



УНИВЕРЗИТЕТ У НОВОМ САДУ
ПРИРОДНО-МАТЕМАТИЧКИ ФАКУЛТЕТ
ДЕПАРТМАН ЗА МАТЕМАТИКУ И
ИНФОРМАТИКУ



Mašinsko učenje: Određivanje jezika dokumenta

Projekat iz predmeta Veštačka inteligencija

Ime i prezime: Tamara Gogić

Sadržaj

1. Uvod.....	3
2. Priprema podataka.....	4
3. Analiza podataka.....	8
3.1 O SMO klasifikatoru.....	8
3.2 Primena SMO klasifikatora uz korišćenje CharacterNGramTokenizer-a.....	8
3.3 Primena SMO klasifikatora uz korišćenje NGramTokenizer-a.....	11
3.4 Primena SMO klasifikatora uz korišćenje WordTokenizer-a.....	12
3.5 Primena SMO klasifikatora uz korišćenje AlphabethicTokenizer-a.....	13
4. Zaključak.....	14

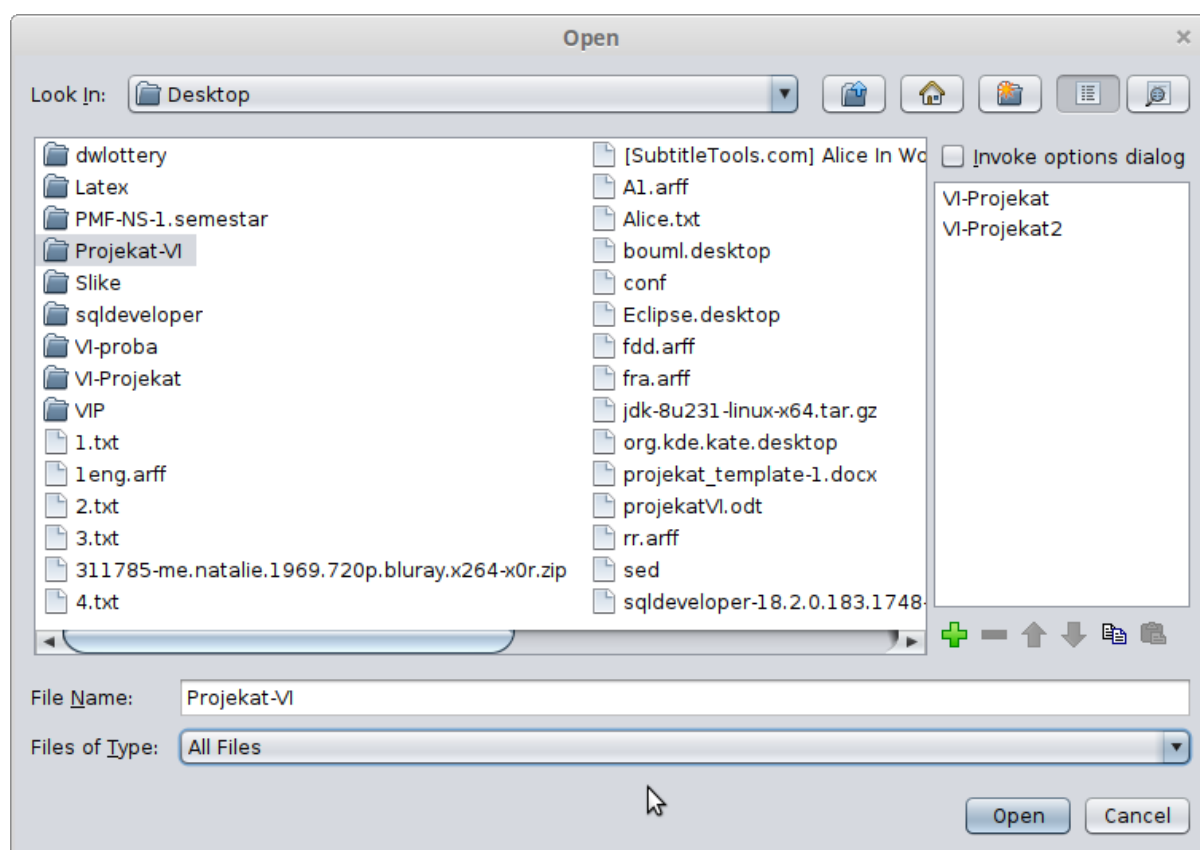
1. Uvod

U ovom projektu prikazano je rešavanje problema određivanja jezika dokumenta na osnovu sadržine tekstualnog dokumenta korišćenjem metoda mašinskog učenja (klasifikacije). Korišćeni podaci su preuzeti sa sajta <https://rs.titlovi.com/?v=1>. U pitanju su prevodi različitih filmova na 5 različitih jezika. Za rešavanje i analizu ovog problema korišćen je alat Weka.

