SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1475

KLASIFIKACIJA TEKSTA POMOĆU STABLA ODLUKE

Zvonimir Szörsén

Zagreb, rujan 2004.

Sadržaj

1 Uvod

Klasifikacija teksta je proces koji automatski klasificira tekst u predodređeni broj kategorija služeći se samo njegovim sadržajem.
U posljednje vrijeme ovo područje doživljava veliki interes, a primarni razlog treba tražiti u sve većoj količini dokumenata u digitalnom obliku i potrebi za njihovom organizacijom. Povećanju dokumenata u digitalnom obliku zasigurno je najviše doprinio razvoj Interneta i Web stranica. Eksponencijalni rast Interneta donio je i stvarnu potrebu za korištenjem metoda za klasifikaciju teksta. Nekada je bilo moguće kategorizirati Web stranice korištenjem ljudi kao radne snage, no uz prije navedeni rast Interneta i dostupnih dokumenata, korištenje ljudi bilo bi neučinkovito i upravo ovdje na scenu dolaze alati za klasifikaciju teksta.
Još jedna vrlo bitna primjena također je vezana uz Internet, odnosno e-mail uslugu. Svakodnevno svaki korisnik e-maila u svojem sandučiću pronalazi mnoštvo reklamnih poruka odnosno spam. Metode klasifikacije teksta ovdje su se pokazale kao vrlo kvalitetan automatski filtar.

Zadatak ovog diplomskog rada je izgradnja programskog sustava za klasifikaciju teksta korištenjem C4.5 algoritma. Izgrađeni sustav bit će testiran na standardnom Reuters-21578 skupu, te na Vjesnik skupu.

U nastavku slijede teoretska objašnjenja funkcioniranja korištenih metoda, opis korištenih skupova, opis implementacije aplikacije te prikaz dobivenih rezultata.

2 Klasifikacija teksta

2.1 Definicija

Kategorizaciju teksta možemo prikazati kao proces dodjeljivanja boolean vrijednosti svakom paru $\langle d_j, c_i \rangle \in D \times C$, gdje je D domena dokumenata, a $C = \{c_1, ..., c_{|C|}\}$ skup unaprijed određenih kategorija. Vrijednost T dodijeljena paru $\langle d_j, c_i \rangle$ označava odluku da se dokument d_j klasificira kao c_i, dok F označava da se ne klasificira kao c_i.

Formalno klasifikaciju teksta možemo prikazati kao proces aproksimacije nepoznate ciljne funkcije $\breve{\Phi} : D \times C \rightarrow \{T, F\}$, koja opisuje kako se dokumenti trebaju klasificirati ,pomoću funkcije $\Phi : D \times C \rightarrow \{T, F\}$ tako da se te dvije funkcije što više slažu.

2.2 Reprezentacija teksta

Prije primjene bilo koje metode za klasifikaciju teksta, dokumente nad kojima se klasifikacija izvršava potrebno je prikazati u prikladnom obliku. Najčešće korištena metoda je vektorska reprezentacija pojedinih riječi. U tom slučaju elementi vektora sadrže frekvencije pojavljivanja pojedinih riječi. Takva reprezentacija obično se naziva vreća riječi (bag of words).

Bitno je primijetiti da se riječi uzimaju bez pravila (redoslijeda u rečenici) i bez pripadnosti bilo kakvoj strukturi dokumenta.

Prilikom korištenje dokumenata na engleskom jeziku, obično se riječima uklanjaju sufiksi koji se pojavljuju korištenjem gramatičkih pravila (eng. word stemming).

Kako dokumenti sadrže mnoštvo riječi, jasno je da će neke riječi imati puno veću klasifikacijsku vrijednost od drugih.

Vrlo visoka klasifikacijska vrijednost riječi znači da je riječ u uskoj vezi sa kategorijom dokumenta.

Riječi koje imaju vrlo malu klasifikacijsku vrijednost nazivamo šum i poželjno ih je eliminirati iz vektora. Da bi se takve riječi eliminirale, obično se koriste liste čestih, odnosno neinformativnih riječi u jeziku (eng. stop-word list). Obično se radi o veznicima, prilozima, prijedlozima i drugim riječima koje se često upotrebljavaju u svim vrstama dokumenata.

Naravno, ista riječ će u nekim kategorijama imati visoku klasifikacijsku vrijednost, dok će u drugima predstavljati šum.

2.3 Izbor značajki (feature selection)

Dimenzionalnost prostora značajki kod klasifikacije teksta obično je određena brojem unikatnih riječi u skupu dokumenata za učenje.
Obično je riječ o vrlo velikim brojkama, koje, ovisno o veličini skupa za učenje, često prelaze veličinu od nekoliko desetaka tisuća.
Proces smanjivanja dimenzionalnosti, odnosno izbacivanja riječi naziva se izbor značajki.
Primjena ovog procesa obično je uvjetovana efikasnošću, odnosno vremenskom i prostornom kompleksnošću metode koju koristimo, a koja gotovo uvijek ovisi o dimenzionalnosti prostora značajki.
Još jedan, vrlo bitan, razlog za primjenu ovog procesa je zaštita od overfittinga. Overfitting se javlja kada se stvaraju stabla koja točno klasificiraju sve primjere iz skupa za učenje, dok im je klasifikacija testnih primjera vrlo loša.
Glavni problem koji se javlja kod izbora značajki je kako izabrati prave riječi, odnosno koje riječi doprinose točnosti klasifikacije i poželjno ih je zadržati, a koje imaju vrlo malu klasifikacijsku vrijednost i poželjno ih je izbaciti.
Razvijeno je nekoliko metoda za izbor značajki, detaljnije o pojedinoj metodi i učinku na točnost klasifikacije može se naći u [5], a u nastavku će biti prikazane dvije metode: document frequency i information gain.

2.3.1 Document Frequency (DF)

Frekvencija pojavljivanja riječi u dokumentima vrlo je jednostavna i učinkovita metoda.
Kod ove metode, za svaku riječ računa se broj dokumenata u kojima se ona pojavljuje, te se izbacuju one riječi koje se pojavljuju u manje od izabranog broja dokumenata.
Osnovna pretpostavka kod ove metoda je da riječi koje se pojavljuju u malom broju dokumenata imaju vrlo malu informacijsku vrijednost za predviđanje kategorije.
Osim samog smanjenja dimenzija prostora značajki, u određenim slučajevima kada rijetke riječi predstavljaju šum, možemo očekivati i poboljšanje točnosti klasifikacije.

Pomoću ove metode dimenzionalnost je moguće smanjiti i do 10 puta bez gubitka točnosti [5], a do 100 puta uz vrlo malen gubitak preciznosti.

2.3.2 Informacijska dobit (IG)

Informacijska dobit često je korištena metoda u području strojnog učenja, te jedan od temeljnih načina konstrukcije stabla odluke.
Ova metoda mjeri broj bita informacije dobivene za predikciju kategorije poznavajući prisutnost određene riječi u dokumentu.
Ako sa $\{c_i\}_{i=1,\dots,m}$ označimo skup kategorija, informacijska dobit riječi t može se definirati na slijedeći način:

$$\begin{aligned} G(t) = &-\sum_{i=1}^{m} P(c_i)\log(P(c_i)) \\ &+P(t)\sum_{i=1}^{m} P(c_i|t)\log(P(c_i|t)) \\ &+P(\bar{t})\sum_{i=1}^{m} P(c_i|\bar{t})\log(P(c_i|\bar{t})) \end{aligned}$$

Nakon što je za svaku riječ izračunata informacijska dobit, selekcija se radi tako da se izaberu one riječi čija informacijska dobit je veća od određene granice.

2.4 Stabla odluke i C4.5

Stabla odluke najčešće su korištena metoda induktivnog zaključivanja, robusna na šum i sposobna učiti disjunktivne koncepte.
Jedan od razloga popularnosti i privlačnosti ove metode je prikaz klasifikacijskih odluka. Za razliku od drugih klasifikacijskih metoda, kao što su primjerice naivni Bayesov klasifikator ili neuronske mreže, koje stvaraju numeričke klasifikacijske odluke, stabla odluke stvaraju odluke koje ljudi mogu vrlo jednostavno interpretirati. Sama čitljivost stabala ima smisla kada su stabla mala, no kod klasifikacije teksta, stvaraju se velika stabla, te je od čitljivosti puno bitnija komponenta točnost klasifikacije.
Postoji čitavi niz algoritama za konstruiranje stabla odluke, no od slobodno dostupnih najpopularniji i najviše korišteni je C4.5 [1] koji je korišten i u ovom diplomskom radu.

2.4.1 Predstavljanje stabla odluke

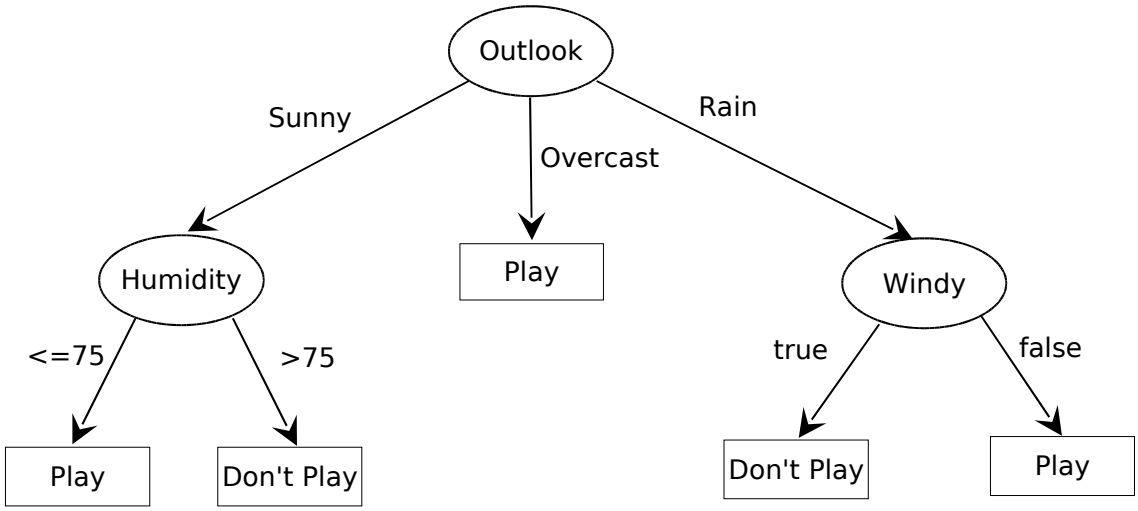
Prije samog opisa strukture stabala odluke, potrebno je pogledati koje zahtjeve sama metoda nameće, odnosno koje su bitne karakteristike problema pogodnih za oblikovanje stablima odluke:

- Primjeri su predstavljeni parovima atribut - vrijednost.
  Svaki objekt odnosno primjer iz skupa za učenje i testiranje, mora biti prikazan fiksnim skupom atributa i vrijednošću vezanom za atribut. Vrijednosti mogu biti diskretne ili numeričke.
- Kategorije su unaprijed određene.
  Kategorije unutar kojih će se klasificirati primjeri moraju biti unaprijed određeni, odnosno riječ je o učenju uz učitelja.
- Kategorije su diskretne.
  Kategorije moraju biti strogo podijeljene. Ako određeni primjer pripada jednoj kategoriji, onda ne pripada niti jednoj drugoj. Isto tako, broj primjera mora biti puno veći nego broj kategorija.
- Dovoljan broj podataka za učenje.
  Kako se kod stabla odluke model izgrađuje induktivno, odnosno generalizacijom određenih primjera, potrebno je imati dovoljno podataka kako bi se unutar njih mogli identificirati uzorci.

Svako stablo odluke sastoji se od čvorova, grana i listova. Čvorovi su označeni sa pojedinim atributima i u njima se vrši testiranje atributa. Grane izlaze iz čvorova i svaka predstavlja jednu moguću vrijednost atributa iz kojeg izlazi.

Sve grane završavaju u listovima koji označavaju pripadnost određenoj kategoriji.
Proces klasifikacije kreće od korijena stabla, te se nastavlja kretati sve dok ne dođe do lista. Kod nailaska na čvor određuje se rezultat testiranja atributa, te proces nastavlja put preko grane čija vrijednost odgovara rezultatu testiranja. Ponavljanjem ovog procesa, dolazi se do lista, te je pretpostavljene kategorija testnog primjera ona koja je označena na listu.

Grafički prikaz vrlo jednostavnog stabla odluke, koje određuje da li je dan pogodan za igranje golfa, dan je na slijedećoj slici:



Slika 2.1 - Primjer jednostavnog stabla odluke koje odlučuje da li je dan pogodan za igranje golfa.

Stablo na slici možemo interpretirati na slijedeći način:

Ako je Outlook overcast , onda je dan pogodan za igranje.
Ako je Outlook rain i Windy je true, onda dan nije pogodan za igranje.

Na sličan način možemo interpretirati i ostale grane u stablu.

Općenito, stabla odluke predstavljaju disjunkciju konjunkcije uvjeta na vrijednosti atributa. Svaki put od korijena stabla prema listu predstavlja konjunkciju vrijednosti atributa, a samo stablo odgovara disjunkciji tih konjunkcija. Da li je dan pogodan za igranje golfa možemo prikazati na slijedeći način:

(Outlook = sunny **I** Humidity <= 75)
ILI (Outlook = overcast)
ILI (Outlook = rain **I** Windy = false)

2.4.2 Konstrukcija stabla odluke

Ideja koja se nalazi iza konstrukcije stabla odluke na osnovu primjera za učenje u svojoj biti je vrlo jednostavna.
Označimo sa T skup primjera za učenje, a klase označimo sa C1 do Ck. Sada imamo dvije moguće situacije, od kojih je prva trivijalna:

- Skup za učenje T sadrži primjere koji pripadaju samo jednoj klasi.
  Kreira se stablo odluke sa samo jednim listom označenim sa kategorijom Ci kojoj pripadaju svi primjeri.
- Skup za učenje T sadrži primjere koji pripadaju u više klasa.
  U ovom slučaju princip konstrukcije možemo prikazati slijedećim algoritmom:
  1. bira se atribut koji najbolje klasificira primjere, on postaje čvor, a njegove vrijednosti silazne grane
  2. primjeri za učenje raspodjeljuju se prema odgovarajućoj silaznoj grani, odnosno svakoj grani se dodjeljuju svi primjeri koji imaju vrijednost atributa u čvoru jednak vrijednosti u grani
  3. cijeli postupak se ponavlja korištenjem primjera dodijeljenih svakoj silaznoj grani

Atribut koji najbolje klasificira primjere

Navedena metoda za konstrukciju stabla odluke spada u pohlepne (greedy) algoritme. To znači da kada se izabere atribut koji postaje čvor, on ostaje fiksiran i algoritam se nikada ne vraća na njega radi ponovnog razmatranja.
Ta činjenica samo naglašava najvažniji izbor prilikom izgradnje stabla, a to je izbor atributa koji će se testirati u pojedinom čvoru.
Izbor najboljeg atributa moguć je samo kada imamo neku mjeru kojom će ocjenjivati atribute, mjera koja se obično koristi kod stabla odluke uključujući i ID3 algoritam (prethodnik C4.5 algoritma) je informacijska dobit.

Informacijska dobit

Teorija informacije govori nam da informacija koju prenosi poruka ovisi o njezinoj vjerojatnosti, označimo je sa p, i može se izraziti u bitovima korištenjem formule:

-log2(p).

Promotrimo skup primjera T koji pripadaju jednoj od kategorija iz skupa C. Ako nasumice uzmemo jedan primjer iz skupa T i odredimo da pripada kategoriji Ci, vjerojatnost te poruke je:

$$freq\frac{(C_i,T)}{|T|}$$

freq(Ci,S)/|S|, odnosno informacije koju ona prenosi je:

$$-\log_2\left(\frac{freq(C_i,T)}{|T|}\right) \quad \text{bita.}$$

Da bi našli očekivanu količinu informacija potrebnu da odredimo kojoj klasi pripada poruka koristimo slijedeći izraz:

$$info(T)=-\sum_{i=1}^{k}\frac{freq(C_i,T)}{|T|}\times\log_2\left(\frac{freq(C_i,T)}{|T|}\right)$$

Ta veličina naziva se i entropija skupa T.
Ako sada skup T podijelimo na n dijelova ovisno o vrijednosti atributa X, očekivanu informacija možemo izraziti na slijedeći način:

$$info_X(T)=\sum_{i=1}^{n}\frac{|T_i|}{|T|}\times info_X(T_i)$$

Sada možemo definirati informacijsku dobit dobivenu podjelom skupa S pomoću atributa X na slijedeći način:

$$gain(X)=info(T)-info_X(T)$$

Kako to računanje u stvarnosti izgleda, možemo ukratko pogledati na primjeru prikazanom na slici 2.1, odnosno tablicom 2.1.

	Outlook	Temp (°F)	Humidity(%)	Windy?	Class
1.	sunny	75	70	true	Play
2.	sunny	80	90	true	Don't Play
3.	sunny	85	85	false	Don't Play
4.	sunny	72	95	false	Don't Play
5.	sunny	69	70	false	Play
6.	overcast	72	90	true	Play
7.	overcast	83	78	false	Play
8.	overcast	64	65	true	Play
9.	overcast	81	75	false	Play
10.	rain	71	80	true	Don't Play
11.	rain	65	70	true	Don't Play
12.	rain	75	80	false	Play
13.	rain	68	80	false	Play
14.	rain	70	96	false	Play

Tablica 2.1 - Primjer jednostavnog skupa za učenje da li je dan pogodan za igranje golfa.

