učenje posebno korisno? Ukratko objasniti.

Odgovor:

Mašinsko učenje je posebno korisno za sledeće vrste problema:

1) Kod zadataka koje ljudi rešavaju vrlo lako, a pri tome nisu u mogućnosti da precizno (algoritamski) opišu kako to rade. ***Primer***: prepoznavanje slika, zvuka, govora

2) Kod zadataka gde se mogu definisati algoritmi za rešavanje, ali su ti algoritmi vrlo složeni i/ili zahtevaju velike baze znanja. ***Primer***: automatsko prevođenje

3) U mnogim oblastima gde se kontinuirano prikupljaju podaci sa ciljem da se iz njih "nešto sazna".

Primer: u medicine (podaci o pacijentima i korišćenim terapijama), sportu (o odigranim utakmicama i igri pojedinih igrača), marketingu (o korisnicima/kupcima i tome šta su kupili, za šta su se interesovali, kako su proizvode ocenili itd.)

3. Koje su osnovne karakteristike nadgledanog mašinskog učenja?

Odgovor:

Osnovne karakteristike **nadgledanog mašinskog učenja** su da algoritam za učenje dobija:

1. Skup ulaznih podataka ($\mathbf{x_1}$, $\mathbf{x_2}$, …, $\mathbf{x_n}$) i
2. Skup željenih/tačnih vrednosti, tako da za svaki ulazni podatak $\mathbf{x_i}$, imamo željeni/tačan izlaz $\mathbf{y_i}$

Zadatak mašine je da "nauči" kako da novom, neobeleženom ulaznom podatku dodeli tačnu izlaznu vrednost.

Izlazna vrednost može biti:

a) *Labela* (tj. nominalna vrednost) - reč je o klasifikaciji

b) *Realan broj* - reč je o regresiji

4. Koje su osnovne karakteristike nenadgledanog mašinskog učenja?

Odgovor:

Osnovne karakteristike **nenadgledanog mašinskog učenja** su da:

1. Nemamo informacija o željenoj izlaznoj vrednosti
2. Algoritam dobija samo skup ulaznih podataka ($\mathbf{x_1}$, $\mathbf{x_2}$, …, $\mathbf{x_n}$)

Zadatak algoritma je da otkrije paterne tj. skrivene strukture/zakonitosti u podacima.

5. Koje su osnovne karakteristike učenja uz podsticaje?

Odgovor:

Osnovne karakteristike **učenja uz podsticaje** su da:

1) Program (agent) deluje na okruženje izvršavanjem niza akcija

2) Akcije utiču na stanje okruženja, koje povratno utiče na agenta pružajući mu povratne informacije koje mogu biti "nagrade" ili "kazne"

3) Cilj agenta je da nauči kako da deluje u datom okruženju tako da vremenom maximizira nagrade (ili minimizira kazne)

6. Koji su osnovni koraci procesa mašinskog učenja?

Odgovor:

Osnovni koraci procesa **mašinskog učenja** su:

1) Prikupljanje podataka potrebnih za formiranje dataset-ova za obuku, validaciju i testiranje algoritama mašinskog učenja

2) Priprema podataka, što tipično podrazumeva "čišćenje" i transformaciju podataka

3) Analiza rezultujućih dataset-ova, i njihovo, eventualno, dalje unapređenje kroz selekciju/transformaciju atributa

4) Izbor jednog ili više algoritama za obuku nad kreiranim trening dataset-om

5) Evaluacija izabranih algoritama na dataset-u za validaciju

6) Izbor algoritma koji će se koristiti (na osnovu rezultata *koraka 5*) i njegovo testiranje na test dataset-u

7. Kakva je uloga podataka u procesu mašinskog učenja? U kojim fazama procesa se podaci direktno koriste i kako se ukupno raspoloživi podaci dele na te faze?

Odgovor:

Uloga podataka u procesu mašinskog učenja je u ***formiranju dataset***-ova koji se direktno koriste u fazama ***obuke***, ***validacije*** i ***testiranja*** *algoritama mašinskog učenja*, gde se ukupno raspoloživi podaci dele:

- ***60***% za trening (obuku),
- ***20***% za validaciju i
- ***20***% za testiranje

8. Koja je uloga atributa (features) u procesu mašinskog učenja?

Odgovor:

Uloga **atributa** (*features*) u procesu mašinskog učenja je da:

1. Najbolje opišu neke pojave/entitete koje prepoznajemo uočavajući njihove osobine (ili izostanak nekih osobina) i uviđajući odnose između različitih osobina

2. Omoguće programu da koristi osobine pojava/entiteta za potrebe njihove identifikacije/grupisanja

9. Kojim faktorima je, generalno posmatrano, uslovljen odabir algoritma mašinskom učenja?

Odgovor:

Odabir algoritma mašinskom učenja je, generalno posmatrano, uslovljen sledećim *faktorima*:

1. ***Vrstom problema*** koji rešavamo,
2. ***Karakteristikama skupa atributa*** (*features*):
    a) *Tipom atributa* i *stepenom homogenosti tipova atributa*
    b) *Stepenom međuzavisnosti* (*korelisanosti*) *atributa*
3. ***Obimom podataka*** koji su nam na raspolaganju

10. Koja je uloga validacije u procesu mašinskog učenja?

Odgovor:

Uloga **validacije** u *procesu mašinskog učenja* je da se:

a) Izabere najbolji model/algoritam između više kandidata

b) Odredi optimalna konfiguracija parametara modela

c) Izbegnu problemi over/under-fitting-a

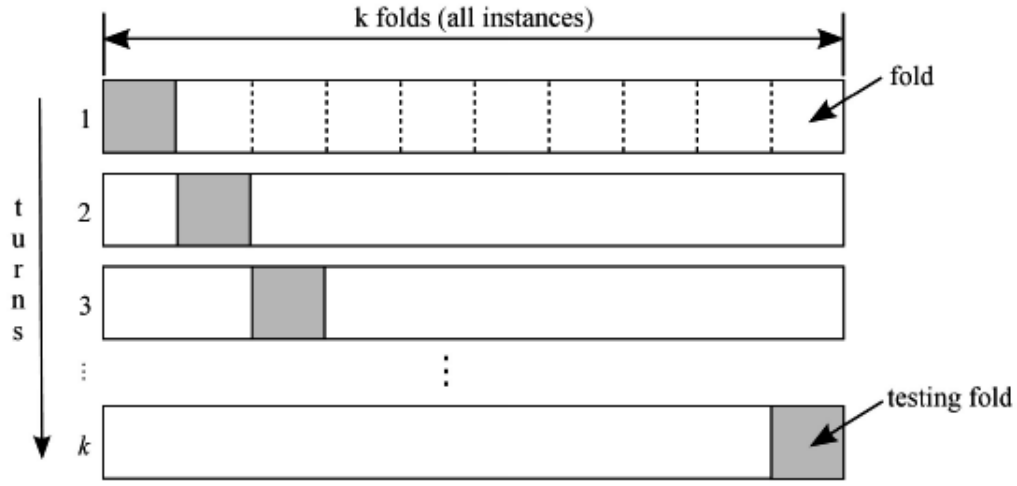
11. Šta je kros-validacija (cross-validation)? Ukratko objasniti kako funkcioniše.

Odgovor:

Kros-validacija (*cross-validation*) je pristup za efikasno korišćenje raspoloživih podataka.

Ona funkcioniše na sledeći način:

1. Raspoloživi skup podataka za trening se podeli na ***K*** delova ili podskupova (folds)
   - Najčešće se uzima 10 podskupova (10 fold cross validation)
2. Zatim se obavlja ***K*** iteracija (trening + validacija) algoritma, a u svakoj iteraciji:
   - Uzima se 1 deo podataka za potrebe validacije, a ostatak (***K-1*** deo) se koristi za učenje
   - Bira se uvek različiti podskup koji će se koristiti za validaciju



Pri svakoj iteraciji računaju se performanse algoritma.

