

## Zadatak 1. - Klasifikacija teksta

Napomena: zadatak je potrebno rešiti korišćenjem Weka Java API-a

Dat je skup podataka za obučavanje modela za klasifikaciju teksta (*kdd\_SyskillWeber-Bands.arff*). Podaci se odnose na ocene korisnika o muzičarima i muzičkim grupama. Konkretno, svaka instanca u datom skupu podataka opisana je sledećim atributima:

- *id* – atribut tipa String; jedinstveni identifikator instance (u okviru datog skupa podataka);
- *text* – atribut tipa String; kompletan tekst Web stranice koja sadrži komentare/ocene nekog muzičara/muz. grupe;
- *review\_type* – nominalni atribut sa 3 moguće vrednosti: *cold*, *hot*, *medium*; vrednosti ovog atributa predstavljaju moguće klase za klasifikaciju teksta.

Potrebno je uraditi sledeće:

- a) S obzirom da atribut *id* nije od značaja za klasifikaciju, ne treba ga koristiti za kreiranje klasifikatora, tj. potrebno je eliminisati ga iz skupa podataka koji se koriste za kreiranje klasifikacionog modela.
- b) Kreirati i evaluirati Naïve Bayes (NB) klasifikator koristeći dati skup podataka za trening. Ispisati (u konzoli) dobijene rezultate i protumačiti ih; konkretno, potrebno je objasniti značenje sledećih evaluacionih metrika u kontekstu datog zadatka: matrica zabune (*confusion matrix*), preciznost (*precision*), odziv (*recall*), F-mera (*F-measure*), ROC AUC; objašnjenje zapisati u posebnom txt dokumentu.
- c) Primeniti kreirani NB klasifikator za klasifikaciju teksta za koji klasa nije poznata. Tekst koji je potrebno klasifikovati dat je u fajlu “*sample\_review.txt*”; učitati tekst iz fajla i iskoristiti ga za kreiranje instance koja će biti prosleđena NB klasifikatoru na klasifikaciju. Ispisati (u konzoli) rezultate klasifikacije.

## **Zadatak 2. - Klasifikacija: određivanje tipa vozila na osnovu datog skupa atributa**

Napomena: zadatak je potrebno rešiti korišćenjem Weka Java API-a

Dat je skup podataka (*vehicle.arff*) u kome je svaka instanca opisana preko niza numeričkih atributa (ukupno 18 atributa) i klasifikovana u jednu od 4 mogućih klasa tj. tipova vozila: *opel*, *saab*, *bus*, *van*.

Potrebljeno je uraditi sledeće:

- a) Kreirati Naïve Bayes klasifikator na osnovu datog skupa podataka. S obzirom da su atributi numerički, potrebno ih je diskretizovati. Diskretizaciju uraditi primenom odgovarajućeg Weka filtera i to tako da se ukupan opseg vrednosti svakog atributa podeli na 10 intervala, i da svaki interval ima približno jednak broj instanci.
- b) Evaluirati klasifikator koristeći dati skup podataka i kros validaciju sa 10 iteracija (10-fold cross-validation). Ispisati (u konzoli) dobijene rezultate i protumačiti ih; konkretno, potrebno je objasniti značenje sledećih evaluacionih metrika u kontekstu datog zadatka: matrica zabune (*confusion matrix*), preciznost (*precision*), odziv (*recall*), F-mera (*F-measure*), ROC AUC; objašnjenje zapisati u posebnom txt dokumentu.
- c) Kreirati novi Naïve Bayes klasifikator koji ne koristi sve attribute iz datog skupa podataka, već samo one za koje se proceni da su relevantni za dati zadatak klasifikacije. Selekciju relevantnih atributa uraditi primenom *AttributeSelection* Weka filtera. Za kreiranje ovog filtera, možete koristiti klase *ClassifierSubsetEval* i *BestFirst* za parametre *evaluator* i *search*, respektivno. Ispisati (u konzoli) performanse klasifikatora dobijene primenom kros validaciju sa 10 iteracija, i prokomentarisati dobijene rezultate u kontekstu inicijalno dobijenih rezultata (jesu li bolji ili lošiji).
- d) Učitati fajl “*unknownVehicle.arff*” koji sadrži podatke za jedno vozilo čija je klasa nepoznata. Odrediti klasu vozila primenom kreiranog klasifikatora. Ispisati (u konzoli) rezultate klasifikacije.

### **Zadatak 3. - Klasterizacija: grupisanje studenata na osnovu njihovih odgovora**

Napomena: zadatak je potrebno rešiti korišćenjem Weka Java API-a

Dat je skup podataka (student-evaluation-answers.arff) u kome se nalaze odgovori studenata na 28 postavljenih pitanja. Atributi kojima su opisani odgovori su:

- *no* – redni broj odgovora
- *instr* – id instruktora, moguće vrednosti su 1, 2, 3
- *class* – id grupe kojem studenti pripadaju, moguće vrednost su od 1-13
- *nb.repeat* – označava po koji put student sluša predmet
- *attendance* – broj prisustva na času
- *difficulty* – težina predmeta, vrednosti na skali od 1-5
- *q1* - *q28* – odgovori na 28 pitanja, vrednosti na skali od 1-5

Potrebno je uraditi sledeće:

- a) Uraditi K-Means klasterizaciju na osnovu datog skupa podataka. Prilikom klasterizacije, ne treba uzimati u obzir atribute *no* i *instr*. Ispisati u konzoli rezultate klasterovanja.
- b) Utvrditi koji je optimalan broj klastera i u posebnom tekstualnom dokumentu dati obrazloženje zbog čega je odabran baš taj broj klastera.
- c) Napisati u tekstualnom fajlu koliko instanci pripada kojem klasteru za optimalan broj klastera. Takođe, napisati koliko iznosi greška *Within cluster sum of squared errors* i opisati šta označava ta mera i kako se računa.