Imamo 14 primjera za učenje, 9 ih pripada klasi Play, a 5 klasi Don't Play.

$$info(T) = -\frac{9}{14} \times \log_2\left(\frac{9}{14}\right) - \frac{5}{14} \times \log_2\left(\frac{5}{14}\right) = 0.940 \text{ bita.}$$

Ako koristimo atribut Outlook za podjelu osnovnog skupa T dobivamo:

$$info_{Outlook}(T) = \frac{5}{14} \times \left(-\frac{2}{5} \times \log_2\left(\frac{2}{5}\right) - \frac{3}{5} \times \log_2\left(\frac{3}{5}\right)\right) + \frac{4}{14} \times \left(-\frac{4}{4} \times \log_2\left(\frac{4}{4}\right) - \frac{0}{4} \times \log_2\left(\frac{0}{4}\right)\right) + \frac{5}{14} \times \left(-\frac{3}{5} \times \log_2\left(\frac{3}{5}\right) - \frac{2}{5} \times \log_2\left(\frac{2}{5}\right)\right)$$

$$info_{Outlook}(T) = 0.694 \text{ bita.}$$

$$gain(Outlook) = info(T) - info_{Outlook}(T) = 0.940 - 0.694 = 0.264 \text{ bita.}$$

Ako koristimo atribut Windy dobivamo slijedeće:

$$info_{Windy}(T)=\frac{6}{14}\times\left(-\frac{3}{6}\times\log_2\left(\frac{3}{6}\right)-\frac{3}{6}\times\log_2\left(\frac{3}{6}\right)\right)$$
$$+\frac{8}{14}\times\left(-\frac{6}{8}\times\log_2\left(\frac{6}{8}\right)-\frac{2}{8}\times\log_2\left(\frac{2}{8}\right)\right)$$

$info_{Windy}(T)=0.892$ bita.

$gain(Outlook)=info(T)-info_{Windy}(T)=0.940-0.892=0.048$ bita

Vidimo da je dobit u prvom slučaju veća te će stablo odluke graditi sa atributom Outlook kao korijenom stabla.
Na analogni način računaju se vrijednosti za pojedina podstabla.

Odnos dobiti (gain ratio)

Jedan od velikih nedostataka biranja atributa pomoću informacijske dobiti je pristranost kriterija prema atributima koji imaju mnogo različitih vrijednosti.
Ako uzmemo primjer gdje je atribut JMBG, podjela po njemu dovest će do onolikog broja podskupova koliko imamo primjera u skupu primjera za učenje, a svaki od podskupova imat će samo jedan element.
Kako svaki element u takvim podskupovima pripada samo jednoj klasi, informacija koju možemo dobiti jednaka je nuli, tj. infoX(T) = 0. Dobit je u slučaju podjele po tom atributu maksimalna, taj atribut postaje korijen stabla odlučivanja, a mi od toga nemamo nikakve koristi.
Novi kriterij koji sprečava mogućnost iz navedenog slučaja naziva se odnos dobiti (engl. gain ratio).
Da bi definirali taj kriterij potrebno je uvesti novu veličinu koja predstavlja potencijalnu informaciju stvorenu podjelom skupa primjera za učenje T u n podskupova:

$$split\,info(T)=-\sum_{i=1}^{n}\frac{|T_i|}{|T|}\times\log_2\left(\frac{|T_i|}{|T|}\right)$$

Koristeći taj izraz možemo definirati odnos dobiti:

$$gain\ ratio(X)=\frac{gain(X)}{split\ info(T)}$$

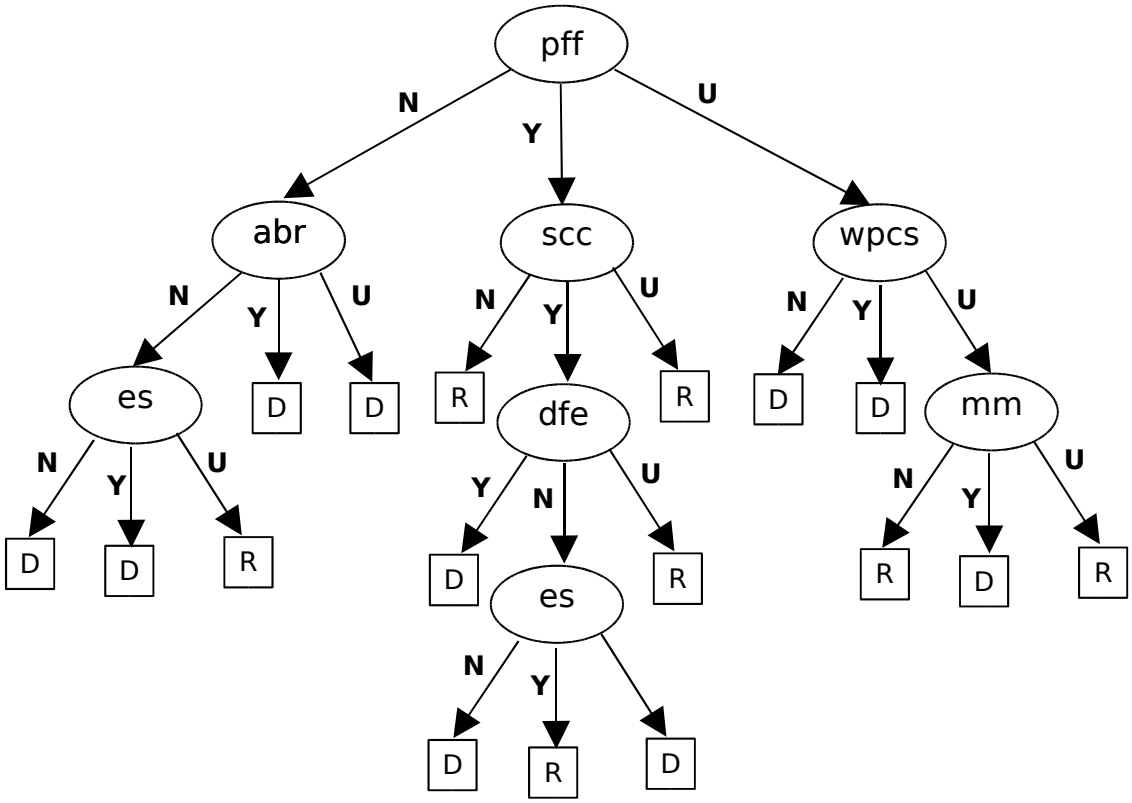
On nam pokazuje proporciju informacije, koju smo dobili podjelom skupa, koja nam je korisna, to jest koja nam pomaže u klasifikaciji.
Kao i prije uzima se onaj atribut koji ima najveći gain_ratio.

Ako se sad vratimo primjeru sa JMBG-ovim sa početka možemo vidjeti da će u tom slučaju gain_ratio biti mali. Ako imamo n primjera za učenje, koji spadaju u k klasa, uz uvjet da imamo puno više primjera za učenje nego klasa, lako možemo vidjeti da će gain_ratio biti malen, jer će informacijska dobit iznositi maksimalno log2(k) dok će split_info imati vrijednost log2(n) koja je puno veća od log2(k).

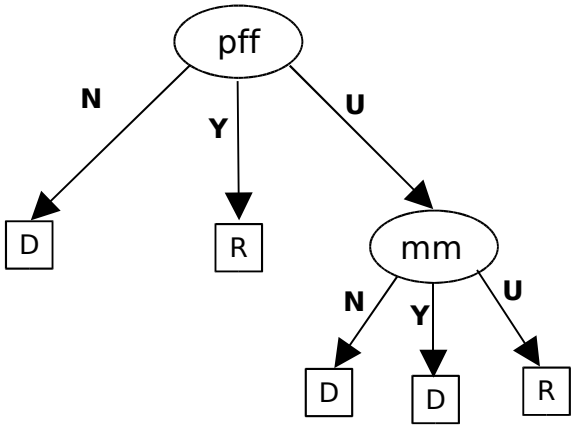
2.4.3 Obrezivanje stabla odluke

Proces izgradnje stabla može stvoriti vrlo velika stabla koja, osim što više nisu prikladna za simbolički prikaz, mogu imati vrlo veliku količinu grešaka u odnosu na manja stabla.
Ideja iza obrezivanja stabla je uklanjanje dijelova koji ne pridonose točnosti klasifikacije na neviđenim, odnosno testnim primjerima.
Postoje dvije različite metode smanjivanja stabla:
- zamjena podstabla sa listom
- zamjena podstabla sa jednom od njegovih grana.

Na slijedećim slikama prikazano je stablo prije i nakon obrezivanja.



Slika 2.2 - Stablo odluke za primjer obrezivanja stabla, prije obrezivanja.



Slika 2.3 - Stablo odluke za primjer obrezivanja stabla, nakon obrezivanja.

Možemo vidjeti da je podstablo sa korijenskim čvorom “abr” zamijenjeno listom “D”, dok je podstablo sa korijenskim čvorom “scc” zamijenjeno listom “R”.
Podstablo sa korijenskim čvorom wpcs zamijenjeno je sa podstablom u njegovoj trećoj grani.

Da bi mogli donijeti odluku da li zamijeniti podstablo sa listom ili jednom od njegovih grana, potrebno je na neki način procijeniti iznos pogreški. Ako imamo procjene pogrešaka za svako podstablo i list unutar stabla, procedura za smanjivanje stabla može se lako predočiti:

Krenemo od dna stabla i promatramo svako podstablo. Ako zamjena podstabla sa listom ili jednom od njegovih grana vodi do manje procijenjene pogreške, potrebno je smanjiti stablo, odnosno zamijeniti podstablo sa listom ili granom.

Kako pogreška cijelog stabla ovisi o pogreškama pojedinih podstabla, proces završava sa stablom koje ima minimiziranu procijenjenu pogrešku.

Pitanje na koje sada treba dati odgovor je kako možemo procijeniti navedene pogreške.

C4.5 koristi metodu koja za procjenu pogrešaka koristi samo skup primjera za učenje od kojih je stablo i konstruirano. Postoje i tehnike koje koriste poseban skup koji služi samo predviđanju pogreške.

Pogledajmo način na koji C4.5 procjenjuje iznos pogreški za svako podstablo,odnosno list.

Označimo sa N broj primjera za učenje koji pripadaju listu, a sa E broj onih koji ne pripadaju klasi koju označava list.

Ako na N gledamo kao na statistički uzorak, moguće je odrediti interval pouzdanosti za vjerojatnost krive klasifikacije primjera za učenje.

Nas zanima gornja granica tog intervala jer ona predstavlja najgori slučaj, označimo tu gornju granicu sa U_{CF}.

U_{CF} se definira kao $P(E/N <= U_{CF}) = CF$ gdje CF predstavlja nivo pouzdanosti.

Ako pretpostavimo da su greške u primjerima za učenje raspodijeljene prema binomnoj distribuciji sa vjerojatnošću p u N slučajeva, moguće je izračunati točnu vrijednost U_{CF} kao vrijednost od p za koju binomna distribucija slučajne varijable X pokazuje E uspješnih događaja u N pokušaja sa vjerojatnošću CF.

To možemo zapisati kao P(X<=E)=CF.

Ako uz prethodni izraz upotrijebimo formulu za binomnu distribuciju

$$P(X=x)=\binom{n}{x}\pi^{x}(1-\pi)^{n-x}$$

dobivamo izraz pomoću kojeg možemo izračunati U_{CF}:

$$P(X\leq E)=\binom{N}{E}U_{CF}^{E}(1-U_{CF})^{N-E}=CF$$

Pri tome je potrebno napomenuti da je standardna vrijednost za nivo pouzdanosti CF u C4.5 algoritmu 25%.

Kada smo našli gornju granicu U_{CF}, procjena pogreška listova i podstabala računa se sa pretpostavkom da se koriste za klasifikaciju

skupa do sad neviđenih primjera jednako velikog kao i skup primjera za učenje. Predviđena broj pogreška za određeni list onda iznosi N x U_{CF}, dok se pogreška za podstablo dobiva zbrajanjem vrijednosti pogreške za pojedini list.

Pogledajmo sada kako sam proces izgleda koristeći primjer prikazan na slikama 2.2. i 2.3.

Podstablo “abr” možemo prikazati na slijedeći način:

abr = N:
    es = N: D (6/0)
    es = Y: R (9/0)
    es = U: D (1/0)

Prikaz je analogan onom na slici 2.2., s jedinom razlikom da su dodani brojevi u zagradama koji su predstavljeni u obliku (N/E).
N označava sumu primjera za učenje koji su došli do određenog lista, dok E označava broj primjera koji su došli do lista, ali ne pripadaju klasi koju list određuje.

Pogledajmo prvi list, imamo 6 primjera, tj. N=6.
Koristeći vrijednost za nivo pouzdanosti 25% (koji je standardna vrijednost u C4.5 algoritmu) lako možemo izračunati gornju granicu U_{CF}.

Koristeći formulu za binomnu distribuciju: $P(X=x)=\binom{n}{x}\pi^x(1-\pi)^{n-x}$

Te uzevši u obzir da $P(X\leq E)=CF$ možemo napisati:

$$P(X\leq E)=\binom{N}{E}U_{CF}^{E}(1-U_{CF})^{N-E}=CF$$

Nakon uvrštenja konkretnih vrijednosti i malo računanja dobivamo slijedeću vrijednost: U_{CF} = 0.206.
Predviđeni broj pogrešaka ako koristimo taj list za klasifikaciju 6 do sad neviđenih slučajeva iznosi 6 x 0.206 = 1,236.

Na isti način dobit ćemo broj pogrešaka za preostala dva lista, one redom iznose:
9 x 0.143 = 1,287 i 1 x 0,750 = 0,750.
Predviđeni broj pogrešaka za cijelo stablo iznosi: 1,236 + 1,287 + 0,750 = 3,273.

Ako cijelo podstablo zamijenimo sa listom 'D', imat ćemo 16 slučajeva od kojih jedan ne pripada klasi 'D'. Dakle N=16, E = 1, te lako možemo izračunati da je U_{CF} jednak 0,157.

Predviđeni broj pogreška u ovom slučaju iznosi 16 x 0,157 = 2,512.
Vidimo da postojeće podstablo ima veći broj predviđenih pogrešaka, pa možemo podstablo zamijeniti sa listom.

2.5 Mjere učinkovitosti

Ocjenjivanje učinkovitost klasifikatora vrlo je bitan korak u klasifikaciji teksta, koji omogućuje uspoređivanje pojedinih metoda ali i pojedinih postavki konkretnog klasifikatora.

Ocjenjivanje se vrši pomoću eksperimenata, a cilj je izmjeriti efikasnost klasifikatora, odnosno sposobnost donošenja ispravnih klasifikacijskih pravila.

Ovdje ćemo prikazati mjere učinkovitosti [3] koje se najčešće koriste – preciznost i odaziv, te kombiniranu mjeru F_1.

2.5.1 Preciznost i odaziv

Prije same definicije preciznosti i odaziv, potrebno je definirati određene funkcije.

Funkcija $\breve{\Phi}(d_x, c_i)$ predstavlja nepoznatu ciljnu funkciju koja točno klasificira primjere, dok $\Phi(d_x, c_i)$ predstavlja njezinu aproksimaciju koju kreira klasifikator. Vrijednost T funkcija označava da je dokument d_x točno klasificiran pod kategoriju c_i.

Preciznost (π) se definira kao uvjetna vjerojatnost $P(\breve{\Phi}(d_x, c_i)=T|\Phi(d_x, c_i)=T)$ odnosno kao vjerojatnost da je klasifikacija slučajno odabranog dokumenta d_x u kategoriju c_i točna.

Odaziv (ρ) se definira kao uvjetna vjerojatnost $P(\Phi(d_x, c_i)=T|\breve{\Phi}(d_x, c_i)=T)$ odnosno kao vjerojatnost da se slučajno odabrani dokument d_x ako pripada u kategoriju c_i klasificira pod kategoriju c_i.

Iz definicije lako možemo vidjeti da se mjere računaju posebno za svaku kategoriju, no kako ćemo kasnije vidjeti, moguće ih je odrediti i za cijeli skup primjera za učenje.

Vrijednosti preciznosti i odaziva mogu se procijeniti koristeći tablice odlučivanja (contingency table).

Kategorija c_i		**Ispravna odluka**	
		DA	NE
Odluka klasifikatora	DA	TP_i (f++)	FP_i (f+-)
	NE	FN_i (f-+)	TN_i (f--)

Tablica 2.2 - Tablica odlučivanja za jednu kategoriju.

TP_i (true positive), odnosno f++ označava broj dokumenata za testiranje točno klasificiranih u kategoriju c_i.
FP_i (false positive), f+-, označava broj dokumenata za testiranje koji su pogrešno klasificiriani u kategoriju c_i.
FN_i (false negative), odnosno f-+, označava broj dokumenata koji pripada u kategoriju c_i ali nije klasificiran u kategoriju c_i.
TN_i (true negative), f--, označava broj dokumenata koji ne pripadaju u kategoriju c_i i nije klasificiran u kategoriju c_i.
Uzevši navedene vrijednosti u obzir preciznost i odaziv možemo aproksimirati na slijedeći način:

$$\pi_i = \frac{TP_i}{TP_i + FP_i} \quad , \quad \rho_i = \frac{TP_i}{TP_i + FN_i}$$

Aproksimaciju vrijednosti preciznosti i odaziva za cijeli skup za učenje moguće je izračunati koristeći dvije različite metode:

- mikroprosjek
  preciznost i odaziv računaju se sumiranjem po svim odlukama klasifikacije:

$$\pi^{\mu} = \frac{TP}{TP + FP} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FP_i)} \quad ,$$

$$\rho^{\mu} = \frac{TP}{TP + FN} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FN_i)}$$

  Oznaka μ označava da je riječ o mikroprosjeku.
  Globalna tablica odlučivanja dobiva se sumiranjem pojedinih tablica po svim kategorijama.