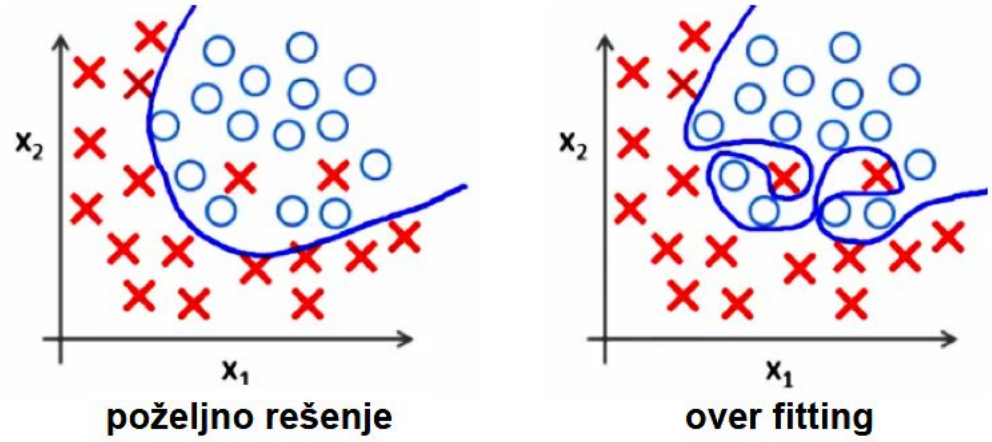
Na kraju se računa prosečna uspešnost na nivou svih ***K*** iteracija - tako izračunate mere uspešnosti daju bolju sliku o performansama algoritma.

Ukoliko su rezultati u svih ***K*** iteracija vrlo slični, smatra se da je procena uspešnosti algoritma pouzdana.

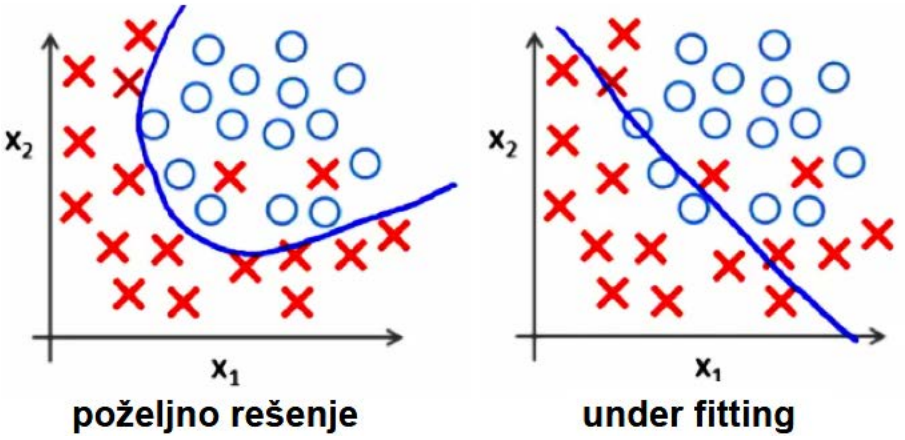
12. Šta označavaju termini over-fitting i under-fitting? Ukratko objasniti.

Odgovor:

Problem prevelikog podudaranja (*over-fitting*) se odnosi na situaciju u kojoj algoritam savršeno nauči da prepoznaje instance iz trening seta, ali nije u mogućnosti da prepozna instance koje se i malo razlikuju od naučenih:



Problem nedovoljnog podudaranja (*under-fitting*) se odnosi na slučaj kad algoritam ne uspeva da aproksimira podatke za trening, tako da ima slabe performance čak i na trening setu:



13. Kako definišemo zadatak klasifikacije?

Odgovor:

Zadatak **klasifikacije** je određivanje klase kojoj neka instanca pripada (instanca je opisana vrednošću atributa, a skup mogućih klasa je poznat i dat).

14. U slučaju primene Naive Bayes algoritma za klasifikaciju teksta, koji je tipičan pristup za formiranje vektora atributa? Ukratko objasniti.

Odgovor:

U slučaju primene *Naive Bayes algoritma* za *klasifikaciju teksta*, tipičan pristup za ***formiranje vektora atributa*** je sledeći:

1. Estrahovati reči iz dokumenata koji čine skup za trening ***D***, i formirati tzv. rečnik ***R***
2. Za svaki dokument ***d*** iz skupa ***D*** definisati skup atributa (feature vector) na osnovu reči iz kojih se ***d*** sastoji:
   - Za svaku reč $\mathbf{r_i}$ iz dokumenta ***d*** uvodi se atribut $\mathbf{x_i}$ čija vrednost je indeks reči $\mathbf{r_i}$ u rečniku ***R***
   - Atributi mogu biti kreirani za sve reči dokumenta ***d*** ili samo za one reči koje su značajne za dati zadatak klasifikacije

Primer:

I **love** this movie! It's **sweet**, but with **satirical** humor. The dialogue is **great** and the adventure scenes are **fun**… It manages to be **whimsical** and **romantic** while **laughing** at the conventions of the fairy tale genre. I would **recommend** it to just about anyone. I've seen it **several** times, and I'm always **happy** to see it **again** whenever I have a friend who hasn't seen it yet.

reči (r_i)	atributi (x_i)
r_1 = love	x_1 = 04567
r_2 = sweet	x_2 = 14321
r_3 = satirical	x_3 = 14007
…	
r_{11} = happy	x_{11} = 02364
r_{12} = again	x_{12} = 00012

indeks reči r_i u Rečniku ***R***

15. Koje pretpostavke uvodi Naive Bayes algoritam (koje ga i čine naivnim)?

Odgovor:

Pretpostavke koje uvodi *Naive Bayes algoritam* su:

- Dokument ***d*** posmatramo kao prost skup reči (*bag-of-words*); tj. pozicija i redosled reči u tekstu se smatraju nevažnim
- Pojavljivanje određene reči u datoj klasi ***c*** je nezavisno od pojavljivanja neke druge reči u toj klasi

16. Navesti osobine Naive Bayes algoritma.

Odgovor:

Osobine Naive Bayes algoritma su sledeće:

1) Veoma je brz i efikasan
2) Najčešće daje dobre rezultate
   - često se pokazuje kao bolji ili bar podjednako dobar kao drugi, sofisticiraniji modeli
3) Nije memorijski zahtevan
4) Ima vrlo mali afinitet ka preteranom podudaranju sa podacima za trening (overfitting)
5) Pogodan kada imamo malu količinu podataka za trening
6) "Otporan" na nevažne atribute
   - atributi koji su podjednako distribuirani kroz skup podataka za trening, pa nemaju veći uticaj na izbor klase
7) Namenjen primarno za rad sa nominalnim atributima, a u slučaju numeričkih atributa:
   - koristiti raspodelu verovatnoća atributa (tipično Normalna raspodela) za procenu verovatnoće svake od vrednosti atributa
   - uraditi diskretizaciju vrednosti atributa

17. Navesti mere koje se tipično koriste za procenu uspešnosti modela klasifikacije.

Odgovor:

Mere koje se tipično koriste za procenu uspešnosti modela klasifikacije su:

1. ***Matrica zabune*** (*Confusion Matrix*)
2. ***Tačnost*** (*Accuracy*)
3. ***Preciznost*** (*Precision*) i ***Odziv*** (*Recall*)
4. ***F mera*** (*F measure*)
5. ***Površina ispod ROC krive*** (*Area Under the Curve - AUC*)

18. Kako definišemo i izračunavamo Tačnost (Accuracy) kao meru uspešnosti modela klasifikacije?

Odgovor:

Tačnost (*Accuracy*) se definiše kao procenat slučajeva (instanci) koji su uspešno (korektno) klasifikovani, a izračunava se kao:

$$\textbf{Accuracy} = (\textbf{TP} + \textbf{TN}) / \textbf{N}$$

gde je:

TP - True Positive

TN - True Negative

N - ukupan broj uzoraka (instanci) u skupu podataka

19. U kom slučaju se Tačnost (Accuracy) ne može smatrati pouzdanom merom uspešnosti klasifikacionog modela? Ukratko objasniti.

Odgovor:

U slučaju vrlo neravnomerne raspodele podataka između klasa (tzv. skewed classes), ova mera je nepouzdana.

Na primer: U slučaju klasifikacije poruka na spam vs. not-spam, možemo imati skup za trening sa 0.5% spam poruka. Ako primenimo "klasifikator" koji svaku poruku svrstava u not-spam klasu, dobijamo tačnost od 99.5% .

20. Navesti karakteristike F mere (F measure). U čemu se ogleda razlika između F mere i F1 mere?

Odgovor:

F mera kombinuje ***Preciznost*** (*Precision*) i ***Odziv*** (*Recall*) i omogućuje jednostavnije poređenje dva ili više algoritama. Računa se kao:

$$\mathbf{F} = (1 + \beta^2) * \mathbf{Precision} * \mathbf{Recall} / (\beta^2 * \mathbf{Precision} + \mathbf{Recall})$$

* Parametar **β** kontroliše koliko više značaja će se pridavati *Odzivu* u odnosu na *Preciznost*.

