Dubinska analiza podataka **Ak. god. 2023./2024.**

Ogledni primjer završnog ispita (maks. 40 bodova)

IME I PREZIME: _____ JMBAG:_____

1. (2 boda) Navedite dvije razlike između modela CRISP-ML(Q) i CRISP-DM.

   Rješenje:
   Kod modela CRISP-ML(Q), svaki korak ima kontrolu kvalitete, praćenje rizika, što CRISP-DM-u nedostaje.
   Također, kod CRISP-ML(Q) radi se nadzor i održavanje modela, a kod CRISP-DM-a ne.

2. (2 boda) Koji su ključni koraci (njih tri) koji su zajednički svim modelima procesa pripreme podataka?

   Rješenje:
   Otkrivanje podataka.
   Karakterizacija podataka.
   Izgradnja skupa podataka za modeliranje.

3. (2 boda) Pojasnite što treba napraviti s otkrivenim stršećim podatkom.

   Rješenje:
   Utvrditi je li prirodan ili ne (moguća konzultacija sa stručnjakom).
   Ako je prirodan ali se zna da će smetati prilikom izgradnje modela koristiti normalizaciju vrijednosti varijabli.
   Ako nije prirodan tretirati ga kao nedostajući podatak i potom primijeniti neki od postupaka za nedostajuće podatke.
   Zapamtiti ga (i njegovu poziciju u skupu) radi otkrivanja razloga unosa pogreške.

4. (2 boda) Navedite dva načina na koji se mogu otkriti statistički redundantne značajke. Objasnite kako se koristi korelacijska matrica.

   Rješenje:
   Statistički redundantne značajke određuju se korelacijskom analizom i Markovljevim pokrivačem. Ako je u korelacijskoj matrici vrijednost korelacije između dviju značajki vrlo visoka, idealno 1, odabire se jedna od značajki za uklanjanje.

5. (2 boda) Navedite i objasnite dvije vrste pretvorbi kategoričkih varijabli u numeričke.

   Rješenje:
   Izravna pretvorba kategoričke varijable u numeričku (engl. *label encoding, integer encoding*)

kategorija1 --> 1 ; kategorija2 --> 2 .... kategorija n --> n samo u slučaju kada poredak kategorija ima smisla.
Pretvorba gdje svaka kategorija neke kategoričke varijable postaje nova binarna varijabla (engl. *one-hot encoding*)
Od n kategorija dobivamo n binarnih značajki, koje imaju vrijednost 1 za one primjere za koje bi dotična kategorija vrijedila, a 0 inače.

6. (2 boda) Navedite najmanje tri primjera filterskih postupaka za odabir pojedinačnih značajki.

Rješenje:
Informacijska dobit, Simetrična nesigurnost, Relief, Hi-kvadrat, Fisherov skor.

7. (2 boda) Objasnite što su to ugrađeni postupci (engl. *embedded methods*) odabira značajki. Navedite barem jedan primjer takvih postupaka.

Rješenje:
Izbor značajki koji se temelji na nekom algoritmu strojnog učenja. Unutarnja struktura izgrađenog modela oslikava važnost značajki, bilo zbog broja pojavljivanja određene značajke u modelu ili njezine težine (značaja) u modelu.
Primjer: slučajna šuma, logistička regresija s penalizacijom (LASSO, elastic net), SVM.

8. (2 boda) Pojasnite naduzorkovanje (engl. *oversampling*) i zašto se koristi. Navedite barem dva načina kako se može koristiti naduzorkovanje.

Rješenje:
Generiranje novih primjeraka za učenje za manjinsku klasu (ili klase) kako bi se postigla ravnoteža s većinskom klasom.
Kopiranjem postojećih primjeraka *n* puta (najlošiji izbor).
Slučajnim naduzorkovanjem s ponavljanjem postojećih primjeraka (engl. *random oversampling with replacement*).
Kao izmijenjeni, sintetski primjerci.

9. (2 boda) Što koriste detektori pomaka u podacima zasnovani na monitoriranju? Kako se prati smanjenje točnosti modela?

Rješenje:
Detektori pomaka zasnovani na monitoriranju koriste statističke testove da prate je li došlo do promjene u razdiobi ciljne značajke na temelju performanse pogreške modela.
Smanjenje točnosti je podijeljeno u dvije razine: upozorenje i pomak koncepta, svaki sa svojim pragom.

10. (2 boda) Objasnite kako algoritam AdaBoost gradi model strojnog učenja i kako se donosi odluka.

Rješenje:

AdaBoost koristi panj odluke (engl. decision stump) koji se sastoji se od jednog čvora (korijena panja) i dva lista s primjercima koji se dobivaju grananjem u panju po nekoj značajci (koristeći npr. informacijski dobitak kao mjeru).
Tijekom iteracija AdaBoost koristi sve veću šumu panjeva da donese ispravne odluke.
Nakon K koraka iteriranja izgrađeni model koristi se zajednički pri donošenju odluke.

11. (2 boda) Objasnite što uključuje područje strojnog učenja s jasnim tumačenjem (interpretacijom) (engl. *interpretable machine learning*). S kojim sličnim područjem ga ne treba miješati?

Rješenje:
Uključuje modele strojnog učenja čije tumačenje (interpretacija) je jasno razumljivo čovjeku (engl. white box models).
Ne miješati s objašnjavanjem black box modela strojnog i dubokog učenja u okviru tzv. objašnjive umjetne inteligencije (engl. eXplainable AI , kraće: XAI).

12. (2 boda) Do čega dovodi konstrukcija skupa pravila koji je kompletan i konzistentan? Ako s *P* označimo ukupan broj pozitivnih primjeraka (primjeraka klase +), s *N* ukupan broj negativnih primjeraka (primjeraka klase -), s *p* pozitivne primjerke koje prekriva pravilo, a s *n* negativne primjerke koje prekriva pravilo, kako je definirano pokrivanje (ili potpora) pravila (engl. *coverage*, *support*)?