Skup kategorija C={c_1,...,$c_{|C|}$}		**Ispravna odluka**	
		DA	NE
Odluka klasifikatora	DA	$\sum_{i=1}^{\|C\|} TP_i$	$\sum_{i=1}^{\|C\|} FP_i$
	NE	$\sum_{i=1}^{\|C\|} FN_i$	$\sum_{i=1}^{\|C\|} TN_i$

Tablica 2.3 - Globalna tablica odlučivanja.

- makroprosjek
  Preciznost i odaziv prvo se računaju posebno za svaku kategoriju, a zatim se globalna vrijednost računa dijeljenjem sa brojem kategorija:
  $$\pi^M = \frac{\sum_{i=1}^{|C|} \pi_i}{|C|} \quad , \quad \rho^M = \frac{\sum_{i=1}^{|C|} \rho_i}{|C|}$$
  Oznaka M označava da je riječ o makroprosjeku.

2.5.2 Kombinirane mjere

Iako preciznost i odaziv točno opisuju performanse klasifikatora, korištenjem dvije mjere teško se uspoređuju rezultati različitih klasifikatora. Zbog toga se često koriste kombinirane mjere.
Od dostupnih mjera koje kombiniraju vrijednosti preciznosti i odaziva, koristit ćemo F_1 mjeru.
Općenito, F_β mjeru definiramo na slijedeći način:

$$F_\beta = \frac{(\beta^2+1)\pi\rho}{\beta^2\pi+\rho}$$

β je parametar, a vrijednost koja se obično koristi je 1, čime se pridaje jednaka težina preciznosti i odazivu.
Nakon uvrštavanja vrijednosti 1 za β, dobivamo slijedeću formulu:

$$F_1 = \frac{2\pi\rho}{\pi+\rho}$$

F_β mjeru možemo procijeniti i pomoću tablice odlučivanja na slijedeći način:

$$F_\beta = \frac{(\beta^2+1)TP}{(\beta^2+1)TP+FP+\beta^2 FN}$$

Odnosno, ako postavimo da je β jednak 1, dobivamo:

$$F_1 = \frac{2\,TP}{2\,TP + FP + FN}$$

3 Opis aplikacije

3.1 Korišteni alati

Svi dijelovi aplikacije pisani su u programskom jeziku C++, izuzev C4.5 algoritma, koji je pisan u programskom jeziku C.
Kao platforma za razvoj aplikacije izabran je Debian GNU/Linux [12] operacijski sustav.

Za razvoj grafičkog sučelja korišten je GTK+ toolkit [10], te njegov C++ wraper GTKMM [11]. GTK je danas gotovo standardni widget set za razvoj GUI aplikacija na GNU/Linux operacijskim sustavima, te temelj na kojem je izgrađeno popularno GNOME desktop okruženje [9].

Kao kompajler korišten je GNU C kompajler (GCC) koji je standardni dio svake GNU/Linux distribucije.

Zahvaljujući portu GTK i GTKMM biblioteka na Windows operacijske sustave te Cygwin okruženju [15], izgrađena je i potpuno funkcionalna Windows verzija aplikacije.

3.2 Ulazni podaci

Za testiranje klasifikatora potrebno je imati skup podataka za učenje i testiranje.
Dostupni skupovi za učenje i testiranje nemaju unificiranu strukturu dokumenata, nego svaki skup ima vlastitu strukturu. Ta činjenica otežava mogućnost izgradnje aplikacije koja bi mogla prihvatiti sve dostupne skupove.
Kao rješenje ovog problema nameću se dvije mogućnosti:

- izgradnja aplikacije koja može parsirati samo neke od dostupnih skupova
- izgradnja aplikacije koja koristi vlastitu strukturu skupova za učenje i testiranje.

U prvom slučaju potrebno je točno znati sa kojim skupovima ćemo raditi te uključiti mogućnost njihovog parsiranja u aplikaciju, ako želimo koristiti dodatne skupove, potrebno je mijenjati aplikaciju.
Drugi slučaj puno je transparentniji. Strukturu skupova sa kojima želimo raditi potrebno je samo pretvoriti u onu kakvu prepoznaje aplikacija. Kada želimo koristiti dodatne skupove, potrebno je napisati kratki program koji će strukturu dokumenata pretvoriti u onu kakvu koristi aplikacija.

Prilikom izgradnje sustava za klasifikaciju teksta pomoću C4.5 algoritma korištena je druga metoda, a za opis strukture dokumenata korišten je XML.

3.2.1 Struktura XML dokumenata

Strukturu XML dokumenata prikazat ćemo na primjeru dokumenta iz Reuters skupa za testiranje:

```
<ROOT>
	<DATA>
		<SET>TEST</SET>
		<NO>2972</NO>
		<TOPIC>acq</TOPIC>
		<TEXT>BROWN DISC TO BUY RHONE-POULENC &lt;RHON.PA> UNIT COLORADO SPRINGS, Colo., Oct 19 - Brown Disc Products Co Inc, a unit fo Genevar Enterprises Inc, said it has purchased the ongoing business, trademarks and certain assets of Rhone- Poulenc's Brown Disc Manufacturing unit, for undisclosed terms. Rhone-Poulenc is a French-based chemical company. Under the agreement, Rhone-Poulenc will supply magnetic tape and media products to Brown Disc Products. Reuter</TEXT>
	</DATA>
</ROOT>
```

ROOT tag nalazi se na početku i kraju svakog dokumenata. Između se može nalaziti proizvoljan broj pojedinih dokumenata razgraničenih DATA tagom.
DATA tag mora obavezno sadržavati slijedeće tagove: SET, TOPIC i TEXT, dok tag NO može, ali i ne mora biti prisutan.
Unutar SET taga definiramo da li je riječ o dokumentu iz skupa za učenje (TRAIN) ili iz skupa za testiranje (TEST).
NO označava broj dokumenta i služi samo za lakšu orijentaciju prilikom pregledavanja XML dokumenata.
TOPIC tag sadrži kategoriju, u slučaju da ima više kategorija, razgraničuju se znakom razmaka “ “.
Unutar TEXT taga nalazi se sam tekst dokumenta.

3.2.2 Reuters skup

Korišten je Reuters-21578 skup za učenje [6] koji sadrži bazu novinskih članaka iz 1987 godine.
Kolekcija se sastoji od 22 datoteke u SGML formatu. Svaka datoteka, osim posljednje sadrži 1000 dokumenata, dok posljednja sadrži 578 dokumenta, odnosno kolekcija sadrži ukupno 21578 dokumenata, po čemu je i dobila ime.

Prilikom konverzije kolekcije u XML format, korištena je Modified Apte (“ModApte”) podjela.
Podjela se sastoji od 9603 dokumenata za učenje i 3299 dokumenata za testiranje. Ova podjela napravljena je tako da bi svi dokumenti u skupu za učenje i za testiranje trebali bi imati barem jednu kategoriju, no neki dokumenti nemaju postavljenu niti jednu kategoriju.
Prilikom testiranja takvi dokumenti su izbačeni, a izbačeni su i dokumenti koji imaju više kategorija od jedne.
Više detalja o broju dokumenata nakon izbacivanja neprikladnih, broju kategorija te detaljna statistika skupova za učenje i testiranje bit će prikazana kasnije u poglavlju sa rezultatima.

Za konverziju originalne Reuters-21578 kolekcije iz SGML formata u XML format korišten je komandno-linijski program reuters_conv.exe koji konvertira kolekciju u skladu sa ModApte podjelom.
Jedini parametar koji program zahtjeva je ime direktorija u kojem se nalazi Reuters kolekcija. Nakon pokretanja programa u istom direktoriju bit će kreirane datoteke sa dodatkom ekstenzije xml na originalno ime datoteke.

3.2.3 Vjesnik skup

Korišten je Vjesnik skup [17], koji sadrži bazu novinskih članaka iz dnevnih novina Vjesnik.
Ovaj je skup vrlo zanimljiv jer je riječ o dokumentima na hrvatskom jeziku. Kategorije dokumenata vezane su uz rubrike u novinama: ck – crna kronika, go – gospodarstvo, ku – kultura, sp – sport, td – teme dana, un – unutarnja politika, vp – vanjska politika, zg – Zagreb.

Korištena je normalizirana verzija, koja je uz pomoć komandno-linijskog programa vjesnik_conv.exe konvertirana u XML format pogodan za korištenje u izgrađenoj aplikaciji.
Prilikom pokretanja jedini parametar koji program zahtjeva je ime direktorija sa Vjesnik kolekcijom, nakon završene konverzije u istom direktoriju bit će kreirane XML datoteke.

3.3 Obrada ulaznih podataka

Nakon što su skupovi podataka konvertirani iz raznih oblika u strogo formatirani XML zapis, moguće je krenuti sa njihovim procesiranjem.
Procesiranje, odnosno obrada podataka vrši se sa ciljem pretvaranja ulaznih podataka u oblik koji algoritam, odnosno program za klasifikaciju prepoznaje.

3.3.1 Princip rada

Tijekom obrade vrši se parsiranje dokumenata, koje omogućava izdvajanje riječi i kategorija iz podataka za učenje i testiranje. Osim toga, prilikom parsiranja vrši se brojanje dokumenata i brojanje pojavljivanja riječi po dokumentima. Ovdje se nudi i mogućnost uklanjanja sufiksa riječi (word stemming) koristeći Porterov algoritam [13]. Treba napomenuti da je ova opcija namijenjena isključivo za tekstove koji su pisani engleskim jezikom.

Iako je riječ u jeziku jasno definirana kao najmanja jedinica govora koja u jeziku ima svoje semantičko značenje, potrebno je definirati riječ onako kako ju vidi parser.

U slučaju implementiranog parsera riječ možemo definirati kao skup nenumeričkih znakova razdvojen prazninom i interpunkcijskim znakovima. Pod pojmom praznina obuhvaćeni su svi znakovi koji na neki način razdvajaju skup riječi kao što su znak za razmak, tabulator, novi red i drugi.
Pod pojam interpunkcijski znak ulaze svi znakovi koji nisu ni brojke ni slova. Ovdje treba napomenuti da se znak minus '-' ne izbacuje zbog različitih složenica koje se pojavljuju u tekstu, a naročito u imenu nekih kategorija u Reuters setu.
Isto tako sva slova u riječi transformiraju se u mala slova, dok se brojke ne koriste.
Prilikom izdvajanja riječi, ako je ta opcija uključena, vrši se uklanjanje nastavka (word stemming).
Nakon što su sve navedene operacije izvršene riječ se dodaje u skup riječi prisutnih u dokumentu, odnosno u skup riječi prisutnih u cijelom setu.

Navedene operacije vrše se neovisno o tome da li je riječ o dokumentu za učenje ili testiranje, no prilikom kreiranja globalnog skupa, odnosno skupa koji je sačinjen od značajki za klasifikaciju, koriste se isključivo riječi prisutne u dokumentima za učenje.

U ulaznom XML skupu pojedini dokument može imati više kategorija. U tom slučaju postoji nekoliko mogućnosti:

- stvaraju se novi potpuno jednaki dokumenti, od kojih svaki ima jednu od originalnih kategorija
- dokumenti sa više kategorija se izbacuju i ne koriste u klasifikaciji

Iako implementacija sadrži obje mogućnosti, njihov izbor moguć je samo prilikom kompajliranja koda, a standardno se koristi druga metoda.
Nakon izbacivanja dokumenata sa više kategorija primijećeno je da dokumenti u skupu za testiranje imaju kategorije koje ne postoje u skupu za učenje. Takvi dokumenti bit će krivo klasificirani, te su i oni izbačeni prilikom parsiranja.

Nakon što je parsiranje dokumenata završeno i strukture podataka su napunjene, moguće je izbaciti riječi koje ne doprinose klasifikaciji.
U programskoj implementaciji nude se dvije mogućnosti za ovaj zadatak:

- izbacivanje riječi koje se često pojavljuju
- korištenje frekvencije pojavljivanja riječi u dokumentima

Obje mogućnosti moguće je koristiti neovisno jednu o drugoj. Riječi koje se često pojavljuju nalaze se u posebnoj datoteci, koju je iz grafičkog sučelja aplikacije moguće pregledavati te dodavati i brisati riječi.
Broj dokumenata u kojima se riječ mora pojaviti da bi ju se uzelo kao značajku za klasifikaciju moguće je podešavati od 0 do 200 u koracima po 1 dokument.

Kada su izabrane značajke, može se krenuti na kreiranje izlaznih datoteka namijenjenih C4.5 klasifikatoru.
Izborom značajki stvoren je vektor određene veličine pomoću kojeg će biti prikazani svi dokumenti u skupu za učenje i testiranje.
Element vektora dokumenta predstavlja broj pojavljivanja riječi u tom dokumentu, a ako se riječ ne pojavljuje stavlja se vrijednost 0.

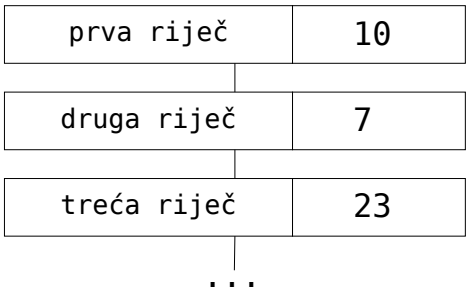
3.3.2 Programska implementacija

Kao i ostatak aplikacije i ovaj dio napisan je koristeći C++ jezik i Standard Template Library (STL).
Stvorena je klasa koja nudi sve mogućnosti potrebne u ovom dijelu kao što su parsiranje XML dokumenata, stvaranje lista riječi, izbacivanje riječi, kreiranje datoteka namijenjenih C4.5 algoritmu.

Strukture podataka

Osnovna struktura koja se koristi za prikaz riječi i njihovog broja unutar pojedinog dokumenta ili cjelokupnog seta je asocijativno polje.
Asocijativno polje, odnosno u C++ terminima map je struktura koja povezuje jedan objekt sa drugim kao kod standardnih polja. Razlika je u tome što se kao ključ, koji je kod standardnih polja integer vrijednost, može koristiti objekt bilo koje vrste. Prednosti asocijativnog polja još su automatsko sortiranje i nemogućnost da dva elementa imaju isti ključ.
Kao ključ polja koriste se pojedine riječi, a vrijednost ovisi o tome da li je riječ o mapi za pojedini dokument ili globalnoj mapi.

U slučaju mape za pojedini dokument vrijednost je broj koji govori koliko puta se riječ pojavila u tom dokumentu, dok kod globalne mape vrijednost označava broj dokumenata u kojima se riječ pojavljuje.
Grafički se ova struktura može prikazati na slijedeći način:

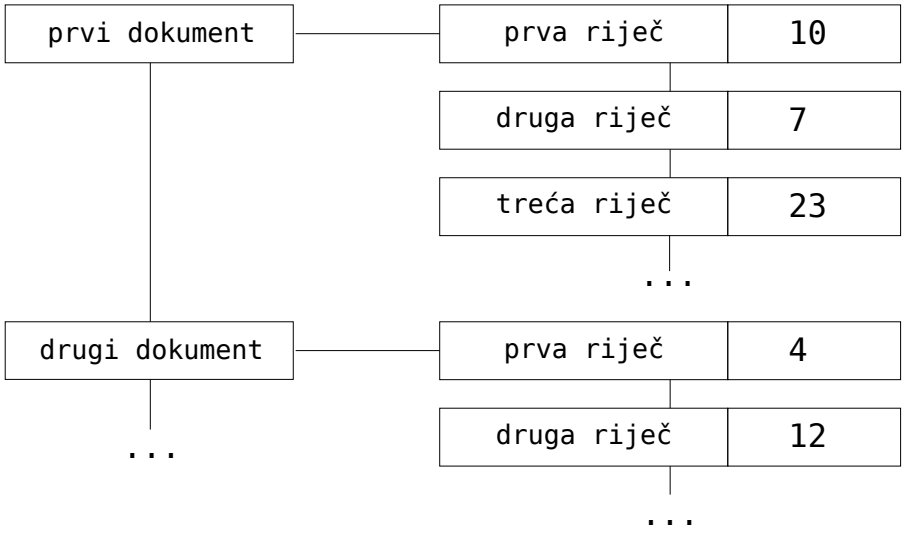


Slika 3.1 - struktura za čuvanje broja pojavljivanja riječi u svim dokumentima

Na potpuno jednak način i korištenjem jednake strukture riješen je i problem smještaja globalne liste kategorija. Ovdje se kao ključ koristi ime kategorije, dok je vrijednost jednaka broju dokumenata koji su sadržani u toj kategoriji.

Kada je riješen problem prikaza riječi unutar dokumenta, red je na izboru strukture koje će sadržavati sve dokumente iz skupa za učenje odnosno testiranje.

Struktura koja je izabrana za ovaj zadatak je red, odnosno deque. Riječ je o strukturi koja je ponašanjem vrlo slična standardnom polju.
Jedan red stvoren je posebno za dokumente za učenje, a drugi za dokumente za treniranje. Objekti od kojih je sačinjen red su polja sa riječi kao ključem i brojem pojavljivanja riječi u dokumentu kao vrijednošću.
Struktura se grafički može prikazati na slijedeći način:



Slika 3.2 - struktura za čuvanje dokumenata iz skupa za učenje ili testiranje

Za spremanje kategorija pojedinog dokumenta također se koristi struktura red. Kao i kod prethodnih struktura jedna je stvorena za testne podatke, a druga za podatke za učenje. Red je u oba slučaja sačinjen od standardnih string objekata.
Na prvi pogled možda čudno izgleda razdvajanje sadržaja dokumenata od kategorije, no na taj način su stvorene jednostavnije strukture. Povezivanje dokumenta sa pripadnom kategorijom vrlo je jednostavno, isti index u oba reda označava isti dokument.