U praksi se najčešće koristi tzv. **F1 mera** („*balansirana*“ **F mera**) koja daje podjednak značaj i *Preciznosti* i *Odzivu*:

$$\mathbf{F1} = 2 * \mathbf{Precision} * \mathbf{Recall} / (\mathbf{Precision} + \mathbf{Recall})$$

21. Navesti karakteristike mere AUC (Area Under the Curve).

Odgovor:

Karakteristike mere **AUC** (*Area Under the Curve*) su sledeće:

1) Meri diskriminacionu moć klasfikatora tj. sposobnost da razlikuje instance koje pripadaju različitim klasama
2) Primenjuje se za merenje performansi binarnih klasifikatora
3) Vrednost za ***AUC*** se kreće u intervalu ***0***-***1***
4) Za metodu slučajnog izbora važi da je ***AUC*** = ***0.5***, a što je ***AUC*** vrednost klasifikatora ***> 0.5***, to je klasifikator bolji
   - **0.7**-**0.8** se smatra prihvatljivim; **0.8**-**0.9** jako dobrim; sve **> 0.9** je odlično

22. Kako definišemo zadatak klasterizacije?

Odgovor:

Zadatak **klasterizacije** jeste grupisanja instanci, tako da za svaku instancu važi da je sličnija instancama iz svoje grupe (klastera), nego instancama iz drugih grupa (klastera).

23. Opisati osnovne korake K-Means algoritma.

Odgovor:

Osnovni koraci **K-Means algoritma** su:

1. Inicijalni izbor težišta klastera, slučajnim izborom
   - težišta se biraju iz skupa instanci za trening, tj. ***K*** instanci za trening se nasumično izabere i proglasi za težišta
2. Ponoviti dok algoritam ne konvergira ili broj iteracija ***<= max***:
   1) *Grupisanje po klasterima*: za svaku instancu iz skupa za trening, ***i = 1***, ***m***, identifikovati najbliže težište i dodeliti instance klasteru kome to težište pripada
   2) *Pomeranje težišta*: za svaki klaster izračunati novo težište uzimajući prosek tačaka (instanci) koje su dodeljene tom klasteru

24. Koji se kriterijumi tipično koriste za procenu kvaliteta klastera formiranih u procesu klasterizacije?

Odgovor:

Kriterijumi koji se tipično koriste za procenu kvaliteta *klastera* formiranih u procesu *klasterizacije* su:

1. ***Međusobna udaljenost težišta*** - što su težišta dalje jedno od drugog, to je stepen preklapanja klastera manji, i njihov kvalitet viši
2. ***Standardna devijacija pojedinačnih instanci*** u odnosu na *težište* - što je standardna devijacija manja, to su instance tešnje grupisane oko težišta i klasteri se smatraju boljim
3. ***Suma kvadrata greške*** unutar ***klastera*** (within cluster sum of squared errors) - daje kvantitativnu meru za procenu kvaliteta kreiranih klastera

25. Kako se prevazilazi problem K-Means algoritma uslovljen nasumičnim (random) inicijalnim izborom težišta? Ukratko objasniti.

Odgovor:

Problem **K-Means algoritma** uslovljen *nasumičnim (random) inicijalnim izborom težišta* se prevazilazi korišćenjem ***višestruke nasumične inicijalizacije***, koja omogućuje da se izbegnu situacije koje **K-Means** dovode u *lokalni minimum*.

Sastoji se u sledećem:

```
for i = 1 to n { // n obično uzima vrednosti od 50 do 1000
        Nasumično odabrati inicijalni skup težišta;
        Izvršiti K-Means algoritam;
        Izračunati funkciju koštanja (cost function)
}
Izabrati instance algoritma koja daje najmanju vrednost za f. koštanja
```

Ovaj pristup daje dobre rezultate ukoliko je broj klastera relativno mali (2 - 10), a za veći broj klastera ne bi ga trebalo koristiti.

26. Koji su pristupi za procenu broja klastera (tj. parametra K) pri primeni K-Means algoritma?

Odgovor:

Pristupi za procenu broja klastera (tj. parametra ***K***) pri primeni **K-Means algoritma** su:

1. U slučaju da ***posedujemo znanje o fenomenu/pojavi*** koju podaci opisuju:
   - Pretpostaviti broj klastera (***K***) na osnovu domenskog znanja
   - Testirati model sa ***K-1***, ***K***, ***K+1*** klastera i uporediti grešku
2. Ukoliko ***ne posedujemo znanje o fenomenu/pojavi*** :
   - Krenuti od malog broja klastera i u više iteracija testirati model uvek sa jednim klasterom više
   - U svakoj od iteracija, uporediti grešku tekućeg i prethodnog modela i kad smanjenje greške postaje zanemarljivo, prekinuti postupak

Zadatak

Za dati opis jednog konkretnog zadatka binarne klasifikacije (npr. klasifikacija email poruka na spam i not-spam), interpretirati matricu zabune (confusion matrix), tj. objasniti značenje polja matrice (vrednosti TP, TN, FP i FN) u kontekstu datog konkretnog zadatka. Potrebno je takođe, izračunati vrednosti sledećih mera:
Tačnost (Accuracy), Preciznost (Precision) i Odziv (Recall).

Pitanja iz oblasti NEURONSKE MREŽE

1. Šta je neuronska mreža?

Odgovor:

Neuronska mreža je *paralelni distribuirani procesor* koji ima prirodnu sposobnost čuvanja i korišćenja iskustvenog znanja.
Sličnost sa mozgom se ogleda kroz dve osobine:

- Mreža stiče znanje kroz proces učenja
- Znanje se čuva u vezama između neurona (sinaptičkim težinama)

2. Koje su osnovne komponente veštačkog neurona?

Odgovor:

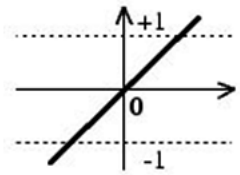
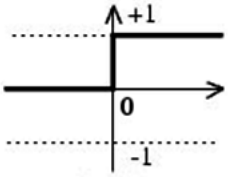
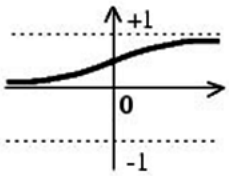
Osnovne komponente **veštačkog neurona** su:

1. ***Ulazna funkcija sumiranja***
2. ***Funkcija transfera***
3. ***Ulazi sa težinskim koeficijentima***
4. ***Izlaz***

3. Navesti i nacrtati grafike osnovnih funkcija prenosa koje se koriste u neuronskim mrežama.

Odgovor:

Osnovne funkcije prenosa koje se koriste u ***neuronskim mrežama*** su:

Linearna	
Odskočna	
Sigmoidna	

4. Navesti osnovne karakteristike neuronskih mreža.

Odgovor:

Osnovne karakteristike **neuronskih mreža** su:

1. Imaju *sposobnost učenja*
2. Imaju *sposobnost generalizacije*
3. *Otporne* na *pogrešan ulaz* i *šum*

5. Navesti tipične probleme za koje probleme se koriste neuronske mreže.

Odgovor:

Tipični problemi za koje se *koriste* **neuronske mreže** su:

1) ***Klasifikacija***
2) ***Prepoznavanje*** (oblika, govora, vektora...)
3) ***Aproksimacija***
4) ***Optimizacija***
5) ***Obrada signala***
6) ***Modeliranje sistema***
7) ***Predviđanje***
8) ***Kontrola*** i ***upravljanje***

6. Navesti tipične slučajeva kada su neuronske mreže kandidat za primenu.

Odgovor:

Tipični slučajevi kada su **neuronske mreže** *kandidat za primenu* su:

- Kada nema jasno definisanog matematičkog modela ili drugog rešenja
- Kada je potrebna otpornost na nepotpun ili pogrešan ulaz
- Kada je potrebna sposobnost učenja
- Visokodimenzionalnost
- Kada se sa NM postižu bolji rezultati nego sa alternativnim rešenjima (npr. odziv u realnom vremenu, tolerancija na greške)

7. Objasniti u čemu se sastoji učenje kod neuronskih mreža.

Odgovor:

Učenje kod **neuronskih mreža** se sastoji od podešavanja težina veza tako da mreža dobije željeno ponašanje/funkcionalnost.