Rješenje:
Dovodi do prenaučenosti i lošijih rezultata na skupu za testiranje.
COV = SUP = (p + n) / P + N

13. (2 boda) Definirajte što je to itemset. Kako označavamo veličinu itemseta? Navedite nekoliko primjera itemsetova.

Rješenje:
Skup od jednog ili više artikala (od ukupnog broja artikala).
Označavamo ga s k-itemset, dakle 1-itemset, 2-itemset.
Primjeri: {kava}, {kava,kolač}, {kava,krafna,kolač}

14. (2 boda) Ukratko objasnite izgradnju temeljne strukture za otkrivanje čestih itemsetova korištenjem algoritma FP-growth. Koje su prednosti algoritma FP-growth u odnosu na Apriori?

Rješenje:
FP-growth je algoritam koji koristi stablastu strukturu podataka FP Tree za sažeti prikaz ulaznih podataka i pronalaženje čestih itemsetova.
FP tree se konstruira čitanjem jedne po jedne transakcije i bilježenjem svake transakcije u nekoj grani stabla.
Budući da transakcije često imaju neke zajedničke artikle, njihovi putovi u stablu se dijelom preklapaju i time se ostvaruje sažimanje velike količine podataka.
FP growth može biti i nekoliko redova veličine brži od Apriorija, ovisno o skupu podataka.

15. (2 boda) Objasnite važnost podjele u prozore tijekom predobrade vremenskog niza. Kako se sve ona može provesti?

Rješenje:
Smanjenje broja točaka u prozoru (k<<N) smanjuje računske zahtjeve i povećava preciznost izvedbe zadataka. Podjela u prozore može biti ovisna o periodičnosti (npr. prozor odgovara širini perioda p) i drugim značajkama niza (npr. prozor od k /2 točaka s obje strane nekog detektiranog oblika) ili neovisna o značajkama niza.
Podjela u prozore može biti s preklapanjem prozora (engl. window overlapping) ili bez preklapanja.

16. (2 boda) Objasnite četiri komponente generaliziranog aditivnog predikcijskog modela za vremenske nizove PROPHET (g, s, h, ε) .

Rješenje:
g (t) (engl. *growth*) je po dijelovima linearna ili logistička funkcija za modeliranje nesezonalnih promjena u nizu (trend).
s (t) (engl. *seasonality*) je funkcija periodičkih promjena (sezonalnosti) u modelu.
h (t) (engl. holidays) je funkcija utjecaja praznika tijekom jednog ili nekoliko dana, s nepravilnim razmacima, koju zadaje korisnik.
ε_t je pogreška (Gaussova razdioba) koja modelira one promjene koje nisu obuhvaćene ostalim komponentama.

17. (2 boda) Objasnite čemu služi funkcija (sloj) gubitka (engl. *loss function*) i koristi li se tijekom testiranja neuronske mreže? Navedite dva primjera funkcije gubitka (nije potrebno navoditi formule).

Rješenje:
Funkcija gubitka ili sloj gubitka završni je sloj neuronske mreže koji specificira kako učenje penalizira razliku između predviđenog izlaza iz mreže (dobivenog aktivacijskom funkcijom) tijekom procesa nadziranog učenja i stvarnog izlaza.
Na temelju funkcije gubitka L računaju se gradijenti koji se koriste u algoritmu učenja mreže za reviziju vrijednosti težina među neuronima. Ne koristi se tijekom testiranja.
Primjeri: kategorijska unakrsna entropija, srednja kvadratna pogreška, fokalni gubitak.

18. (3 boda) Opišite strukturu varijacijskog autoenkodera. Koji je zadatak koderskog sloja a koji dekoderskog? Kako se generiraju novi primjerci? Kako se funkcija gubitka regularizira?

Rješenje:
Sastoji se od kodera, latentnog sloja i dekodera. Koder preslikava ulaze u lokacije u latentnom prostoru, koji je opisan sa srednjom vrijednosti μ i standardnom devijacijom σ normalne razdiobe koja se uči iz podataka, ne postoji sloj koda.
Faktor šuma koristi se za generiranje primjeraka vrijednosti vektora z koji služi za rekonstrukciju.
Funkcija gubitka se regularizira KL divergencijom između latentnog vektora generiranog iz naučene Gaussove latentne razdiobe i apriorne latentne razdiobe (normalne).

19. (1 bod) Konvolucijske mreže s mogućnosti deformacije jezgre adaptivno mijenjaju mjesta u receptivnom polju na koja djeluju. Što se uči tijekom procesa učenja ovakve mreže?

Rješenje:
Uči se skup pomaka (engl. *offsets*) za svaku prostornu lokaciju unutar jezgre.

20. (2 boda) Navedite najmanje dvije značajke putem kojih jezični model RoBERTa nadograđuje jednostavniji model BERT.

Rješenje:
Veći skup podataka korišten za učenje (sa 16 GB na 160 GB) BookCorpus, English Wikipedia, CC News, OpenWebText, Stories.
Povećanje veličine slučajnog podskupa podataka (*batch size*) s 256 na čak 8000.
Učenje na duljim sekvencama podataka do 512 tokena u jednom podatku (primjerku za učenje), što čini jednu ili više kontinuiranih rečenica, BERT češće s manje tokena.
Korištenje većeg broja različitih tokena ispod razine riječi (engl. sub word units), od 30k tokena na 50k.
Dinamična promjena parametra maskiranja primijenjena tijekom učenja, za razliku od prije početka učenja kod BERT-a.