Proces parsiranja

Sam proces procesiranja datoteka sa dokumentima te stvaranje struktura može se, pojednostavljeno, prikazati slijedećim pseudokodom:

```
s = pretvori_datoteku_u_string_objekt();
i = 0;
dok je s različit od kraja stringa radi
      pronađi DATA tag
            pronađi SET tag
                  ako tag SET == TRAIN postavi zastavicu z = TRAIN;
                  inače postavi zastavicu z = TEST;
            pronađi TOPIC tag
                  ako z == TRAIN
                        dodaj kategoriju u mapu kategorije;
                        dodaj kategoriju u red train_kategorija[i];
                  inače
                        dodaj  kategoriju  u  red  test_kategorije[i];

            pronađi TEKST tag
                  tekst = sadržaj unutar TEKST taga;
                  lista_riječi=razdvoji_riječi(tekst);
                  za_svaku riječ iz liste_riječi radi
                        ako z == TRAIN
                              dodaj riječ u mapu riječi;
                              dodaj riječ u red dokument_train[i];
                        inače
                              dodaj riječ u red dokument_test[i];
      i=i+1;
```

Izbacivanje riječi

Kad proces parsiranja završi, strukture podataka su pune i može se krenuti na proces izbacivanja riječi. Nakon završenog procesa, bit će stvorena nova mapa jednaka onoj za čuvanje svih riječi u svim dokumentima, sa razlikom da će u ovoj biti samo bitne riječi.
Zahvaljujući dobro izabranih strukturama sam proces je vrlo jednostavan.

Izbacivanje čestih riječi svodi se na učitavanje riječi iz datoteke u red stringova, uspoređivanje sa riječima u mapi, te stvaranje nove mapa sa onim riječima i pripadnim vrijednostima koje nisu sadržane u redu stvorenom iz datoteke.

Izbacivanje riječi koje se pojavljuju u određenom broju dokumenata također je vrlo jednostavno. Gledaju se vrijednost pojedine riječi te se stvara nova mapa sa onim riječima čija vrijednost prelazi određenu granicu.

Stvaranje datoteka za C4.5 klasifikator

Za korištenje C4.5 klasifikatora potrebno je pripremiti tri datoteke sa slijedećim ekstenzijama: .names, .data i .test.
U .names datoteci nalazi se popis kategorija i atributa. Pogledajmo primjer jedne .names datoteke (datoteka ne odgovara stvarnom slučaju):

```
acq, alum, bop, carcass, cocoa, coffee, copper, cotton, cpi,
cpu.

account: continuous.
base: continuous.
class: continuous.
domestic: continuous.
expand: continuous.
final: continuous.
giving: continuous.
health: continuous.
investment: continuous.
information: continuous.
japan: continuous.
key: continuous.
late: continuous.
main: continuous.
name: continuous.
offset: continuous.
parent: continuous.
remain: continuous.
session: continuous.
time: continuous.
unchanged: continuous.
volume: continuous.
week: continuous.
year-ago: continuous.
```

Iako je struktura .names datoteke vidljiva iz primjera, neke stvari potrebno je dodatno objasniti.

U prvom redu nalaze se kategorije razdvojene znakom razmaka i zarezom, popis se može protezati i kroz više redova, a na kraju završava točkom.
Nakon jednog praznog reda slijedi popis atributa, osim imena navedena je i vrsta. Svaki atribut stavlja se u vlastiti red, nakon imena potrebno je staviti znak dvotočke i razmak, a zatim slijedi ime vrste. U slučaju klasifikacije teksta vrsta je uvijek continuous odnosno atribut poprima kontinuirane numeričke vrijednosti.
Kategorije, kojih u primjeru ima 10, nastali se ispisivanjem mapa sa popisom svih kategorija.
Atributi koji odgovaraju riječima, kojih u primjeru ima 24, su nastali ispisivanjem mape koja je nastala u procesu izbacivanje riječi.

Slijedeća datoteka je .data u kojoj se nalaze zapisani pojedini dokumenti iz skupa za učenje. Radi lakšeg objašnjenja formata i ovdje ćemo pogledati jednu proizvoljno kreiranu datoteku:

```
0, 1, 3, 5, 0, 1, 0, 0, 0, 1, 3, 0, 1, 0, 1, 0, 0, 3, 0, 2, 0,
3, 0, 0, acq
0, 0, 0, 6, 0, 0, 2, 0, 0, 0, 7, 0, 0, 0, 0, 4, 0, 0, 2, 0, 0,
0, 0, 4, alum
0, 0, 0, 0, 0, 6, 0, 0, 1, 0, 0, 6, 0, 6, 1, 0, 3, 0, 0, 0, 1,
0, 4, 0, bop
3, 0, 2, 0, 2, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 3, 0, 2, 0,
0, 0, 0, acq
1, 0, 0, 0, 5, 0, 0, 4, 0, 0, 0, 0, 1, 0, 0, 0, 0, 3, 0, 2, 0,
0, 0, 0, cotton
0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 2, 0, 1, 5, 0, 2, 0, 0, 0, 2, 0,
4, 0, 6, cpu
3,0, 1, 0, 5, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 3, 0, 2, 0,
  4, 0, 6, cpu
4,
```

Svaki redak u datoteci predstavlja jedan dokument iz skupa za učenje, u ovom primjeru tako postoji 6 dokumenata za učenje. Kako je u .names datoteci prisutno 24 atributa, tako se ovdje nalazi 24 broja odvojenih zarezom. Nakon brojeva nalazi se ime kategorije kojoj dokument pripada.Brojevi označuju koliko puta se pojedina riječ pojavljuje u dokumentu.
Kreiranje datoteke vrši se uz pomoć mape sa bitnim riječima, redom sa kategorijama te redom sa mapama pojedinih dokumenata. Sam proces najlakše je opisati slijedećim pseudokodom:

```
i = 0;
dok je i <  broj dokumenta iz skupa za učenje radi
     za_svaku riječ iz mape bitnih riječi radi
          ako riječ postoji u mapi dokument_train[i]
               ispiši vrijednost vezanu uz riječ u mapi
dokument_train[i];
          inače
               ispiši 0;
          ispiši ',';
     ispiši train_kategorija[i];
     ispiši znak_za_novi_redak;
     i=i+1;
```

Slijedeća datoteka je .test koja je istovjetna datoteci data, jedino što se umjesto datoteka za učenje koriste datoteke za testiranje.

3.4 C4.5

Prilikom izgradnje sustava za klasifikaciju teksta korištena je originalna implementacija C4.5 algoritma od autora J.R. Quinlana
Opis temeljne ideje algoritma dan je odjeljku 2.3. dok se detaljan opis algoritma, najvećim dijelom u obliku C koda, može pronaći u [1], a sam C kod je dostupan na web stranicama autora [14].
Za potrebe ovog rada sam kod je doživio minorne promjene koje su vezane uz prezentaciju rezultata klasifikacije. Kako je riječ o komandno linijskoj verziji, nemoguće je kvalitetno prikazati sve rezultate zbog ograničenog prostora u komandno linijskom sučelju. No kako implementacija sustava za klasifikaciju teksta koristi grafičko sučelje, sam izlaz iz C4.5 programa preusmjeren je i nakon njegovog parsiranja rezultati se mogu pregledno prikazati.

3.5 Upute za korištenje aplikacije

Kako je već prije spomenuto aplikacija je pisana korištenjem GTK widget seta, te će ovdje biti prikazano grafičko sučelje.
GTK nudi mogućnost promjene izgleda aplikacije korištenjem različitih skinova, a ovdje prikazane slike napravljene su korištenjem standardnog skina.

Kao i sve aplikacije namijenjene grafičkom okruženju sa prozorima, tako se i ova sastoji od prozora i dijaloga.
Pokretanjem aplikacije otvara se glavni prozor koji je prikazan slijedećom slikom:

Slika 3.3 - Glavni prozor aplikacije.- Text Processing tab

Sam prozor podijeljen je u tri dijela:

- traku sa izbornicima
- alatnu traku
- glavni dio sa dva taba – jedan za procesiranje teksta i drugi za C4.5 klasifikator

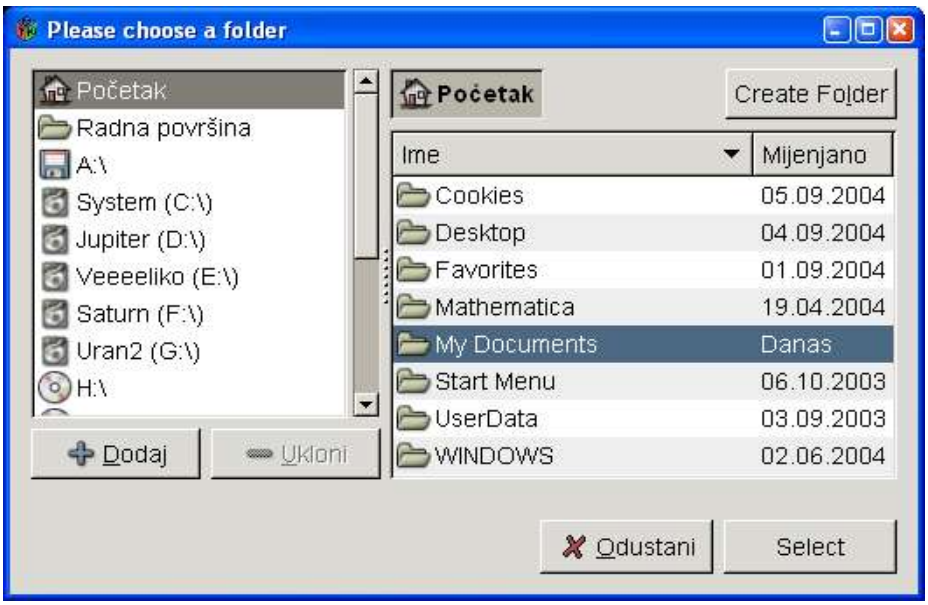
3.5.1 Text Processing dio

Traka sa izbornicima sadrži dva elementa – 'File' i 'Help'. Pod elementom 'Help' nalazi se samo opcija koja daje osnovne podatke o programu, dok element 'File' sadrži izbornik prikazan na slijedećoj slici:

Slika 3.4 - Sadržaj 'File' izbornika.

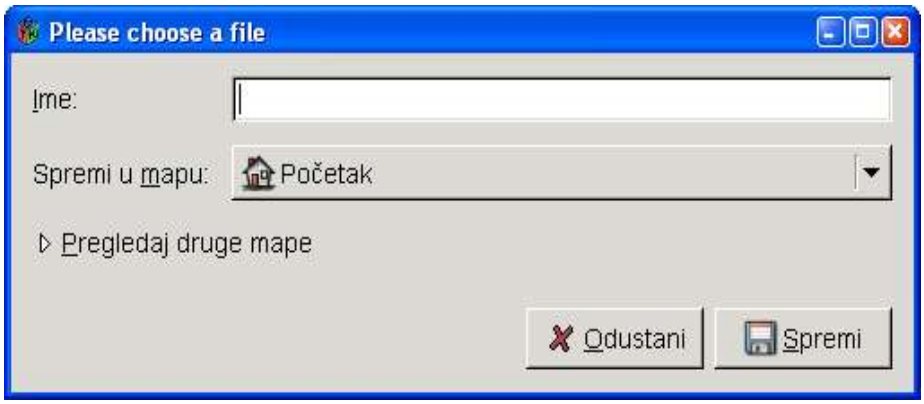
Pogledajmo čemu je namijenjenaa svaka od opcija.

Opcija 'Otvori' omogućava izbor direktorija u kojem se nalaze XML dokumenti sa skupovima za učenje i testiranje.
Sam dijalog nije nalik standardnim Windows dijalozima za izbor direktorija ili datoteke te je prikazan na slijedećoj slici:



Slika 3.5 - Dijalog za izbor direktorija

Opcija 'Spremi kao' omogućuje spremanje parsiranih podataka u obliku pogodnim za C4.5 klasifikator, odnosno kreiraju se prije navedene datoteke te nova datoteka sa ekstenzijom .info. Prilikom spremanja dovoljno je upisati samo ime, dok će program automatski dodati potrebne ekstenzije.
U datoteci sa ekstenzijom .info nalaze se podaci o broju dokumenata za učenje i testiranje, te broja atributa kako bi iste bile odmah dostupne u grafičkom sučelju.
Dijalog za spremanje datoteka prikazan je na slijedećoj slici:



Slika 3.6 - Dijalog za spremanje datoteka

Slijedeća grupa stavki u izborniku omogućuje spremanje i učitavanje postavki. Postavke su u biti sve strukture koje se kreiraju nakon parsiranja dokumenata, a spremaju se zbog toga da ne bi bilo potrebno ponovno parsirati ulazne XML datoteke, što kod velikih skupova može biti vremenski zahtjevna operacija.
Ekstenzija za datoteku u kojoj se nalaze postavke je .tpp i prilikom snimanja automatski će se dodati, dok će prilikom otvaranja dijaloga prikazati datoteke sa samo tom ekstenzijom. Sami dijalozi su analogni onima prikazanim na prethodnim slikama.

'Export' stavka omogućuje snimanje raspodijele kategorija po dokumentima u CSV (comma separeted values) formatu.

Pretposljednja stavka 'Opcije' nudi mogućnost izbora da li se prilikom parsiranja koristi Porterov stemming algoritam ili ne.

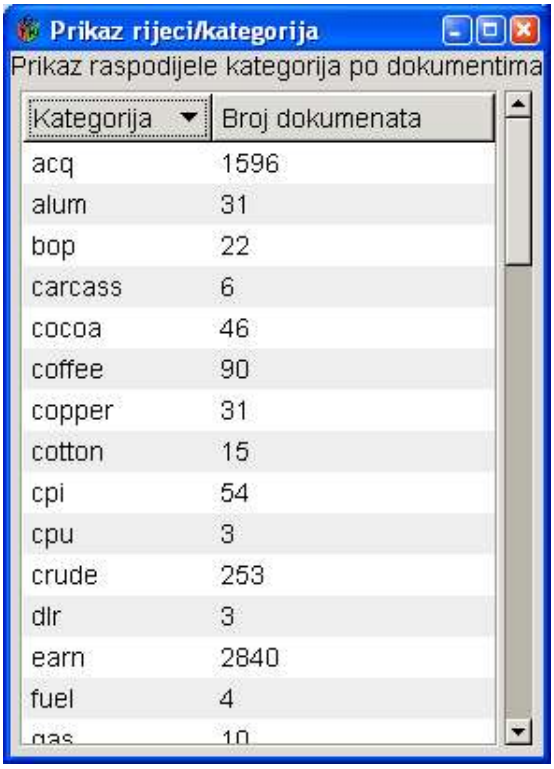
'Završi' zatvara samu aplikaciju.

Slijedeći element prozora je alatna traka. Na njoj se nalaze gumbi koji omogućuju slijedeće operacije: izbor direktorija sa XML datotekama, pokretanje procesa parsiranje, te snimanje C4.5 datoteka.

Glavni dio prozora podijeljen je pomoću tabova na dvije cjeline. Prvo ćemo pogledati elemente cjeline nazvane 'Text Processing'.

Ova cjelina podijeljena je na četiri dijela: 'Dokumenti', 'Kategorije', 'Riječi' i 'Izbor riječi'.

Prva dva dijela prikazuju broj dokumenata za učenje i testiranje, te broj kategorija. Klikom na gumb 'Raspodjela' kod dijela 'Kategorije' otvara se novi prozor, prikazan na slici #, koji prikazuje raspodjelu kategorija po dokumentima za učenje.



Slika 3.7 - Prozor koji prikazuje raspodjelu dokumenata po kategorijama

Slijedeća dva dijela međusobno su povezana. U dijelu 'Riječi' prikazan je ukupan broj riječi koje se pojavljuju u dokumentima za učenje, te broj bitnih riječi, odnosno broj onih riječi koje se koriste za samu klasifikaciju.

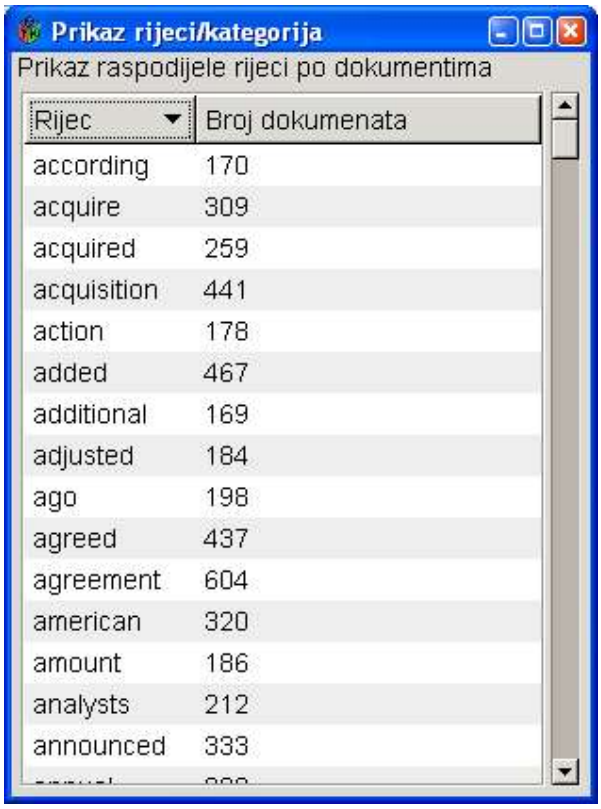
Da bi se dobio broj bitnih riječi, potrebno je podesiti postavke u dijelu 'Izbor riječi' te pritisnuti gumb 'Izbaci'.

Kao što je već i prije spomenuto, za izbacivanje riječi odnosno izbor značajki koristi se frekvencija pojavljivanja riječi u dokumentima. Pomoću slidera ili pomoću spin gumba vrši se izbor broja dokumenata u kojima se riječ mora pojaviti da bi se koristila u procesu klasifikacije.