8. Koji su osnovni problemi u primeni neuronskih mreža?

Odgovor:

Osnovni problemi u *primeni* **neuronskih mreža** su:

- Nedostatak semantike u strukturi
- Da li je neki problem uopšte rešiv sa NM
- Problemi sa određivanjem arhitekture i treningom za određenu primenu
- Plastičnost /stabilnost

9. Navesti nekoliko vrsta neuronskih mreža sa prostiranjem signala unapred.

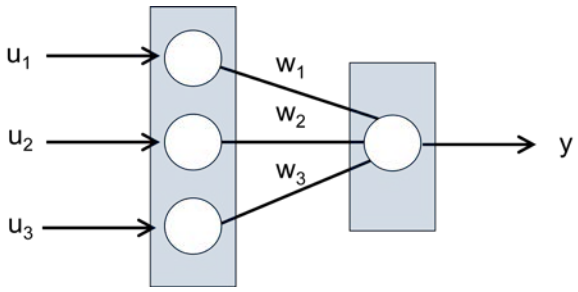
Odgovor:

Vrste **neuronskih mreža** sa *prostiranjem signala unapred* mogu biti:

1. **ADALINE** - odlikuju je:
   - Linearna funkcija transfera
   - Linearna kombinacija ulaza:

$$\mathbf{y = w_1u_1 + w_2u_2 + \ldots + w_nu_n}$$

   - Učenje metodom najmanjih kvadrata



2. **LMS učenje** - može izraziti kroz sledeće jednačine:
   - Greška izlaznog neurona za **p**-ti uzorak iz skupa za trening

$$\mathbf{\varepsilon_p = d_p - y_p}$$

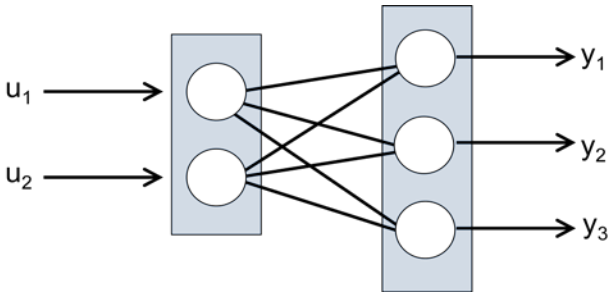
   - Promena težine veze proporcionalno grešci

$$w_{ji}(k+1) = w_{ji}(k) + \mu\varepsilon(k)u_{ji}(k)$$

   - Ukupna greška mreže za sve uzorke iz skupa za trening (kriterijum za zaustavljanje treninga, tj. mreža je naučila kada je greška svedena na prihvatljivu meru)

$$E = \frac{1}{2n}\sum_{p=1}^{n}\varepsilon_p{}^2$$

3. **PERCEPTRON** - odlikuju je:
   - Step funkcija transfera
   - *Perceptron learning* - prvi algoritam za učenje nelinearnih sistema
   - Samo za linearno separabilne probleme



10. Šta je osnovno ograničenje kod neuronske mreže tipa Perceptron?

Odgovor:

Osnovno ograničenje kod **neuronske mreže** tipa ***Perceptron*** je to što je ona predviđena samo za linearno separabilne probleme.

11. Koji algoritam se koristi za učenje neuronskih mreža tipa višeslojni Perceptron (Multi Layer Perceptron) ?

Odgovor:

Algoritam koji se koristi za *učenje neuronskih mreža* tipa ***višeslojni Perceptron*** (*Multi Layer Perceptron*) je **Backpropagation algoritam** čije su osobine:

- Služi za *trening višeslojnog perceptrona* - može da podešava težine u skrivenim slojevima
- Predstavlja *supervizorni algoritam* koji se zasniva na LMS algoritmu
- Višeslojni perceptron sa Backpropagation algoritmom predstavlja *univerzalni aproksimator*

12. Opisati proceduru rešavanja problema pomoću neuronskih mreža.

Odgovor:

Procedura *rešavanja problema* pomoću **neuronskih mreža** se sastoji od:

1) Prikupljanja i pripreme podataka
2) Treninga mreže
3) Testiranja mreže
4) Određivanja optimalnih parametara mreže i treninga eksperimentalnim putem (broj neurona, broj slojeva neurona, parametri algoritma za učenje, podaci za trening)

Pitanja iz oblasti ANALIZA i RAZUMEVANjE TEKSTA

1. Navesti razloge zbog kojih razumevanje teksta predstavlja vrlo složen problem.

Odgovor:

Razlozi zbog kojih razumevanje teksta predstavlja vrlo složen problem su ti što je prirodni jezik:

- Pun višesmislenih reči i izraza
- Zasnovan na korišćenju konteksta za definisanje i prenos značenja
- Pun fuzzy, probabilističkih izraza
- Baziran na zdravorazumskom znanju i rezonovanju
- Pod uticajem je i sam utiče na interakcije među ljudima

2. Navesti i ukratko obrazložiti različite nivoe razumevanja jezika.

Odgovor:

Različiti ***nivoi razumevanja jezika*** i njihova obrazloženja data su u sledećoj tabeli:

	Opis	Primer
Morfologija	prepoznavanje reči i njihovih različitih formi	use, uses, user – različiti oblici jedne reči
Sintaksa i Gramatika	prepoznavanje tipa reči	There are 5 *rows* in the table. – rows je imenica; She *rows* 5 times per week. – rows je glagol
	razumevanje kako su reči povezane	Bob went out; *he* needed some fresh air. – Zamenica he se odnosi na Bob-a.
Semantika	razumevanje značenja reči (na osnovu konteksta)	The car *driver* was injured. vs. The *driver* was installed in the computer

3. Navesti i ukratko obrazložiti osnovne pristupe modelovanju jezika.

Odgovor:

*Osnovni **pristupi modelovanju jezika*** su:

1. **Logički modeli**
   - Zasnovani na lingvističkoj analizi teksta i formiranju apstraktnog modela strukture rečenica (tipično u formi stabla)
   - Lingvistička analiza vodi ka ručnom kreiranju modela

2. **Stohastički modeli**
   - Zasnovani na verovatnoći pojavljivanja pojedinačnih reči ili niza od ***n*** reči (najčešće 2-4 reči)*
   - Ovi modeli se "uče" tj. kreiraju se kroz primenu algoritama mašinskog učenja nad velikim korpusima teksta

3. **Hibridni pristup**
   - Kombinuje logički i stohastički pristup
   - Na primer, pridruživanje verovatnoća pojedinim elementima stabloidnog modela strukture rečenice

4. Šta je to ekstrakcija informacija?

Odgovor:

Ekstrakcija informacija je tehnologija zasnovana na *analizi prirodnog jezika* sa ciljem ekstrakcije informacija o predefinisanim tipovima *entiteta*, *relacija* i/ili *događaja*.

5. Objasniti razliku između ekstrakcije informacija (information extraction) i pronalaženja informacija (information retrieval).

Odgovor:

Razliku između **ekstrakcije informacija** (*information extraction* - **EI**) i **pronalaženja informacija** (*information retrieval* - **IR**) je ta što ***IR*** sistem pronalazi (*potencijalno*) *relevantne tekstove* i prezentuje ih korisniku, dok ***EI*** sistem analizira tekstove i prezentuje samo *segmente informacija* (izvučene iz teksta) za koje korisnik može biti zainteresovan.

6. Navesti osnovne tipove zadataka kojima se oblast ekstrakcije informacija bavi.

Odgovor:

Osnovni tipovi zadataka kojima se oblast **ekstrakcije informacija** bavi su:

1. ***Prepoznavanje imenovanih entiteta*** (*Named Entity recognition*) - može se odnositi na različite vrste entiteta (ljudi, organizacije, datumi, valute i slično)

2. ***Razrešavanje koreferenci*** (*Co-reference resolution*) - obuhvata:
    1) *Anaphoric resolution* - na primer, utvrditi da se u tekstu: "Tom is my best friend. I know him since we were kids." zamenica 'him' odnosi na imenicu 'Tom';
    2) *Proper noun resolution* - na primer, utvrditi da sledeće imenice označavaju isti entitet: 'IBM', 'IBM Europe', 'International Business Machines Ltd.' ...

3. ***Prepoznavanje opisa entiteta*** (*Descriptions resolution*) - koje atribute entiteti imaju?
4. ***Prepoznavanje relacija*** (*Relations resolution*) - koje relacije postoje među entitetima?
5. ***Prepoznavanje događaja*** (*Events resolution*) - identifikacija događaja u kojima entiteti učestvuju

7. Navesti činioce koji utiču na performanse procesa ekstrakcije informacija.

Odgovor:

Činioci koji utiču na *performanse* procesa **ekstrakcije informacija** su:

1. ***Specifičnosti konkretnog EI zadatka***:
    1) *Tip teksta* - vrsta teksta sa kojim se radi; npr. novinski članci ili email poruke ili poslovni izveštaji ili naučni radovi I slično
    2) *Tema* (ili *domen*) - šire definisan opseg tema (domen) kome sadržaj teksta pripada
    3) *Stil pisanja* - nivo formalnosti jezika, korišćenje stručne terminologije i slično
    4) Konkretni *tipovi informacija* za koje je korisnik zainteresovan

2. ***Kompleksnosti EI zadatka***

8. Objasniti meru Preciznost (Precision) za procenu efikasnosti ekstrakcije informacija - šta ona predstavlja i kako se izračunava?

Odgovor:

Preciznost (*precision*) je mera za procenu efikasnosti ekstrakcije informacija koja govori da li su svi *estrahovani segmenti informacija* relevantni.

Izračunava se na osnovu sledeće tabele:

	Tačno	***Pogrešno***
Estrahovani	*A*	*B*
Nisu estrahovani	*C*	*D*

kao:

$$\textbf{Precision} = \textbf{A} / (\textbf{A} \cup \textbf{B})$$

9. Objasniti meru Odziv (Recall) za procenu efikasnosti ekstrakcije informacija - šta ona predstavlja i kako se izračunava?

Odgovor:

Odziv (*recall*) je mera za procenu efikasnosti ekstrakcije informacija koja govori da li su svi *relevantni segmenti informacija* prepoznati.

Izračunava se na osnovu sledeće tabele:

	Tačno	***Pogrešno***
Estrahovani	*A*	*B*
Nisu estrahovani	*C*	*D*

kao:

$$\textbf{Recall} = \textbf{A} / (\textbf{A} \cup \textbf{C})$$

10. U kakvom su odnosu mere Preciznost (Precision) i Odziv (Recall)? Ukratko objasniti.

Odgovor:

Mere **Preciznost** (*Precision*) i **Odziv** (*Recall*) su u "konfliktu":

- Možemo razviti sistem koji neće praviti mnogo grešaka (visoka preciznost), ali će propustiti da prepozna puno relevantnih informacija (nizak odziv)
- Alternativno, možemo staviti akcenat na odziv i propustiti manje relevantnih informacija, ali po ceni pravljenja više grešaka

11. Koja su to dva osnovna pod-problema od kojih se sastoji problem razrešavanja koreferenci (co-reference resolution)?

Odgovor:

Dva osnovna *pod-problema* od kojih se sastoji *problem* ***razrešavanja koreferenci*** (*co-reference resolution*) su:

1. ***Glavni domen primene*** - pridruživanje deskriptivnih informacija "rasutih" po tekstu entitetima na koje se odnose
2. ***Performanse***:
    - Neprecizan proces
    - Rezultati značajno variraju od domena do domena (domenski zavistan zadatak)
    - Zavisno od domena, preciznost je na nivou 50-60%

12. Navesti osnovne problem (izazove) sa kojima se oblast prepoznavanja entiteta u tekstu susreće.

Odgovor:

Osnovni *problemi* (*izazovi*) sa kojima se oblast **prepoznavanja entiteta u tekstu** susreće su:

1. *Pravilna identifikacija segmenata teksta* kojima su *entiteti* predstavljeni (tzv. chunking)
    - entiteti mogu biti predstavljeni jednom reči (npr. MIT) ili nizom reči (Massachusetts Institute of Technology)
2. Zaključivanje da određeni *segment teksta* stvarno predstavlja *entitet*
    - posebno nezgodno u slučajevima kad se višeznačne reči nađu na početku rečenice (npr., May, Galaxy, …)
3. Određivanje *tipa entiteta*
4. Prepoznavanje *segmenata teksta* koji se odnose na *isti entitet*
    - Problem: različiti načini referenciranja na isti entitet; primeri: John Smith; Mr Smith; John ili UMBC; University of Maryland Baltimore County
5. Održavanje *ažurnim lista/rečnika* koji sadrže *nazive entiteta* (potrebni za većinu aktuelnih sistema)

13. Navesti osnovne grupe pristupa za prepoznavanje entiteta u tekstu.

Odgovor:

Osnovne *grupe pristupa* za **prepoznavanje entiteta u tekstu** su:

1. ***List lookup*** pristupi - zasnovani na korišćenju rečnika i gazetter lista
2. Pristupi zasnovani na ***pravilima***
    a) *Shallow parsing* pristupi
    b) pristupi zasnovani na *regularnim izrazima*
3. Pristupi zasnovani na ***mašinskom učenju***
4. Pristupi zasnovani na ***mašinskom učenju*** i ***bazama znanja***
5. ***Hibridni*** pristupi - kombinuju dva ili više navedenih pristupa i najčešće se primenjuju u praksi

14. Navesti osnovne karakteristike list lookup pristupa za prepoznavanje entiteta u tekstu.

Odgovor:

Osnovne *karakteristike* **list lookup** pristupa za prepoznavanje entiteta u tekstu su:

1) Primenjuju se kad imamo *unapred date liste imena entiteta* koje *tražimo* - na primer liste kompanija i/ili eksperata iz određene branše

2) Prepoznaju samo one *entitete* čija *imena* su prisutna u *listama/rečniku*

15. Navesti prednosti i nedostatke list lookup pristupa za prepoznavanje entiteta u tekstu.

Odgovor:

Prednosti ***list lookup*** pristupa:

- Jednostavnost
- Brzina (brži u odnosu na ostale pristupe)
- Nezavisni od jezika
- Mogućnost jednostavne adaptacije na nove vrste teksta

Nedostaci ***list lookup*** pristupa:

- Kreiranje/prikupljanje i održavanje lista imena
- Ne mogu da prepoznaju entitete u slučaju slabog preklapanja imena u listama i u tekstu
- Nemaju mogućnost razumevanja entiteta u kontekstu i razrešavanja dvosmislenosti (ili višeznačnosti)

16. U čemu se sastoji Shallow parsing pristup za prepoznavanje entiteta u tekstu. Ukratko objasniti.

Odgovor:

Shallow parsing *pristup* za prepoznavanje entiteta u tekstu se oslanja na heuristička, iskustvena 'pravila' vezana za:

- strukturu teksta i pojedinačnih rečenica,
- tipično korišćene izraze i fraze u tekstu

Ideja je identifikovati uobičajene jezičke formulacije i predstaviti ih u formi templejta. Identifikovani templejti se, zatim, mogu formalizovati korišćenjem jezika za modelovanje pravila.

17. U čemu se sastoji osnovna ideja primene nadgledanog mašinskog učenja za potrebe identifikacije entiteta u tekstu. Ukratko objasniti.

Odgovor:

Osnovna ideja primene **nadgledanog mašinskog učenja** za potrebe identifikacije entiteta u tekstu se sastoji od toga da:

- Program *uči karakteristike/osobine* koje odlikuju *entitete* određenog *tipa*
- *Osobine entiteta* se određuju na osnovu *termina* kojima su *entiteti* predstavljeni u tekstu, kao i *termin*a koji čine njihov *kontekst/okruženje*

18. Koji se atributi (features) obično koriste (kao sastavni deo modela nadgledanog m. učenja) za prepoznavanje entiteta u tekstu?

Odgovor:

Atributi (*features*) koje se obično koriste (kao sastavni deo modela nadgledanog m. učenja) za ***prepoznavanje entiteta u tekstu*** su:

1) **Atributi** koji se odnose na ***pojedinačne reči***: dužina reči, prisutnost velikih slova, vrsta reči, učestanost pojavljivanja reči u dok. za trening, prisutnost znakova interpunkcije, pozicija reči u rečenici itd.

2) **Atributi** koji se odnose na ***okruženje reči***: opseg okruženja, vrsta reči u okruženju i slično

19. Šta je glavna prepreka primeni metoda nadgledanog m. učenja za potrebe identifikacije entiteta u tekstu? Koji su alternativni pristupi i zašto?

Odgovor:

Glavna *prepreka* primeni **metoda nadgledanog m. učenja** za potrebe *identifikacije entiteta u tekstu* je to što je priprema dovoljno velikog *skupa anotiranih dokumenata* (*korpusa*) potrebnog za ***trening*** prilično zahtevan zadatak.

Alternativni pristupi su:

1. ***Polu-nadgledano*** i
2. ***Nenadgledano mašinsko učenje***

jer:

- ne zahtevaju anotirani skup dokumenata
- tradicionalno su imali slabije performanse u odnosu na pristupe nadgledanog m. učenja, ali su nova rešenja sve bolja

20. Šta je Bootstrapping? Ukratko objasniti kako funkcioniše u kontekstu prepoznavanja entiteta u tekstu.