Ovdje se nalazi i checkbutton 'Izbaci česte riječi', koji kada je pritisnut, izbacuje sve riječi koje se često pojavljuju iz skupa riječi.
Pritiskom na gumb 'Edit' otvara se novi prozor, prikazan slikom 3.8, koji omogućuje editiranje same datoteke sa čestim riječima. Datoteka se nalazi u istom direktoriju kao i sama aplikacija, a naziva se ceste_rijeci.

Slika 3.8 - Prozor koji omogućuje pregled, dodavanje i brisanje stop riječi.

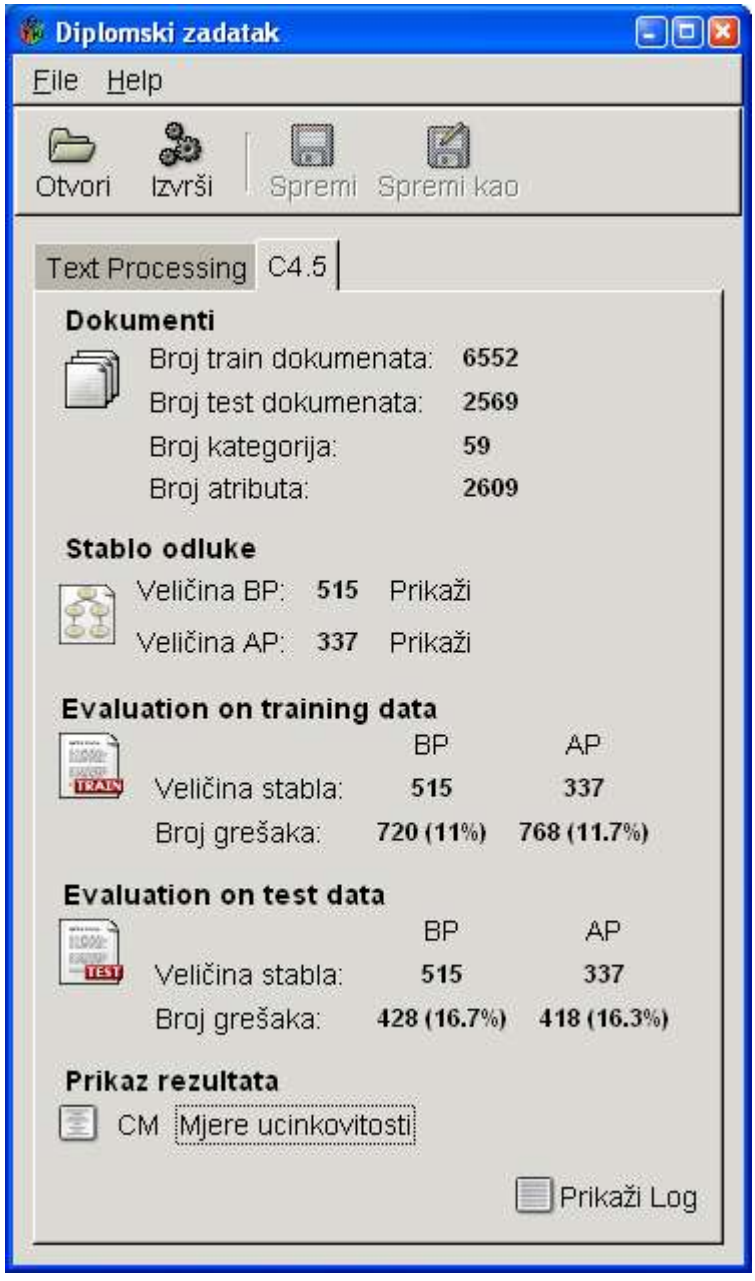
Nakon što je pritisnut gumb 'Izbaci' te su izbačene riječi koje se neće koristiti prilikom klasifikacije, moguće je kliknuti na gumb 'Bitne riječi' u dijelu 'Riječi'.
Tom akcijom otvara se novi prozor, prikazan na slici 3.9, koji prikazuje riječi te broj dokumenata u kojem se riječ pojavljuje.



Slika 3.9 - Prozor koji prikazuje raspodijelu bitnih riječi po dokumentima

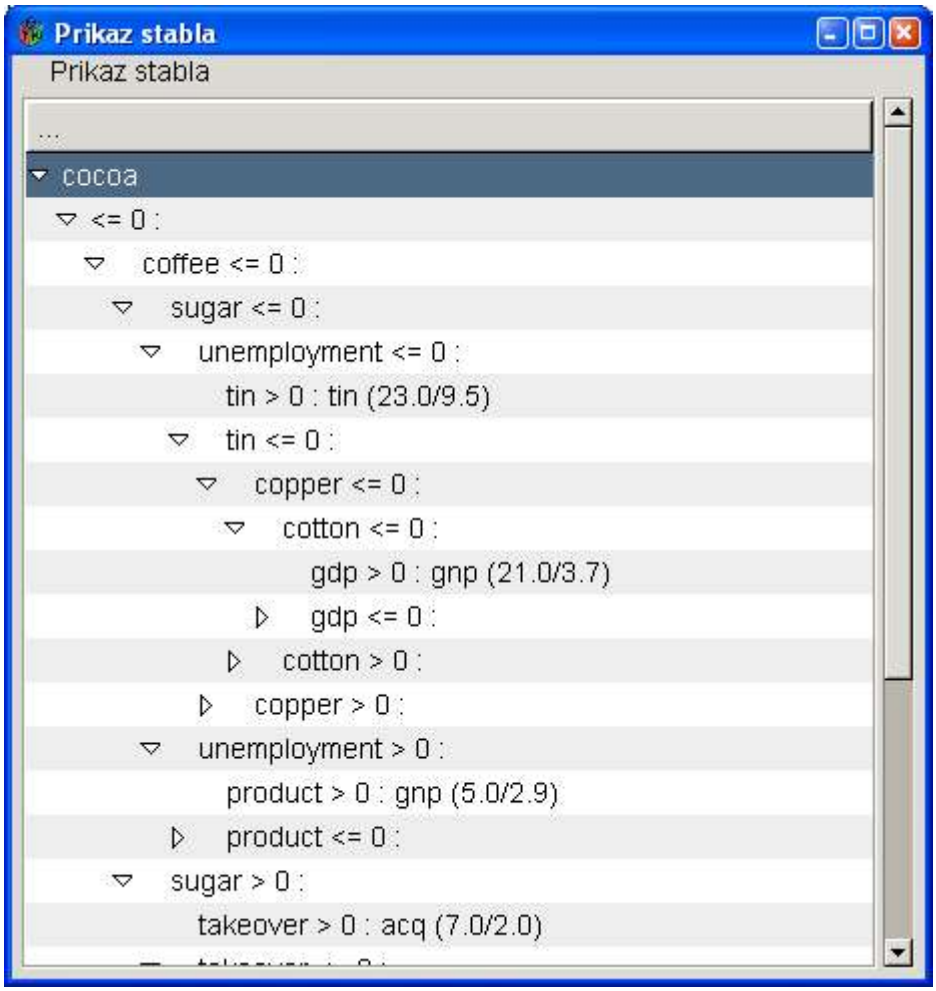
3.5.2 C4.5 dio

Nakon što je završen proces parsiranje, izbačene su nebitne riječi te snimljeni datoteke namijenjene c4.5 klasifikatoru, potrebno je pritisnuti drugi tab u glavnom prozoru, time glavni prozor poprima slijedeći oblik:



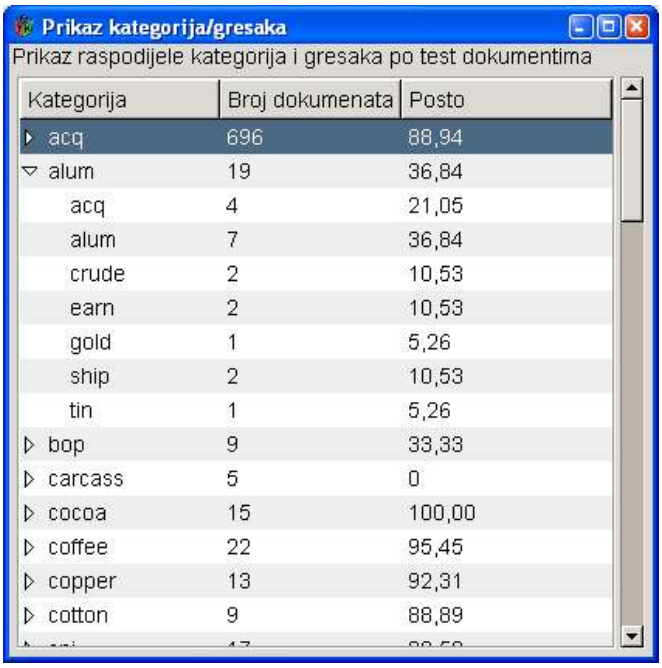
Slika 3.10 - Glavni prozor aplikacije.- C4.5 tab

File izbornik u ovom slučaju ima potpuno jednaku strukturu kao i u prethodnom, sa razlikom da su stavke 'Spremi' i 'Spremi Kao' onemogućene.
Pomoću 'Otvori' stavke bira se datoteka za c4.5 klasifikator sa .names ekstenzijom.
Postavke u ovom slučaju predstavljaju rezultati klasifikacije, a ekstenzija koja se koristi kod postavki za ovaj dio programa je c45p.
'Export' stavka omogućuje snimanje rezultata klasifikacije u CSV obliku.
Alatna traka nudi mogućnost izbora .names datoteka te sadrži gumb 'Izvrši' koji u ovom slučaju služi za pokretanje procesa klasifikacije.
Rezultati klasifikacije prikazani su u četiri dijela.
Prvi dio 'Dokumenti' prikazuje broj dokumenata za učenje i testiranje, broj kategorija i broj atributa. Broj atributa je broj bitnih riječi iz 'Text processing' cjeline.
Slijedeći dio 'Stablo odluke' prikazuje veličinu stabla prije procesa obrezivanja(BP – before prunning) te nakon procesa obrezivanja (AP – after prunning).. Osim brojčane vrijednosti, pritiskom na gumb 'Prikaži' moguće je vidjeti samo stablo. Stablo se prikazuje na istovjetan način kako se prikazuje stablo direktorija u Windows Exploreru ili Gnome Nautilusu.



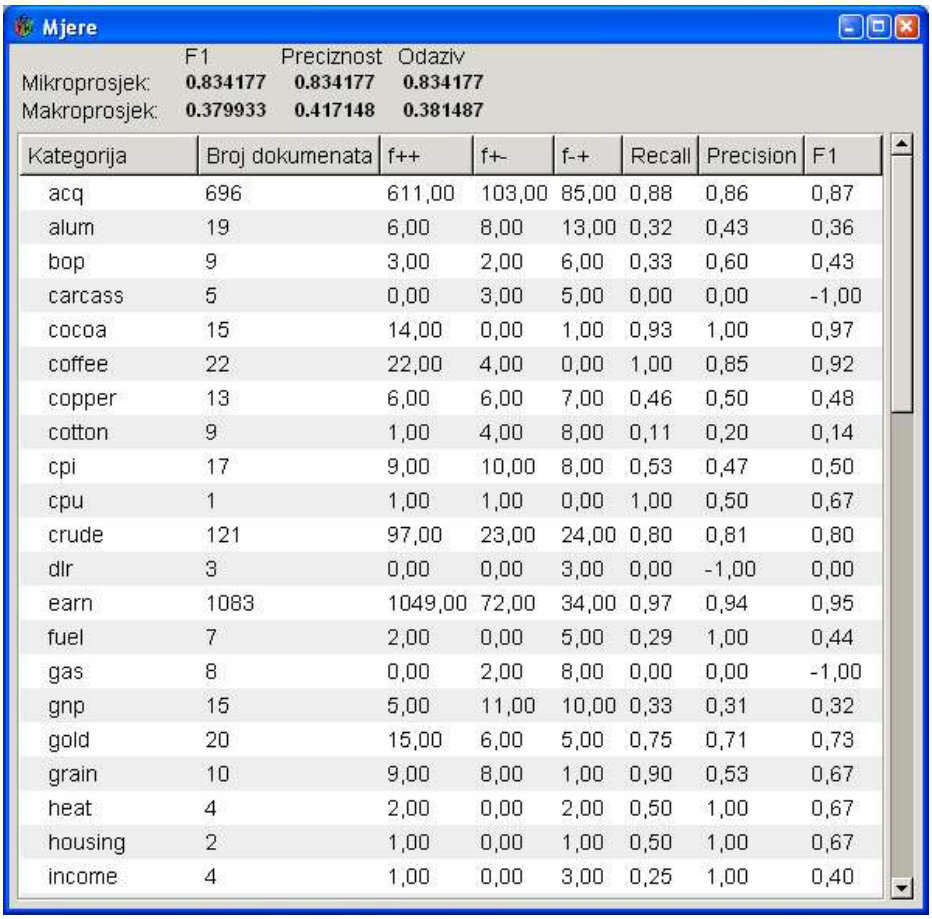
Slika 3.11 - Prikaz stabla odluke.

Slijedeći dio 'Evaluation on training data' prikazuje rezultate klasifikacije korištenjem samo podataka za učenje.
Broj grešaka predstavlja broj pogrešno klasificiranih dokumenata.
Dio 'Evaluation on test data' prikazuje rezultate klasifikacije za testne dokumente.
Posljednji dio je 'Prikaz rezultata' koji prikazuje confusion matrix – 'CM' gumb te mjere učinkovitosti.
Pritiskom na 'CM' gumb otvara se prozor prikazan na slici 3.12 koji prikazuje raspodjelu testnih dokumenata po kategorijama, postotak točno klasificiranih dokumenata iz određene kategorije te kategorije unutar kojih su klasificirani dokumenti koji su pogrešno klasificirani.



Slika 3.12 - Prikaz rezultata klasifikacije pomoću 'confusion matrice'.

Pritiskom na gumb 'Mjere učinkovitosti' pokreće se proces računanja preciznosti, odaziva, mikroprosjeka, makroprosjeka te F_1 mjere koji su definirani u odjeljku 2.4.
Nakon što je računanje završeno otvara se prozor sa rezultatima, prikazan na slici 3.13.



Kategorija	Broj dokumenata	f++	f+-	f-+	Recall	Precision	F1
acq	696	611,00	103,00	85,00	0,88	0,86	0,87
alum	19	6,00	8,00	13,00	0,32	0,43	0,36
bop	9	3,00	2,00	6,00	0,33	0,60	0,43
carcass	5	0,00	3,00	5,00	0,00	0,00	-1,00
cocoa	15	14,00	0,00	1,00	0,93	1,00	0,97
coffee	22	22,00	4,00	0,00	1,00	0,85	0,92
copper	13	6,00	6,00	7,00	0,46	0,50	0,48
cotton	9	1,00	4,00	8,00	0,11	0,20	0,14
cpi	17	9,00	10,00	8,00	0,53	0,47	0,50
cpu	1	1,00	1,00	0,00	1,00	0,50	0,67
crude	121	97,00	23,00	24,00	0,80	0,81	0,80
dlr	3	0,00	0,00	3,00	0,00	-1,00	0,00
earn	1083	1049,00	72,00	34,00	0,97	0,94	0,95
fuel	7	2,00	0,00	5,00	0,29	1,00	0,44
gas	8	0,00	2,00	8,00	0,00	0,00	-1,00
gnp	15	5,00	11,00	10,00	0,33	0,31	0,32
gold	20	15,00	6,00	5,00	0,75	0,71	0,73
grain	10	9,00	8,00	1,00	0,90	0,53	0,67
heat	4	2,00	0,00	2,00	0,50	1,00	0,67
housing	2	1,00	0,00	1,00	0,50	1,00	0,67
income	4	1,00	0,00	3,00	0,25	1,00	0,40

Slika 3.13 - Prikaz rezultata klasifikacije pomoću mjera učinkovitosti.

Posljednji gumb je 'Prikaži Log', a pritiskom na njega otvara se novi prozor koji sadrži sadržaj kakav bi ispisala komandno linijska verzija.

4 Rezultati

4.1 Statistika skupova za učenje

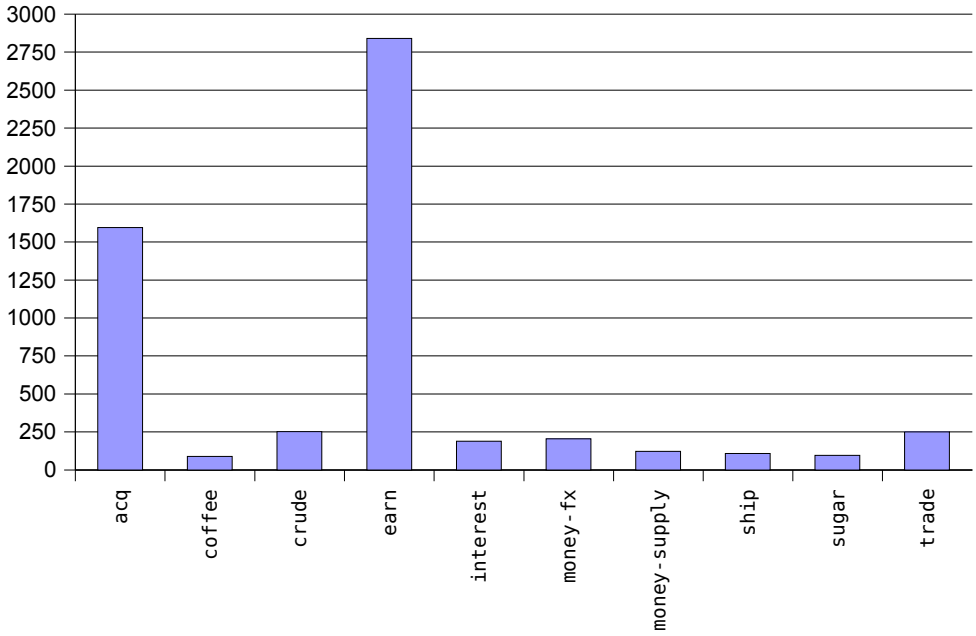
Aplikacija je testirana koristeći Reuters-21578 skup, te Vjesnik skup. Osnovne informacije o ovim skupovima date su u sekciji 3.2. a ovdje će biti prikazani njihovi detaljni statistički podaci.