Odgovor:

Bootstrapping je popularna tehnika polu-nadgledanog mašinskog učenja koja podrazumeva mali stepen "nadgledanja", tipično u formi inicijalno zadatog skupa primera, potrebnog za pokretanje procesa učenja.

On u kontekstu *prepoznavanja entiteta u tekstu* funkcioniše na sledeći način:

- inicijalno, korisnik zadaje mali broj primera tj. naziva različitih entiteta
- sistem kreće sa analizom teksta i pokušava da identifikuje elemente koji karakterišu kontekst zadatih primera, zatim, pokušava da identifikuje druga pojavljivanja entiteta na osnovu identifikovanih karakteristika konteksta
- proces učenja se ponovo primenjuje polazeći od novo-otkrivenih instanci (entiteta), što vodi otkrivanju novih relevantnih konteksta
- ponavljajući ovaj proces, veliki broj naziva entiteta i konteksta u kojima se ona pojavljuju će biti otkriven

21. U čemu se ogledaju prednosti pristupa koji kombinuju mašinsko učenje i baze znanja? Ukratko objasniti.

Odgovor:

Prednosti pristupa koji kombinuju ***mašinsko učenje*** i ***baze znanja*** se ogledaju u:

- pored prepoznavanja *tipa entiteta*, omogućuju i *jedinstveno identifikovanje entiteta* (*disambiguation*)
- jednostavnije kreiranje *skupa podataka* za *obučavanje algoritma*

22. Koji su osnovni koraci u procesu prepoznavanja entiteta u tekstu u slučaju pristupa koji kombinuju mašinsko učenje i baze znanja?

Odgovor:

Osnovni koraci u procesu ***prepoznavanja entiteta u tekstu*** u slučaju pristupa koji kombinuju ***mašinsko učenje*** i ***baze znanja*** su:

1) **Spotting** - identifikacija tzv. entity-spots tj. termina za koje se pretpostavlja da bi mogli predstavljati entitete u tekstu

2) **Candidate selection** - za svaki entity-spot, vrši se selekcija potencijalnih koncepata* iz baze znanja (npr. Wikipedia-e)

3) **Disambiguation** - izbor "najboljeg" koncepta za svaki entity-spot, tj. koncepta koji najbolje odražava semantiku datog termina u datom kontekstu

4) **Filtering** - filtriranje rezultata u cilju eliminacije irelevantnih entiteta

23. Šta je Wikipedia Miner i kakve servise obezbeđuje?

Odgovor:

Wikipedia Miner je alat koji služi za prepoznavanje entiteta u tekstu i implementira pristup zasnovan na mašinskom učenju i bazama znanja.

Servisi koje obezbeđuje su:

1. ***Wikify*** - identifikuje entitete/koncepte iz Wikipedia-e u zadatom tekstu
2. ***Compare*** - utvrđuje i objašnjava povezanost između dva Wikipedia koncepta
3. ***Suggest*** - predlaže teme/koncepte koji su semantički slični/srodni zadatim konceptima

24. Šta je TagMe i kakve servise obezbeđuje?

Odgovor:

TagMe je alat koji služi za prepoznavanje entiteta u tekstu i implementira pristup zasnovan na mašinskom učenju i bazama znanja.

Servisi koje obezbeđuje su:

1. ***Tagging*** - identifikuje entitete/koncepte iz Wikipedia-e u zadatom tekstu
2. ***Spotting*** - detektuje relevantne termine u tekstu (ne povezuje ih sa Wikipedia konceptima)
3. ***Relating*** - određuje semantičku povezanost dva zadata koncepta

Pitanja iz oblasti WEB, WEB PODATAKA i SEMANTIČKI WEB

1. Šta karakteriše fazu razvoja Web-a poznatu pod nazivom ”Web 2.0“?

Odgovor:

Fazu razvoja Web-a poznatu pod nazivom ”**Web 2.0**“ karakterišu:

1) Promene u *načinu* na koji ljudi koriste Web, a ***NE*** na novi tehnološki talas

2) Drugu generaciju Internet servisa fokusiranih primarno na *online kolaboraciju* i *deljenje sadržaja* među *korisnicima*

2. Šta označava termin “Web 3.0“? Ukratko objasniti.

Odgovor:

Termin “**Web 3.0**“ označava treću generaciju Internet servisa koji, kolektivno posmatrani, čine *Inteligentni Web*.

3. Šta karakteriše fazu razvoja Web-a poznatu pod nazivom ”Web 3.0“?

Odgovor:

Fazu razvoja Web-a poznatu pod nazivom ”**Web 3.0**“ karakterišu:

1) ***Web podatka*** (*Web of Data*)

2) ***Mobilni Web*** - portabilan, svestan lokacije korisnika, uvek pristutan, uvek dostupan

3) ***Primena tehnologija*** baziranih na ***Veštačkoj inteligenciji***: *Mašinsko učenje*, *Zaključivanje zasnovano na pravilima*, *Modelovanje korisnika i personalizacija*, *Razumevanje prirodnog jezika* i *Personalni agenti*

4. Navesti osnovne karakteristike današnjeg Web-a.

Odgovor:

Osnovne karakteristike današnjeg **Web**-a su:

- Dizajniran za direktno korišćenje od strane ljudi
- Primarni objekti su dokumenti i multi-medija
- Stepen strukturiranosti objekata je prilično nizak
- Linkovi su između dokumenata (ili njihovih delova)
- Semantika sadržaja i linkova je implicitna
- Analogija sa globalnim fajl sistemom

5. Navesti i ukratko obrazložiti izazove sa kojima se susreće današnji Web.

Odgovor:

Izazovi sa kojima se susreće današnji **Web** su:

Izazov 1: ***Integracija podataka*** - realizacija upita koji zahtevaju integrisanje podataka iz različitih izvora

Izazov 2: ***Razvoj naprednih servisa***

Izazov 3: ***Kreiranje adaptivnih RSS feeds*** - iz bilo kog izvora, filtrirani proizvoljnim skupom kriterijuma

6. Šta je to Web podataka (Web of Data)?

Odgovor:

Web podataka (*Web of Data*) je vizija Web-a kao jedne gigantske globalne *baze podataka*.

7. Navesti glavne karakteristike Web-a podataka (Web of Data).

Odgovor:

Glavne *karakteristike* **Web**-a **podataka** (*Web of Data*) su:

- Podaci (na Web-u) su strukturirani i interlinkovani
- Semantika podataka i linkova je eksplicitno data
- Omogućeno je izvršavanje složenih upita nad više izvora

8. Za koje tipove problema se preporučuje primena Web of Data tehnologija?

Odgovor:

Tipovi problema za koje se preporučuje primena **Web of Data** *tehnologija* su:

1. "***Open-ended problems***":
   - Model podataka nije konačan/precizno definisan
   - Slučajevi korišćenja (aplikacije) nisu konačni
   - Lista mogućih korisnika sistema nije konačna

2. Potreba za ***integrisanjem podataka*** iz ***različitih izvora*** korišćenjem *otvorenih standarda*
3. Rad sa ***nestrukturiranim sadržajima***
   - Dokumenti
   - Web stranice
   - Novinski članci
   - Stručni tekstovi itd.

9. Za koje tipove problema se NE preporučuje primena Web of Data tehnologija?

Odgovor:

Tipovi problema za koje se ***NE*** preporučuje primena **Web of Data** *tehnologija* su:

1. Rad sa *ogromnim* ***količinama podataka*** (> 100 miliona redova)
2. *Visoka* ***frekvencija transakcija*** (više hiljada transakcija u sekundi)
3. *Numeričke operacije* nad *ogromnom količinom* (terabajti) ***numeričkih podataka***

10. Navesti osnovne domene primene Web of Data tehnologija u organizacijama.

Odgovor:

Osnovne *domeni primene* **Web of Data** tehnologija *u organizacijama* su:

1) Agilna ***integracija podataka***
2) ***Anotacija***, ***klasifikacija***, ***pretraga informacija***
3) Dinamičko ***kreiranje sadržaja***

11. Šta je to Semantički Web? Ukratko objasniti.

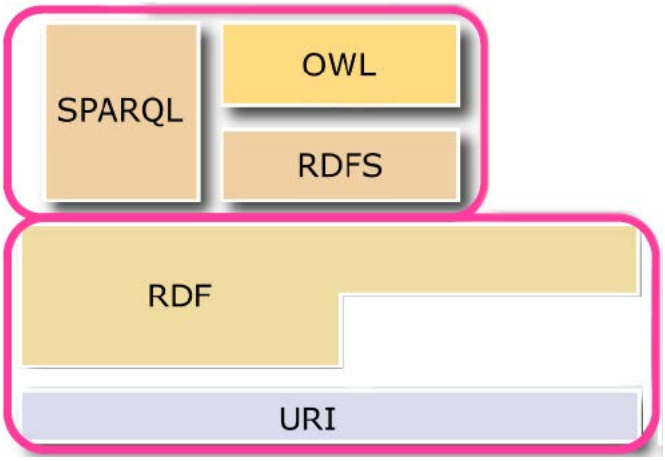
Odgovor:

Semantički Web predstavlja *naredni korak* u *evoluciji Web-a podataka*, tj. '***Inteligentni*** ***Web***:

- Rezonovanje nad podacima integrisanim iz različitih izvora
- Sistemi za preporuku na nivou čitavog Web-a
- Inteligentni agenti vrše pretragu i preporuku sadržaja

12. Na slici su predstavljeni "gradivni blokovi" (tj. tehnologije) Semantičkog Web-a. Složite ih tako da oni čine Semantic Web Layer Cake (tačnije jedan njen deo). Objasniti zbog čega su ovi blokovi upravo na ovaj način poređani u okviru Semantic Web Layer Cake-a.

Odgovor:
Na slici su predstavljeni "gradivni blokovi" (tehnologije) **Semantičkog Web**-a, složeni tako da čine ***Semantic Web Layer Cake*** (jedan njegov deo):



URI - je ***hypertext web*** *tehnologija* koja čini bazu **Semantičkog Web**-a [2]
RDF, **RDFS**, **OWL** i **SPARQL** - su *standardizovane tehnologije* (*W3C*) za razvoj aplikacija **Semantičkog Web**-a

Da bi se kompletno ostvarila vizija **Semantičkog Web**-a, potrebno je u potpunosti implementirati sve gore navedene tehnologije.

13. Šta je URI? Ukratko objasniti.

Odgovor:
URI (*Uniform Resource Identifier*) identifikuje stvari koje opisujemo i omogućava lako povezivanje podataka sa različitih izvora. Ako se na dva različita mesta kreiraju podaci koristeći *isti URI*, to znači da se govori o istoj stvari.

14. Šta je RDF? Ukratko objasniti.

Odgovor:
RDF je *W3C standard* za opis podataka na Web-u, odnosno jednostavan *model* (zasnovan na *grafu*) koji opisuje relacije između "stvari" (things).

[2] "Semantic Web Stack", Wikipedia, http://en.wikipedia.org/wiki/Semantic_Web_Stack (30.1.2014. 3:24)

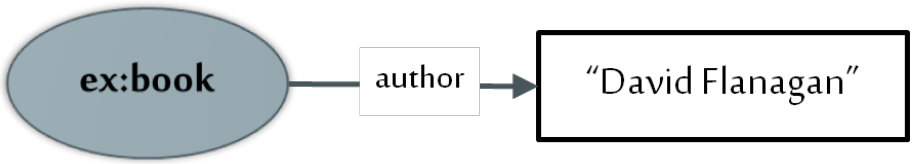
15. Napisati jedan (proizvoljan) RDF triplet i predstaviti ga: 1) grafički; 2) korišćenjem TURTLE sintakse.

Odgovor:

RDF je graf baziran na *tripletima* oblika:

(**subjekat** **predikat** **objekat**)

1) *Grafički prikaz* jednog (proizvoljnog) RDF tripleta:



2) ***TURTLE sintaksa*** gore navedenog RDF tripleta:

@prefix ex: <http://example.com> .
ex:book ex:author "David Flanagan" .

16. Šta je RDFS? Ukratko objasniti.

Odgovor:

RDFS (*RDF Schema*) je šema koja ima ulogu u dodavanju semantike u RDF i kreiranju vokabulara.

17. Navesti po čemu se koncept property-a u RDFS-u razlikuje od koncepta property-a u objektno-orijentisanim jezicima.

Odgovor:

Koncept ***property***-a u **RDFS**-u se razlikuje od koncepta ***property***-a u **objektno-orijentisanim jezicima** po tome što:

- ***Propertiji*** mogu imati svoju *hijerarhiju*
- Ne mogu se *overwrite*-ovati na nižem nivou hijerarhije

18. Korišćenjem RDFS-a može se definisati domen i opseg bilo kog property-a. Šta predstavlja domen, a šta opseg jednog property-a?

Odgovor:

Domen property-a - pokazuje na *klasu* (ili skup klasa) na koje se *relacija* može primeniti

Opseg property-a - predstavlja *klasu* (ili skup klasa) koje mogu predstavjati *vrednost relacije*

19. Ukoliko za neki property nisu definisani ni domen ni opseg, da li se i kako taj property može koristiti?

Odgovor:

I **domen** i **opseg** su opcioni.

Ukoliko ***domen*** nije definisan, *relacija* se može primeniti na bilo koju klasu.

Ukoliko ***opseg*** nije definisan, *vrednost relacije* može biti bilo koja klasa.

20. Šta je to Dublin Core i čemu je namenjen?

Odgovor:

Dublin Core je *rečnik metapodataka* koji nema klase već samo property-e, a namenjen je opisivanju dokumenata uz pomoć *skupa RDF elemenata*.

21. Šta je to FOAF i čemu je namenjen?

Odgovor:

FOAF (*Friend Of A Friend*) je *šema* bazirana na ***RDF***-u, čija je namena da:

- Opiše osnovne podatke o ljudima (ime, prezime, email adresa, homepage...)
- Poveže ljude koji se poznaju (knows)

Nema granice kao socijalne mreže.

22. Navesti pristupe za umetanje strukturiranih podataka u Web (HTML) stranice.

Odgovor:

Pristupi za umetanje strukturiranih podataka u ***Web*** (***HTML***) ***stranice*** su:

1. ***Meta tagovi***
2. ***Microformats***
3. ***RDFa***
4. ***Microdata***

23. Koje se prednosti mogu ostvariti umetanjem strukturiranih podataka u Web (HTML) stranice?

Odgovor:

Prednosti koje se mogu ostvariti *umetanjem strukturiranih podataka* (konkretno ***RDF***-a [3] i ***Microdata***) u ***Web*** (***HTML***) ***stranice*** su:

- efikasna implementacija
- struktuiranje informacija upotrebom grafa
- mogućnost obrade i u odsustvu detaljnih informacija (*RDF šeme*)
- modularnost
- kompaktna sintaksa omogućava lako kodiranje itd.

*** *NAPOMENA*: ne postoji eksplicitno naveden odgovor u materijalima, prednosti za *Microdata* potražiti na Internetu

[3] "The RDF Advantages Page", http://www.w3.org/RDF/advantages.html

24. Navesti i ukratko objasniti bar jedan nedostatak RDF Schema jezika koji je prevaziđen uvođenjem OWL ontološkog jezika.

Odgovor:

*Nedostaci **RDF Schema*** jezika koji su prevaziđeni uvođenjem **OWL ontološkog jezika** su:

1) Nije moguće definisati ***lokalizovana ograničenja*** *domena* i *opsega svojstava* - na primer, ne može se reći da je opseg svojstva *hasChild* osoba, jer to recimo važi i za životinje

2) Nema ***egzistencijalnih ograničenja***, niti ***ograničenja kardinalnosti*** - na primer, ne može se reći da sve osobe (tj. instance klase *Person*) imaju majku (tj. da mora postojati svojstvo *hasMother*) i da je ona takođe osoba, ili da svaka osoba ima tačno dva roditelja

3) Nema ***tranzitivnih***, ***inverznih*** ili ***simetričnih svojstava*** - na primer, ne može se reći da je *isPartOf* tranzitivno svojstvo, da je *hasPart* inverzno od *isPartOf*, ili da je *touches* simetrično svojstvo

25. Koje je značenje OWL ograničenja owl:allValuesFrom? Navesti jedan primer njegove primene.

Odgovor:

OWL ograničenje ***owl*:*allValuesFrom*** (naziva se i **Univerzalno ograničenje**) zahteva da u kontekstu klase na koju se primenjuje, sva pojavljivanja property-a sadrže kao svoju vrednost isključivo instance zadate klase.

Ovim se ne sprečava da instance klase na koju se ovo ograničenje primenjuje nemaju pridružen property datog tipa.