4.1.1 Reuters-21578 – The Modified Apte (ModApte) Split

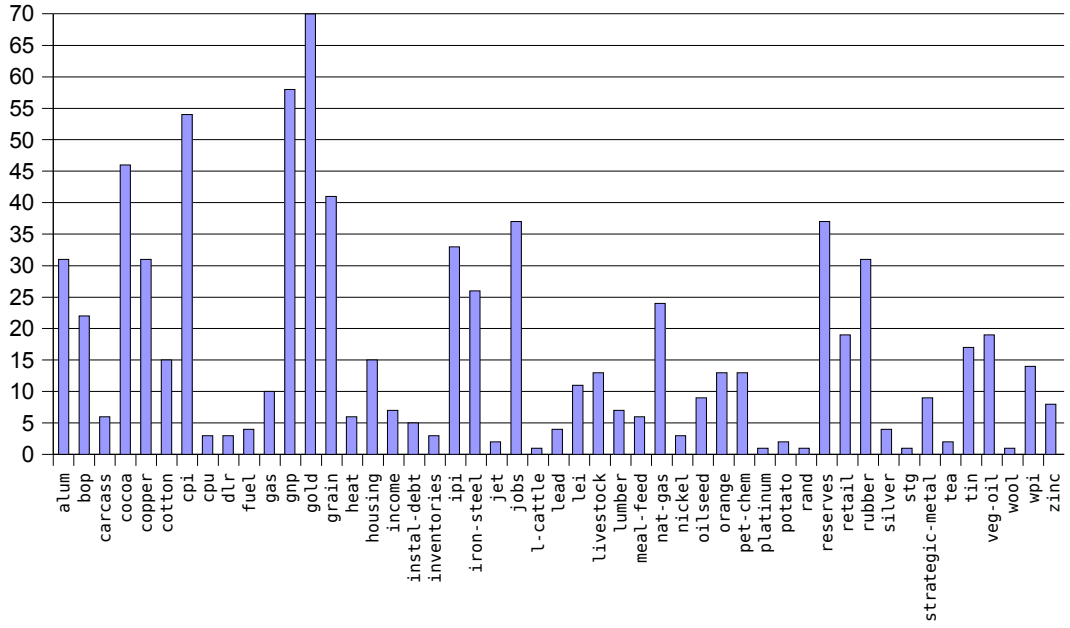
Originalna ModApte podjela sadrži 9603 dokumenata za učenje i 3299 dokumenata za testiranje.
Problem ove podjele je u tome što neki od dokumenata za učenje i testiranje nemaju postavljenu kategoriju, dok drugi imaju više kategorija. Kako C4.5 algoritam zahtjeva prisutnost samo jedne kategorije, izbačeni su svi dokumenti koji nemaju postavljenu kategoriju ili imaju više od jedne kategorije.

Time smo dobili kolekciju sa slijedećim karakteristikama:
- broj dokumenata za učenje: 6552
- broj dokumenata za testiranje: 2569
- broj kategorija: 59
- broj različitih riječi: 24342

Raspodjela kategorija po dokumentima za učenje prikazana je na slikama 4.1. i 4.2.

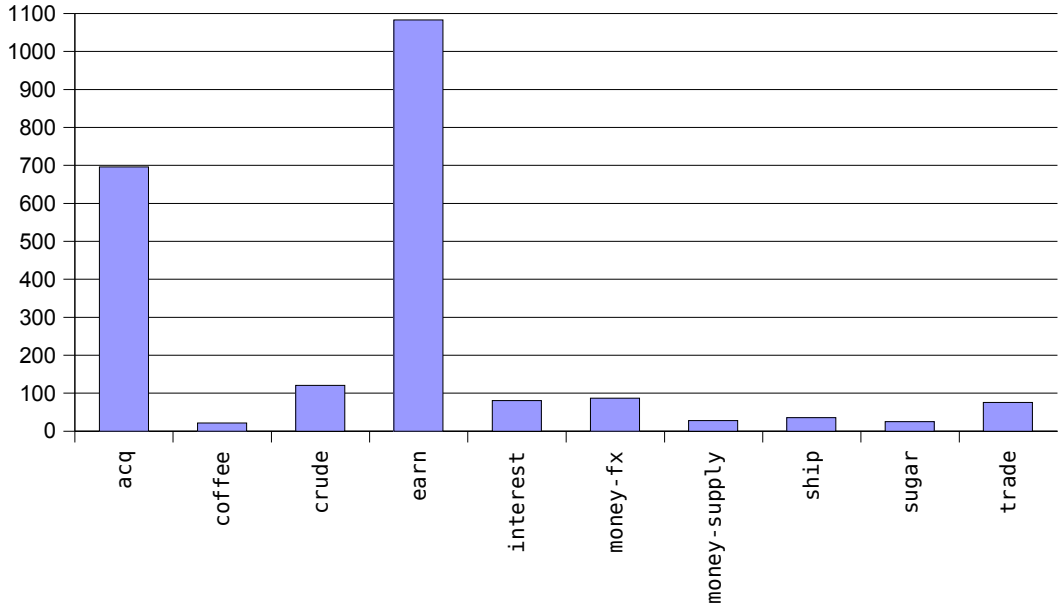


Slika 4.1 - Raspodjela kategorija po dokumentima za učenje – prvih 10 kategorija.

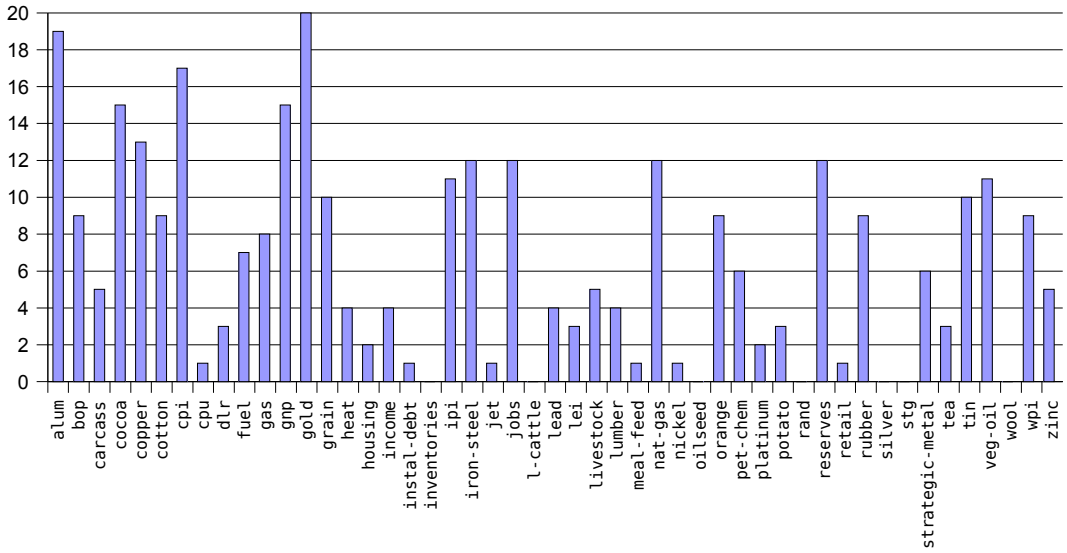


Slika 4.2 - Raspodjela kategorija po dokumentima za učenje – ostale kategorije.

Raspodjela kategorija po dokumentima za testiranje prikazana je na slikama 4.3. i 4.4.



Slika 4.3 - Raspodjela kategorija po dokumentima za testiranje – prvih 10 kategorija.



Slika 4.4 - Raspodjela kategorija po dokumentima za testiranje – ostale kategorije.

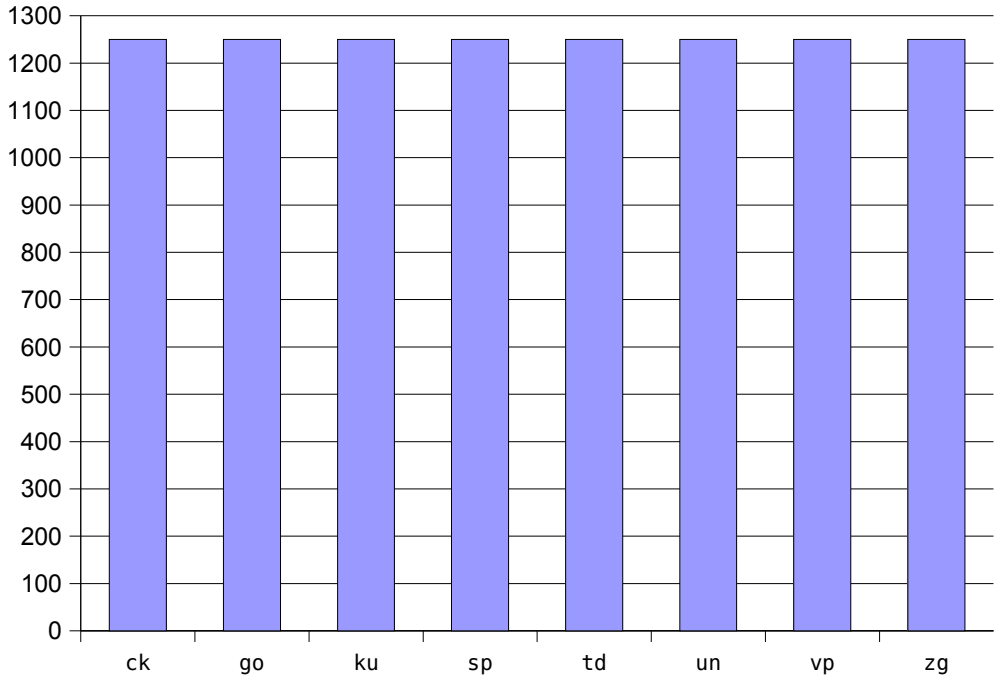
Na slikama je uočljivo da je distribucija dokumenata vrlo neujednačena te da najviše dokumenata u oba seta pripada kategorijama „acq“ i „earn“.

4.1.2 Vjesnik

Vjesnik kolekcija ima slijedeće karakteristike:

- broj dokumenata za učenje: 10000
- broj dokumenata za testiranje: 10000
- broj kategorija: 8
- broj različitih riječi: 72555

Raspodijela kategorija po dokumentima za učenje prikazana je na slici 4.5. Na potpuno jednak način kategorije su raspodijeljene i u dokumentima za testiranje.



Slika 4.5 - Raspodjela kategorija po dokumentima za učenje i testiranje.

4.2 Testiranja

4.2.1 Reuters set

Na Reuters skupu izvedeno je nekoliko testova sa različitim brojem značajki.
Prilikom svih testova izbačene su riječi koje se često pojavljuju u engleskom jeziku.

1. test

Izbačene su sve riječi koje se pojavljuju u manje od 3 dokumenata.

Broj različitih riječi:	24342
Broj riječi nakon izbacivanja:	8859
Broj train dokumenata:	6552
Broj test dokumenata:	2569
Broj kategorija:	59

Veličina stabla prije obrezivanja:	1013	
Broj pogrešno klasificiranih dokumenata:	449	(17,5 %)
Broj točno klasificiranih dokumenata:	2120	(82,5 %)

Veličina stabla nakon obrezivanja:	741	
Broj pogrešno klasificiranih dokumenata:	423	(16,5 %)
Broj točno klasificiranih dokumenata:	2146	(83,5 %)

Rezultati izračuna mjera učinkovitosti definirane u sekciji 2.4. prikazane su u tablici 4.1. dok se u tablici 4.2. nalaze rezultati za svaku pojedinu kategoriju.

	Preciznost	***Odaziv***	F_1
Mikroprosjek	0.84	0.84	0.84
Makroprosjek	0.49	0.44	0.45

Tablica 4.1 - Rezultati klasifikacije za sve kategorije, 1.test.

Oznaka ND u tablicima sa prikazom svih kategorija znači da vrijednost nije definirana.

Kategorija	Br. dok.	TP	FP	FN	Odaziv	Preciznost	F_1
acq	696	607	95	89	0.87	0.86	0.87
alum	19	9	14	10	0.47	0.39	0.43
bop	9	5	3	4	0.56	0.63	0.59
carcass	5	0	0	5	0.00	ND	0.00
cocoa	15	14	0	1	0.93	1.00	0.97
coffee	22	22	4	0	1.00	0.85	0.92
copper	13	11	0	2	0.85	1.00	0.92
cotton	9	9	0	0	1.00	1.00	1.00
cpi	17	6	7	11	0.35	0.46	0.40
cpu	1	1	1	0	1.00	0.50	0.67
crude	121	91	24	30	0.75	0.79	0.77
dlr	3	0	0	3	0.00	ND	0.00
earn	1083	1041	93	42	0.96	0.92	0.94
fuel	7	1	0	6	0.14	1.00	0.25
gas	8	3	1	5	0.38	0.75	0.50
gnp	15	8	2	7	0.53	0.80	0.64
gold	20	9	4	11	0.45	0.69	0.55
grain	10	9	7	1	0.90	0.56	0.69
heat	4	2	1	2	0.50	0.67	0.57
housing	2	1	0	1	0.50	1.00	0.67
income	4	2	1	2	0.50	0.67	0.57
instal-debt	1	1	0	0	1.00	1.00	1.00
interest	81	51	32	30	0.63	0.61	0.62
inventories	0	0	0	0	ND	ND	ND
ipi	11	6	5	5	0.55	0.55	0.55
iron-steel	12	4	3	8	0.33	0.57	0.42
jet	1	0	0	1	0.00	ND	0.00
jobs	12	11	1	1	0.92	0.92	0.92
l-cattle	0	0	0	0	ND	ND	ND
lead	4	0	0	4	0.00	ND	0.00
lei	3	2	1	1	0.67	0.67	0.67
livestock	5	3	5	2	0.60	0.38	0.46
lumber	4	0	0	4	0.00	ND	0.00
meal-feed	1	0	1	1	0.00	0.00	ND
money-fx	87	49	34	38	0.56	0.59	0.58
money-supply	28	17	15	11	0.61	0.53	0.57
nat-gas	12	8	11	4	0.67	0.42	0.52
nickel	1	0	0	1	0.00	ND	0.00
oilseed	0	0	5	0	ND	0.00	0.00
orange	9	8	2	1	0.89	0.80	0.84
pet-chem	6	1	2	5	0.17	0.33	0.22
platinum	2	0	0	2	0.00	ND	0.00
potato	3	0	0	3	0.00	ND	0.00
rand	0	0	0	0	ND	ND	ND
reserves	12	5	2	7	0.42	0.71	0.53
retail	1	0	2	1	0.00	0.00	ND
rubber	9	3	2	6	0.33	0.60	0.43
ship	36	17	14	19	0.47	0.55	0.51
silver	0	0	0	0	ND	ND	ND
stg	0	0	0	0	ND	ND	ND
strategic-metal	6	0	0	6	0.00	ND	0.00
sugar	25	25	3	0	1.00	0.89	0.94
tea	3	0	0	3	0.00	ND	0.00
tin	10	10	2	0	1.00	0.83	0.91
trade	76	60	23	16	0.79	0.72	0.75
veg-oil	11	7	0	4	0.64	1.00	0.78
wool	0	0	0	0	ND	ND	ND
wpi	9	2	0	7	0.22	1.00	0.36
zinc	5	5	1	0	1.00	0.83	0.91

Tablica 4.2 - Rezultati klasifikacije prema kategorijama, 1. test.

2. test

Izbačene su sve riječi koje se pojavljuju u manje od 7 dokumenta.

Broj različitih riječi:	24342
Broj riječi nakon izbacivanja:	4787
Broj train dokumenata:	6552
Broj test dokumenata:	2569
Broj kategorija:	59

Veličina stabla prije obrezivanja:	1027	
Broj pogrešno klasificiranih dokumenata:	438	(17 %)
Broj točno klasificiranih dokumenata:	2131	(83 %)

Veličina stabla nakon obrezivanja:	761	
Broj pogrešno klasificiranih dokumenata:	413	(16,1 %)
Broj točno klasificiranih dokumenata:	2156	(83,9 %)

Rezultati izračuna mjera učinkovitosti definirane u sekciji 2.4. prikazane su u tablici 4.1. dok se u tablici 4.2. nalaze rezultati za svaku pojedinu kategoriju.

	Preciznost	***Odaziv***	F_1
Mikroprosjek	0.84	0.84	0.84
Makroprosjek	0.48	0.44	0.44

Tablica 4.3 - Rezultati klasifikacije za sve kategorije, 2.test.

Oznaka ND u tablicima sa prikazom svih kategorija znači da vrijednost nije definirana.