Primer primene: preporuka restorana na osnovu specifičnih preferenci u ishrani: osobi koja je vegeterijanac predložiti samo restorane sa širokom ponudom vegeterijanskih jela.

26. Koje je značenje OWL ograničenja owl:someValuesFrom? Navesti jedan primer njegove primene.

Odgovor:

OWL ograničenje ***owl:someValuesFrom*** (naziva se i **Egzistencijalno ograničenje**) zahteva da u kontekstu klase na koju se primenjuje, postoji bar jedno pojavljivanje property-a čija je vrednost instanca zadate klase.

Ovim se ne sprečava da instance klase na koju se ovo ograničenje primenjuje imaju veći broj property-a datog tipa sa nekim drugim vrednostima.

Primer primene: ograničenje izbora materijala kojim je nameštaj presvučen iz asortimana spavaćih soba, tako da bar jedan komad nameštaja pripada klasi kreveta

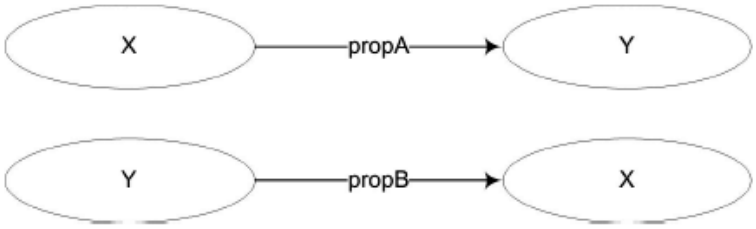
27. Navesti i ukratko objasniti tipove relacija (properties) koje OWL uvodi.

Odgovor:

Tipovi **relacija** (**properties**) koje **OWL** uvodi su:

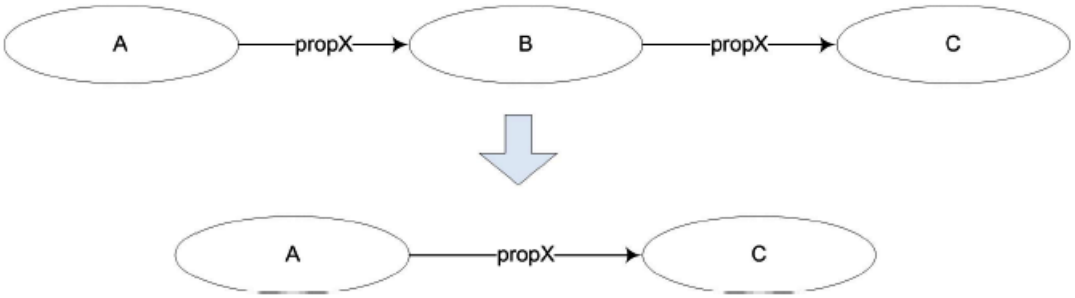
1. ***Funkcionalni property*** (*owl*:*FunctionalProperty*) - najviše jedno pojavljivanje property-a ovog tipa se može pridružiti resursima koji čine njegov domen

2. ***Inverzni property*** (*owl*:*inverseOf*) - ako su properties *propA* i *propB* definisani kao *inverzni*, to znači da je domen property-a *propA* predstavlja opseg property-a *propB* i obrnuto (opseg property-a *propA* mora biti domen property-a propB). To dalje znači da, subjekat tripleta u kome se property *propA* pojavljuje (u ulozi predikata) mora biti objekat tripleta u kome se property *propB* pojavljuje (kao predikat) i obrnuto.




```
<owl:ObjectProperty rdf:about="#propA">
    <owl:inverseOf rdf:resource="#propB"/>
</owl:ObjectProperty>
```

3. ***Tranzitivni property*** (*owl*:*TransitiveProperty*) - radi o klasičnom svojstvu tranzitivnosti (iz Matematike)



4. ***owl*:*disjoinedWith*** - je relacija, primitiva OWL jezik, kojom se kaže da neke dve *klase* (ili u opštem slučaju više klasa) nemaju zajedničke instance

5. ***Egzistencijalno ograničenje*** (*owl*:*someValuesFrom*) - opisano u odgovoru na pitanje ***26.*** iz ove oblasti

6. ***Univerzalno ograničenje*** (*owl*:*allValuesFrom*) - opisano u odgovoru na pitanje ***25.*** iz ove oblasti

28. Navesti osnovne principe na kojima se zasniva koncept linkovanih podataka (Linked Data).

Odgovor:

Osnovni *principi* na kojima se zasniva koncept **linkovanih podataka** (*Linked Data*) su:

1. Koristiti URI za jedinstvenu identifikaciju entiteta/objekata/pojava/…
2. Koristiti HTTP URI tako da se informacije o entitetima učine dostupnim posredstvom Web-a
3. Opisati entitete korisnim podacima primenom RDF modela. U te svrhe, preporučuje se korišćenje postojećih RDF vokabulara
4. Uspostaviti imenovane linkove ka drugim entitetima/objektima/pojavama…

29. Šta je to Linked Open Data Cloud? Ukratko objasniti.

Odgovor:

Linked Open Data Cloud predstavlja dataset-ove koji su objavljeni u *Linked Data* formatu [4] , koji opisuje metodu objavljivanja strukturiranih podataka, kako bi mogli da se međusobno povežu i postanu korisniji [5] .

30. Šta je to DBPedia? Ukratko objasniti.

Odgovor:

DBPedia je *baza znanja* strukturiranih podataka koji su estrahovani iz Wikipedije.

31. Šta je to GeoNames? Ukratko objasniti.

Odgovor:

GeoNames je otvorena, globalna *geografska baza znanja* (*Wikipedia* za geografske podatke/znanje)

Omogućuje odgovore gde se nalazi neko mesto, koje su njegove koordinate, kojoj regiji ili provinciji mesto pripada, koji grad ili adresa su u blizini zadate geografske širine i dužine itd.

32. Koje uslove moraju da ispunjavaju otvoreni podaci da bi bili ocenjeni sa 5 zvezdica u okviru Linked Open Data star scheme?

Odgovor:

Uslovi koje moraju da ispunjavaju *otvoreni podaci* da bi bili ocenjeni sa 5 zvezdica u okviru **Linked Open Data star scheme** su [6] :

1. Omogućiti da podaci budu dostupni na Web-u (u bilo kom formatu)
2. Omogućiti da budu dostupni kao strukturirani podaci
3. Omogućiti da budu u nezaštićenom (*open*) formatu
4. Koristiti *URL*-ove za identifikaciju, kako bi mogli da usmeravaju ka tim podacima
5. Povezati podatke sa podacima ostalih korisnika, u odgovarajućem kontekstu

[4] "The Linking Open Data cloud diagram", http://lod-cloud.net/
[5] "Linked data", http://en.wikipedia.org/wiki/Linked_data
[6] "5 star Open Data", http://5stardata.info/

33. SPARQL se sastoji iz više specifikacija, navesti o kojim specifikacijama je reč (navesti bar 3).

Odgovor:

SPARQL se sastoji iz više *specifikacija*:

1. Specifikaciju ***upitnog jezika***
2. Specifikaciju ***jezika*** za ***modifikaciju RDF grafa***
3. Specifikaciju ***rezultata upita***
4. Specifikaciju ***protokola*** za ***pristup podacima***
5. Specifikaciju ***federativnih upita*** itd.

34. Čemu je namenjen SPARQL ASK upit i kakav tip rezultata vraća?

Odgovor:

*SPARQL **ASK** upit* je namenjen za proveru da li neki upit uopšte ima rešenje i ne vraća nikakvu informaciju o samom rešenju upita, već samo da li ono postoji. Tip rezultata koji vraća je **boolean** (*true*/*false*).

35. Čemu je namenjen SPARQL DESCRIBE upit i kakav tip rezultata vraća?

Odgovor:

*SPARQL **DESCRIBE*** upit vraća graf koji sadrži sve raspoložive triplete o resursu koji je mečiran u okviru *graf paterna* (tj. u *WHERE* delu upita).

36. Čemu je namenjen SPARQL CONSTRUCT upit i kog oblika je rezultat ovog upita?

Odgovor:

*SPARQL **CONSTRUCT** upit* je namenjen za kreiranje novih RDF grafova na osnovu postojećih tj. za *transformaciju RDF grafova*. Rezultat ovog upita je u obliku ***tripleta*** [7] .

[7] "SPARQL CONSTRUCT in AllegroGraph - Parts, stores, and CONSTRUCT", http://franz.com/agraph/support/documentation/v4/sparql-construct.html