Kategorija	Br. dok.	TP	FP	FN	Odaziv	Preciznost	F_1
acq	696	612	92	84	0.88	0.87	0.87
alum	19	9	14	10	0.47	0.39	0.43
bop	9	5	3	4	0.56	0.63	0.59
carcass	5	0	0	5	0.00	ND	0.00
cocoa	15	14	0	1	0.93	1.00	0.97
coffee	22	22	4	0	1.00	0.85	0.92
copper	13	11	0	2	0.85	1.00	0.92
cotton	9	9	0	0	1.00	1.00	1.00
cpi	17	6	7	11	0.35	0.46	0.40
cpu	1	1	1	0	1.00	0.50	0.67
crude	121	94	19	27	0.78	0.83	0.80
dlr	3	0	0	3	0.00	ND	0.00
earn	1083	1042	87	41	0.96	0.92	0.94
fuel	7	1	0	6	0.14	1.00	0.25
gas	8	3	1	5	0.38	0.75	0.50
gnp	15	8	2	7	0.53	0.80	0.64
gold	20	9	4	11	0.45	0.69	0.55
grain	10	9	9	1	0.90	0.50	0.64
heat	4	2	1	2	0.50	0.67	0.57
housing	2	1	1	1	0.50	0.50	0.50
income	4	1	1	3	0.25	0.50	0.33
instal-debt	1	1	0	0	1.00	1.00	1.00
interest	81	52	27	29	0.64	0.66	0.65
inventories	0	0	0	0	ND	ND	ND
ipi	11	6	5	5	0.55	0.55	0.55
iron-steel	12	4	7	8	0.33	0.36	0.35
jet	1	0	0	1	0.00	ND	0.00
jobs	12	11	1	1	0.92	0.92	0.92
l-cattle	0	0	0	0	ND	ND	ND
lead	4	0	0	4	0.00	ND	0.00
lei	3	2	1	1	0.67	0.67	0.67
livestock	5	3	5	2	0.60	0.38	0.46
lumber	4	0	0	4	0.00	ND	0.00
meal-feed	1	0	1	1	0.00	0.00	ND
money-fx	87	51	35	36	0.59	0.59	0.59
money-supply	28	17	16	11	0.61	0.52	0.56
nat-gas	12	8	11	4	0.67	0.42	0.52
nickel	1	0	0	1	0.00	ND	0.00
oilseed	0	0	5	0	ND	0.00	0.00
orange	9	6	2	3	0.67	0.75	0.71
pet-chem	6	1	1	5	0.17	0.50	0.25
platinum	2	0	0	2	0.00	ND	0.00
potato	3	0	0	3	0.00	ND	0.00
rand	0	0	0	0	ND	ND	ND
reserves	12	6	4	6	0.50	0.60	0.55
retail	1	0	1	1	0.00	0.00	ND
rubber	9	3	2	6	0.33	0.60	0.43
ship	36	17	14	19	0.47	0.55	0.51
silver	0	0	0	0	ND	ND	ND
stg	0	0	0	0	ND	ND	ND
strategic-metal	6	0	0	6	0.00	ND	0.00
sugar	25	25	3	0	1.00	0.89	0.94
tea	3	0	0	3	0.00	ND	0.00
tin	10	10	2	0	1.00	0.83	0.91
trade	76	60	23	16	0.79	0.72	0.75
veg-oil	11	7	0	4	0.64	1.00	0.78
wool	0	0	0	0	ND	ND	ND
wpi	9	2	0	7	0.22	1.00	0.36
zinc	5	5	1	0	1.00	0.83	0.91

Tablica 4.4 - Rezultati klasifikacije prema kategorijama, 2. test.

3. test

Izbačene su sve riječi koje se pojavljuju u manje od 17 dokumenta.

Broj različitih riječi:	24342
Broj riječi nakon izbacivanja:	2485
Broj train dokumenata:	6552
Broj test dokumenata:	2569
Broj kategorija:	59

Veličina stabla prije obrezivanja:	1041	
Broj pogrešno klasificiranih dokumenata:	424	(16,5 %)
Broj točno klasificiranih dokumenata:	2145	(83,5 %)

Veličina stabla nakon obrezivanja:	785	
Broj pogrešno klasificiranih dokumenata:	416	(16,2 %)
Broj točno klasificiranih dokumenata:	2153	(83,8 %)

	Preciznost	***Odaziv***	F_1
Mikroprosjek	0.84	0.84	0.84
Makroprosjek	0.47	0.41	0.42

Tablica 4.5- Rezultati klasifikacije za sve kategorije, 3. test.

Kategorija	Br. dok.	TP	FP	FN	Odaziv	Preciznost	F_1
acq	696	614	92	82	0.88	0.87	0.88
alum	19	5	10	14	0.26	0.33	0.29
bop	9	4	4	5	0.44	0.50	0.47
carcass	5	0	5	5	0.00	0.00	ND
cocoa	15	14	0	1	0.93	1.00	0.97
coffee	22	22	4	0	1.00	0.85	0.92
copper	13	11	0	2	0.85	1.00	0.92
cotton	9	9	0	0	1.00	1.00	1.00
cpi	17	5	9	12	0.29	0.36	0.32
cpu	1	1	1	0	1.00	0.50	0.67
crude	121	94	18	27	0.78	0.84	0.81
dlr	3	0	0	3	0.00	ND	0.00
earn	1083	1044	85	39	0.96	0.92	0.94
fuel	7	2	0	5	0.29	1.00	0.44
gas	8	3	5	5	0.38	0.38	0.38
gnp	15	9	3	6	0.60	0.75	0.67
gold	20	8	4	12	0.40	0.67	0.50
grain	10	9	3	1	0.90	0.75	0.82
heat	4	0	0	4	0.00	ND	0.00
housing	2	1	0	1	0.50	1.00	0.67
income	4	1	1	3	0.25	0.50	0.33
instal-debt	1	0	1	1	0.00	0.00	ND
interest	81	51	30	30	0.63	0.63	0.63
inventories	0	0	0	0	ND	ND	ND
ipi	11	6	4	5	0.55	0.60	0.57
iron-steel	12	2	7	10	0.17	0.22	0.19
jet	1	0	4	1	0.00	0.00	ND
jobs	12	11	1	1	0.92	0.92	0.92
l-cattle	0	0	0	0	ND	ND	ND
lead	4	1	3	3	0.25	0.25	0.25
lei	3	2	1	1	0.67	0.67	0.67
livestock	5	2	1	3	0.40	0.67	0.50
lumber	4	0	0	4	0.00	ND	0.00
meal-feed	1	0	1	1	0.00	0.00	ND
money-fx	87	50	34	37	0.57	0.60	0.58
money-supply	28	18	15	10	0.64	0.55	0.59
nat-gas	12	7	5	5	0.58	0.58	0.58
nickel	1	0	0	1	0.00	ND	0.00
oilseed	0	0	9	0	ND	0.00	0.00
orange	9	7	4	2	0.78	0.64	0.70
pet-chem	6	1	1	5	0.17	0.50	0.25
platinum	2	0	0	2	0.00	ND	0.00
potato	3	2	0	1	0.67	1.00	0.80
rand	0	0	0	0	ND	ND	ND
reserves	12	4	2	8	0.33	0.67	0.44
retail	1	0	2	1	0.00	0.00	ND
rubber	9	9	1	0	1.00	0.90	0.95
ship	36	19	19	17	0.53	0.50	0.51
silver	0	0	0	0	ND	ND	ND
stg	0	0	0	0	ND	ND	ND
strategic-metal	6	0	0	6	0.00	ND	0.00
sugar	25	25	3	0	1.00	0.89	0.94
tea	3	0	0	3	0.00	ND	0.00
tin	10	10	2	0	1.00	0.83	0.91
trade	76	62	21	14	0.82	0.75	0.78
veg-oil	11	6	0	5	0.55	1.00	0.71
wool	0	0	0	0	ND	ND	ND
wpi	9	2	0	7	0.22	1.00	0.36
zinc	5	0	1	5	0.00	0.00	ND

Tablica 4.6 - Rezultati klasifikacije prema kategorijama, 3. test.

4. test

Izbačene su sve riječi koje se pojavljuju u manje od 50 dokumenta.

Broj različitih riječi:	24342
Broj riječi nakon izbacivanja:	975
Broj train dokumenata:	6552
Broj test dokumenata:	2569
Broj kategorija:	59

Veličina stabla prije obrezivanja:	1087	
Broj pogrešno klasificiranih dokumenata:	448	(17,4 %)
Broj točno klasificiranih dokumenata:	2121	(82,6 %)

Veličina stabla nakon obrezivanja:	833	
Broj pogrešno klasificiranih dokumenata:	426	(16,6 %)
Broj točno klasificiranih dokumenata:	2143	(83,4 %)

	Preciznost	***Odaziv***	F_1
Mikroprosjek	0.83	0.83	0.83
Makroprosjek	0.42	0.38	0.38

Tablica 4.7. - Rezultati klasifikacije za sve kategorije, 4. test.

Kategorija	Br. dok.	TP	FP	FN	Odaziv	Preciznost	F_1
acq	696	633	110	63	0.91	0.85	0.88
alum	19	4	7	15	0.21	0.36	0.27
bop	9	3	4	6	0.33	0.43	0.38
carcass	5	0	1	5	0.00	0.00	ND
cocoa	15	3	15	12	0.20	0.17	0.18
coffee	22	8	16	14	0.36	0.33	0.35
copper	13	3	5	10	0.23	0.38	0.29
cotton	9	2	3	7	0.22	0.40	0.29
cpi	17	5	18	12	0.29	0.22	0.25
cpu	1	0	0	1	0.00	ND	0.00
crude	121	94	33	27	0.78	0.74	0.76
dlr	3	0	0	3	0.00	ND	0.00
earn	1083	1057	61	26	0.98	0.95	0.96
fuel	7	2	0	5	0.29	1.00	0.44
gas	8	0	0	8	0.00	ND	0.00
gnp	15	9	31	6	0.60	0.23	0.33
gold	20	1	23	19	0.05	0.04	0.05
grain	10	0	18	10	0.00	0.00	ND
heat	4	1	1	3	0.25	0.50	0.33
housing	2	0	1	2	0.00	0.00	ND
income	4	1	0	3	0.25	1.00	0.40
instal-debt	1	0	0	1	0.00	ND	0.00
interest	81	47	35	34	0.58	0.57	0.58
inventories	0	0	0	0	ND	ND	ND
ipi	11	6	5	5	0.55	0.55	0.55
iron-steel	12	3	5	9	0.25	0.38	0.30
jet	1	0	0	1	0.00	ND	0.00
jobs	12	4	4	8	0.33	0.50	0.40
l-cattle	0	0	0	0	ND	ND	ND
lead	4	0	2	4	0.00	0.00	ND
lei	3	0	2	3	0.00	0.00	ND
livestock	5	2	5	3	0.40	0.29	0.33
lumber	4	0	0	4	0.00	ND	0.00
meal-feed	1	0	0	1	0.00	ND	0.00
money-fx	87	36	31	51	0.41	0.54	0.47
money-supply	28	18	10	10	0.64	0.64	0.64
nat-gas	12	1	5	11	0.08	0.17	0.11
nickel	1	0	0	1	0.00	ND	0.00
oilseed	0	0	1	0	ND	0.00	0.00
orange	9	1	2	8	0.11	0.33	0.17
pet-chem	6	0	6	6	0.00	0.00	ND
platinum	2	0	0	2	0.00	ND	0.00
potato	3	0	0	3	0.00	ND	0.00
rand	0	0	0	0	ND	ND	ND
reserves	12	5	3	7	0.42	0.63	0.50
retail	1	0	3	1	0.00	0.00	ND
rubber	9	1	4	8	0.11	0.20	0.14
ship	36	4	17	32	0.11	0.19	0.14
silver	0	0	0	0	ND	ND	ND
stg	0	0	0	0	ND	ND	ND
strategic-metal	6	0	0	6	0.00	ND	0.00
sugar	25	14	24	11	0.56	0.37	0.44
tea	3	0	0	3	0.00	ND	0.00
tin	10	2	1	8	0.20	0.67	0.31
trade	76	57	24	19	0.75	0.70	0.73
veg-oil	11	1	2	10	0.09	0.33	0.14
wool	0	0	0	0	ND	ND	ND
wpi	9	0	3	9	0.00	0.00	ND
zinc	5	0	0	5	0.00	ND	0.00

Tablica 4.8 - Rezultati klasifikacije prema kategorijama, 4. test.

5. test

Izbačene su sve riječi koje se pojavljuju u manje od 192 dokumenta.

Broj različitih riječi:	24342
Broj riječi nakon izbacivanja:	243
Broj train dokumenata:	6552
Broj test dokumenata:	2569
Broj kategorija:	59

Veličina stabla prije obrezivanja:	1293	
Broj pogrešno klasificiranih dokumenata:	570	(22,2 %)
Broj točno klasificiranih dokumenata:	1999	(77,8 %)

Veličina stabla nakon obrezivanja:	927	
Broj pogrešno klasificiranih dokumenata:	541	(21,1 %)
Broj točno klasificiranih dokumenata:	2028	(78,9 %)

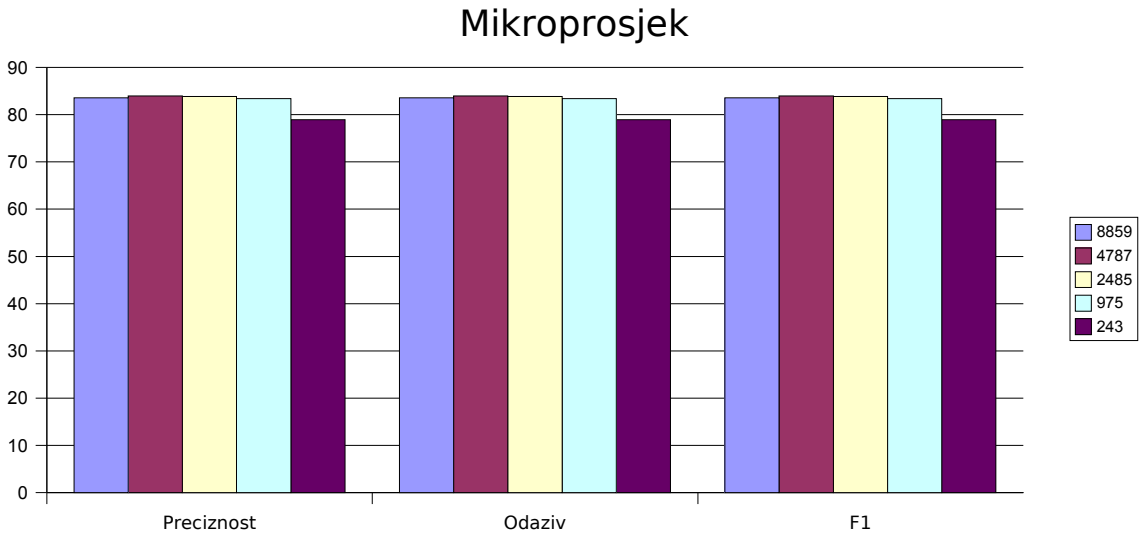
	Preciznost	***Odaziv***	F_1
Mikroprosjek	0.79	0.79	0.79
Makroprosjek	0.25	0.2	0.2

Tablica 4.9- Rezultati klasifikacije za sve kategorije, 5. test.

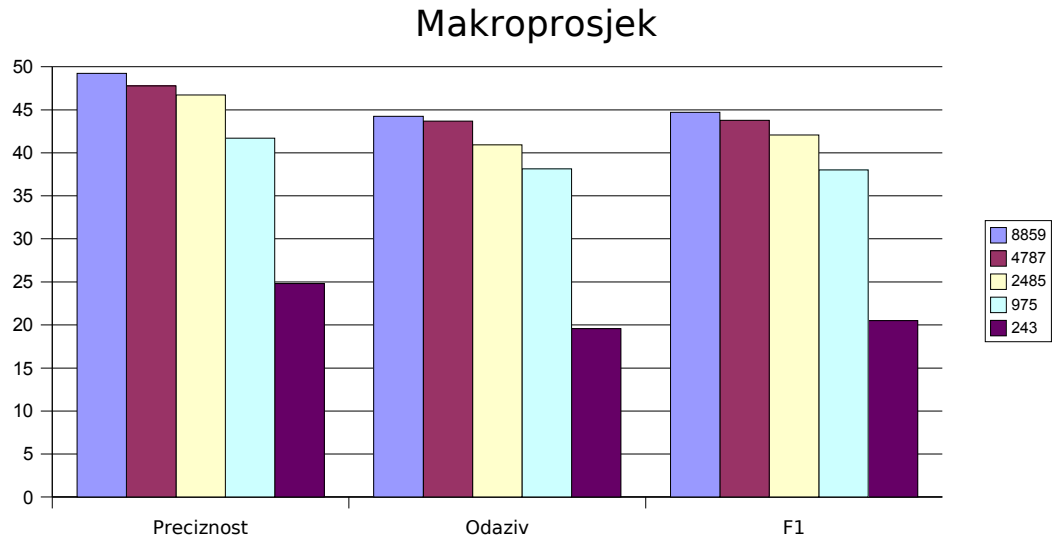
Kategorija	Br. dok.	TP	FP	FN	Odaziv	Preciznost	F_1
acq	696	611	103	85	0.88	0.86	0.87
alum	19	6	8	13	0.32	0.43	0.36
bop	9	3	2	6	0.33	0.60	0.43
carcass	5	0	3	5	0.00	0.00	ND
cocoa	15	14	0	1	0.93	1.00	0.97
coffee	22	22	4	0	1.00	0.85	0.92
copper	13	6	6	7	0.46	0.50	0.48
cotton	9	1	4	8	0.11	0.20	0.14
cpi	17	9	10	8	0.53	0.47	0.50
cpu	1	1	1	0	1.00	0.50	0.67
crude	121	97	23	24	0.80	0.81	0.80
dlr	3	0	0	3	0.00	ND	0.00
earn	1083	1049	72	34	0.97	0.94	0.95
fuel	7	2	0	5	0.29	1.00	0.44
gas	8	0	2	8	0.00	0.00	ND
gnp	15	5	11	10	0.33	0.31	0.32
gold	20	15	6	5	0.75	0.71	0.73
grain	10	9	8	1	0.90	0.53	0.67
heat	4	2	0	2	0.50	1.00	0.67
housing	2	1	0	1	0.50	1.00	0.67
income	4	1	0	3	0.25	1.00	0.40
instal-debt	1	1	1	0	1.00	0.50	0.67
interest	81	51	22	30	0.63	0.70	0.66
inventories	0	0	0	0	ND	ND	ND
ipi	11	6	5	5	0.55	0.55	0.55
iron-steel	12	2	5	10	0.17	0.29	0.21
jet	1	0	0	1	0.00	ND	0.00
jobs	12	8	3	4	0.67	0.73	0.70
l-cattle	0	0	0	0	ND	ND	ND
lead	4	3	4	1	0.75	0.43	0.55
lei	3	2	1	1	0.67	0.67	0.67
livestock	5	2	1	3	0.40	0.67	0.50
lumber	4	0	0	4	0.00	ND	0.00
meal-feed	1	0	6	1	0.00	0.00	ND
money-fx	87	61	29	26	0.70	0.68	0.69
money-supply	28	18	10	10	0.64	0.64	0.64
nat-gas	12	6	3	6	0.50	0.67	0.57
nickel	1	0	0	1	0.00	ND	0.00
oilseed	0	0	0	0	ND	ND	ND
orange	9	4	2	5	0.44	0.67	0.53
pet-chem	6	2	4	4	0.33	0.33	0.33
platinum	2	0	0	2	0.00	ND	0.00
potato	3	0	0	3	0.00	ND	0.00
rand	0	0	0	0	ND	ND	ND
reserves	12	8	1	4	0.67	0.89	0.76
retail	1	0	1	1	0.00	0.00	ND
rubber	9	3	6	6	0.33	0.33	0.33
ship	36	17	22	19	0.47	0.44	0.45
silver	0	0	0	0	ND	ND	ND
stg	0	0	0	0	ND	ND	ND
strategic-metal	6	0	0	6	0.00	ND	0.00
sugar	25	25	3	0	1.00	0.89	0.94
tea	3	0	0	3	0.00	ND	0.00
tin	10	2	2	8	0.20	0.50	0.29
trade	76	60	15	16	0.79	0.80	0.79
veg-oil	11	7	13	4	0.64	0.35	0.45
wool	0	0	0	0	ND	ND	ND
wpi	9	1	4	8	0.11	0.20	0.14
zinc	5	0	0	5	0.00	ND	0.00

Tablica 4.10 - Rezultati klasifikacije prema kategorijama, 5. test.

Kombinirani prikaz rezultata testova



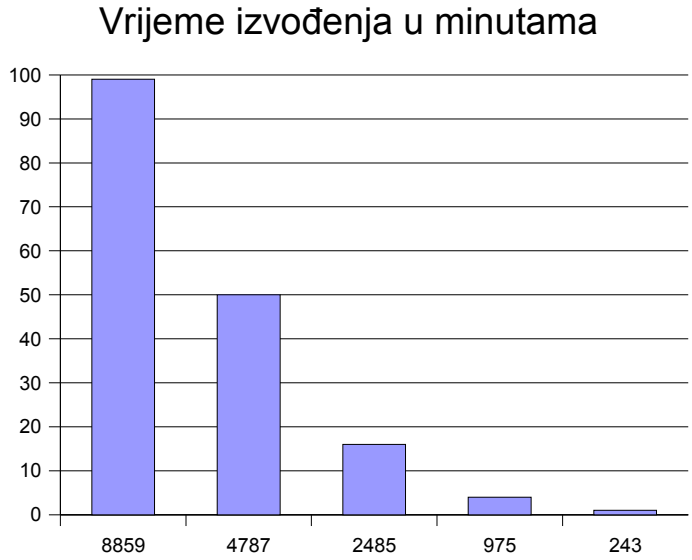
Slika 4.6 - Prikaz preciznosti, odaziva i F1 mjere u postocima u ovisnosti o broju značajki.



Slika 4.7 - Prikaz preciznosti, odaziva i F1 mjere u postocima u ovisnosti o broju značajki.

Kao što je u [3] naglašeno, rezultati mikroprosjeka i makroprosjeka su dosta različiti.

Razlog tome je što se u makroprosjeku vidi mogućnost točnog klasificiranja dokumenata koji imaju kategoriju koja je slabo zastupljena u skupu dokumenata za učenje.
Smanjivanjem broja atributa, odnosno riječi od kojih se grade vektori za klasifikaciju najviše se utječe na kategorije koje imaju mali broj dokumenata.
Rezultati testiranja nešto su bolji nego rezultati navedeni u [2] i [3], a razlog tome leži u drukčijem izboru dokumenta za učenje i testiranje.



Slika 4.8 - Vrijeme izvođenja u ovisnosti o broju atributa.

Zanimljivo je pogledati vrijeme trajanje izgradnje stabla odluke u ovisnosti o veličini vektora, odnosno broju riječi, prikazano na slici 4.3.
Iz slike je jasno vidljivo da se vrijeme naglo smanjuje sa smanjivanjem broja atributa, a uzevši u obzir rezultate klasifikacije, kao idealnu vrijednost mogao bi se uzeti broj atributa koji je 10 ili čak 25 puta manji od broja riječi u skupu za učenje.

4.2.2 Vjesnik set

Kao i na Reuters skupu i na Vjesnik skupu izvedeno je nekoliko testova sa različitim brojem značajki.

1. Test

Izbačene su sve riječi koje se pojavljuju u manje od 33 dokumenta.

Broj različitih riječi:	72555
Broj riječi nakon izbacivanja:	7168
Broj train dokumenata:	10000
Broj test dokumenata:	10000
Broj kategorija:	8

Veličina stabla prije obrezivanja:	2539	
Broj pogrešno klasificiranih dokumenata:	3501	(35 %)
Broj točno klasificiranih dokumenata:	6499	(65 %)

Veličina stabla nakon obrezivanja:	2317		
Broj pogrešno klasificiranih dokumenata:	3440	(34,4 %)	
Broj točno klasificiranih dokumenata:	6560	(65,6 %)	

	Preciznost	***Odaziv***	F_1
Mikroprosjek	0.66	0.66	0.66
Makroprosjek	0.65	0.66	0.65

Tablica 4.11- Rezultati klasifikacije za svih 8 kategorija, 1. test.

Kategorija	Br. dok.	TP	FP	FN	Odaziv	Preciznost	F_1
ck	1250	1018	306	232	0.81	0.77	0.79
go	1250	872	562	378	0.70	0.61	0.65
ku	1250	993	341	257	0.79	0.74	0.77
sp	1250	1150	146	100	0.92	0.89	0.90
td	1250	398	633	852	0.32	0.39	0.35
un	1250	531	691	719	0.42	0.43	0.43
vp	1250	846	393	404	0.68	0.68	0.68
zg	1250	752	368	498	0.60	0.67	0.63

Tablica 4.12 - Rezultati klasifikacije prema kategorijama, 1. test.

2. Test

Izbačene su sve riječi koje se pojavljuju u manje od 87 dokumenta.

Broj različitih riječi: 72555
Broj riječi nakon izbacivanja: 3579
Broj train dokumenata: 10000
Broj test dokumenata: 10000
Broj kategorija: 8

Veličina stabla prije obrezivanja: 2713
Broj pogrešno klasificiranih dokumenata: 3677 (36,8 %)
Broj točno klasificiranih dokumenata: 6323 (63,2 %)

Veličina stabla nakon obrezivanja: 2451
Broj pogrešno klasificiranih dokumenata: 3620 (36,2 %)
Broj točno klasificiranih dokumenata: 6380 (63,8 %)

	Preciznost	***Odaziv***	***F_1***
Mikroprosjek	0.64	0.64	0.64
Makroprosjek	0.63	0.64	0.63

Tablica 4.13 - Rezultati klasifikacije za svih 8 kategorija, 2. test.

Kategorija	Br. dok.	TP	FP	FN	Odaziv	Preciznost	F_1
ck	1250	991	292	259	0.79	0.77	0.78
go	1250	787	505	463	0.63	0.61	0.62
ku	1250	942	384	308	0.75	0.71	0.73
sp	1250	1153	158	97	0.92	0.88	0.90
td	1250	433	704	817	0.35	0.38	0.36
un	1250	490	753	760	0.39	0.39	0.39
vp	1250	841	390	409	0.67	0.68	0.68
zg	1250	743	434	507	0.59	0.63	0.61

Tablica 4.14 - Rezultati klasifikacije prema kategorijama, 2. test.

3. Test

Izbačene su sve riječi koje se pojavljuju u manje od 87 dokumenta.

Broj različitih riječi:	72555
Broj riječi nakon izbacivanja:	1811
Broj train dokumenata:	10000
Broj test dokumenata:	10000
Broj kategorija:	8

Veličina stabla prije obrezivanja:	2771	
Broj pogrešno klasificiranih dokumenata:	3816	(38,2 %)
Broj točno klasificiranih dokumenata:	6184	(61,8 %)

Veličina stabla nakon obrezivanja:	2379	
Broj pogrešno klasificiranih dokumenata:	3733	(37,3 %)
Broj točno klasificiranih dokumenata:	6267	(62,7 %)

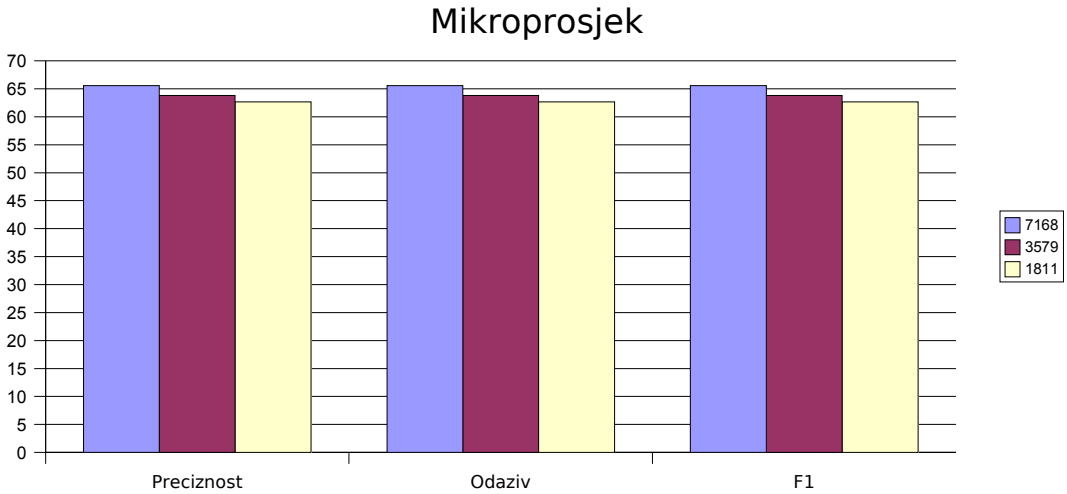
	Preciznost	***Odaziv***	***F_1***
Mikroprosjek	0.63	0.63	0.63
Makroprosjek	0.62	0.63	0.62

Tablica 4.15 - Rezultati klasifikacije za svih 8 kategorija, 3. test.

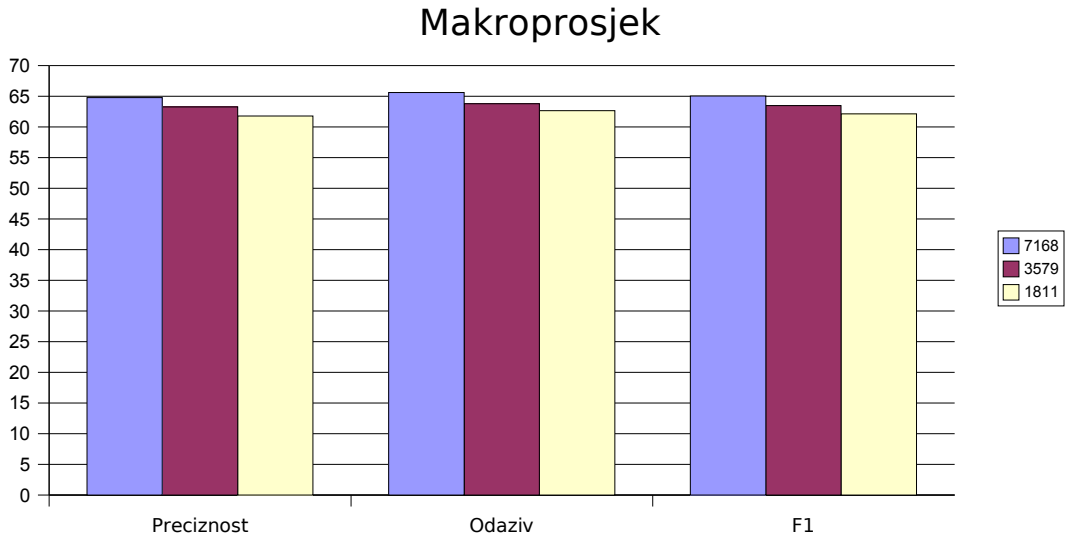
Kategorija	Br. dok.	TP	FP	FN	Odaziv	Preciznost	F_1
ck	1250	952	313	298	0.76	0.75	0.76
go	1250	790	501	460	0.63	0.61	0.62
ku	1250	968	411	282	0.77	0.70	0.74
sp	1250	1133	164	117	0.91	0.87	0.89
td	1250	424	670	826	0.34	0.39	0.36
un	1250	443	697	807	0.35	0.39	0.37
vp	1250	855	457	395	0.68	0.65	0.67
zg	1250	702	520	548	0.56	0.57	0.57

Tablica 4.16 - Rezultati klasifikacije prema kategorijama, 3. test.

Kombinirani prikaz rezultata testova



Slika 4.9 - Prikaz preciznosti, odaziva i F1 mjere u postocima u ovisnosti o broju značajki.



Slika 4.10 - Prikaz preciznosti, odaziva i F1 mjere u postocima u ovisnosti o broju značajki.

Za razliku od Reuters rezultata ovdje možemo vidjeti gotovo jednake iznose mikro i makroprosjeka. Razlog tome je jednaka raspodijela dokumenata po kategorijama.
Isto tako ovdje je primijećen dosta slabiji rezultat klasifikacije nego kod Reuters seta. Uzrok tome ponajprije treba tražiti u veličini skupa za

učenje. Stabla odluke stvorena iz velikih skupova sa velikim vektorom značajki imaju tendenciju jako dobro klasificirati dokumente iz skupa za učenje, odnosno javlja se problem overfittinga.

Slika 4.11 - Vrijeme izvođenja u ovisnosti o broju atributa

Kao i kod Reuters skupa i ovdje je uočljiva vremenska zahtjevnost izgradnje stabla odluke o broju atributa.

5 Zaključak

Zadatak ovog diplomskog rada je izgradnja sustava za klasifikaciju teksta temeljenog na C4.5 algoritmu.
Finalni rezultat je aplikacija, namijenjena radu u grafičkom sučelju bilo da je riječ o Windows ili GNU/Linux operacijskim sustavima, koja vrši parsiranje ulaznih podataka, izbor značajki, samu klasifikaciju i prikaz rezultata.
Kao i uvijek, neke stvari mogle bi se poboljšati, a prva stvar koja bi bila zanimljiva za dodati su drugi načini izbora značajki. Samom algoritmu klasifikacije nema se što nadodati, a vrlo zanimljivo bi bilo vidjeti usporedbu performansi klasifikacije C4.5 algoritma i njegovog komercijalnog nasljednika C5.0/See5 [16].

Pregled rezultata na dva dosta različita skupa za učenje može dati odgovor na pitanje za koju vrsta skupova je C4.5 algoritam primjenjiv kao klasifikator.
Vidljivo je da je algoritam skloniji skupovima sa manje dokumenata za učenje, odnosno manjim brojim riječi, što se osim u točnosti klasifikacije može očitati i u vremenu izvođenja, dok kod velikih skupova dolazi do problema prevelike prilagođenosti skupu za učenje.

6 Literatura

[1] Quinlan, J.R. (1993), “C4.5: programs for machine learning”, Morgan Kaufmann Publisher, Inc.

[2] Joachims, T. (2002), “Learning to classify text using support vector machines”, Kluwer Academic Publishers

[3] Sebastiani, F. (2002), “Machine Learning in Automated Text Categorization”, ACM Computing Surveys, Vol. 34, No. 1, March 2002, pp. 1-47., URL: http://faure.iei.pi.cnr.it/~fabrizio/Publications/ACMCS02.pdf

[4] Sebastian, F. (1999), “A Tutorial on Automated Text Categorisation”, Proceedings of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence, Buenos Aires, AR, 1999, pp. 7-35., URL: http://faure.iei.pi.cnr.it/~fabrizio/Publications/ASAI99.pdf

[5] Yang, Y.,Pedersen, J.O. (1997), “A Comparative Study on Feature Selection in Text Categorization”, Proceedings of ICML-97, 14th International Conference on Machine Learning, URL: http://citeseer.ist.psu.edu/yang97comparative.html

[6] Izvor za Reuters-21578 skup: http://www.daviddlewis.com/resources/testcollections/reuters21578/

[7] Eckel, B. (2000), “Thinking in C++, Volume 1”, URL: http://mindview.net/Books/TICPP/ThinkingInCPP2e.html

[8] Eckel, B., Allison, C. (2003), “Thinking in C++, Volume 2”, URL: http://mindview.net/Books/TICPP/ThinkingInCPP2e.html

[9] GNOME Desktop Suite and development platform, URL: http://www.gnome.org/

[10] GTK+ multiplatform toolkit for creating graphical user interfaces, URL: http://www.gtk.org/

[11] GTKMM C++ interface for GTK+, URL: http://www.gtkmm.org/

[12] Debian GNU/Linux operacijski sustav, URL: http://www.debian.org/

[13] Porter Stemming Algorithm,
http://www.tartarus.org/~martin/PorterStemmer/

[14] Izvor za C4.5 algoritam: URL: http://www.cse.unsw.edu.au/~quinlan/

[15] Cygwin - Linux-like environment for Windows, URL:
http://www.cygwin.com/

[16] Rulequest Research, Data Minining Tools See5 and C5.0, URL:
http://www.rulequest.com/see5-info.html

[17] Izvor za Vjesnik skup, URL: http://www.zemris.fer.hr/~jan/textm/