2. Priprema podataka

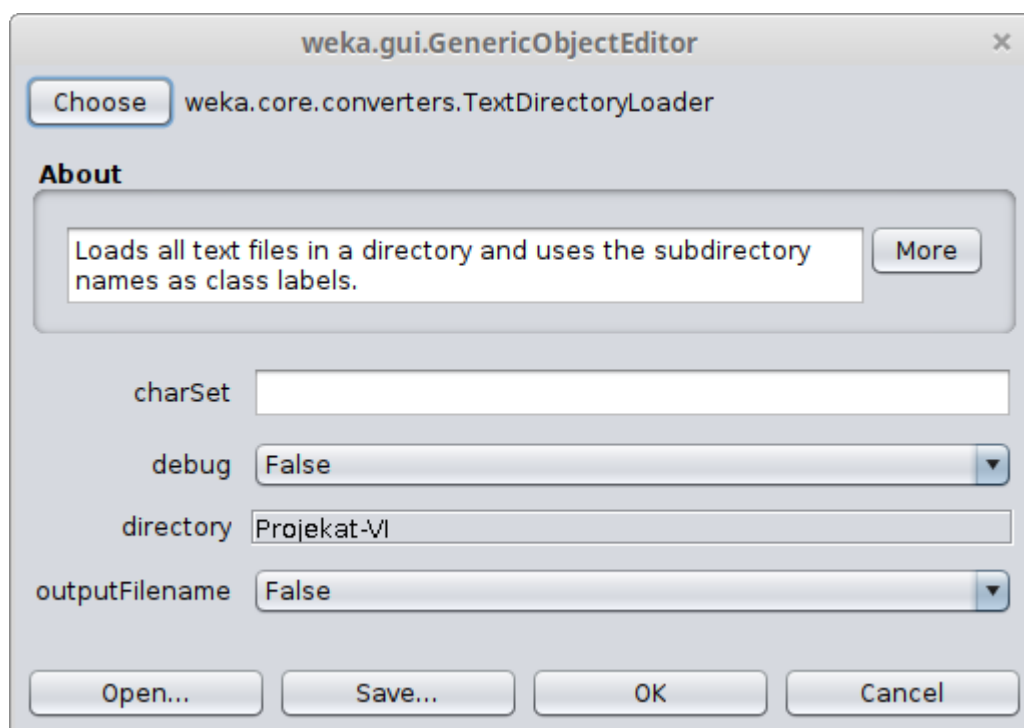
Korišćeni podaci preuzeti su kao *.srt* fajlovi. Za konvertovanje ovih fajlova u *.txt* format korišćen je online konverter <https://subtitletools.com/>. Zatim je od tih tekstualnih fajlova konstruisan *.arff* fajl uz pomoć ugrađenog konvertera u alatu Weka *TextDirectoryLoader*. ARFF format je osnovni format za predstavljanje tekstualnog zapisa skupa korišćenih podataka, koji na početku ima zaglavlje (header) s nazivom relacije i opisom atributa, a u nastavku obučavajuće primere (instance) u CSV (Comma Separated Values) formatu. U nastavku je opisan detaljan postupak konstruisanja *.arff* fajla.

U okruženju *Weka Explorer* potrebno je odabrati polje *Open file*, nakon čega će se otvoriti prozor (Slika 2.1) u kom je potrebno odabrati folder sa potrebnim podacima. U tom folderu tekstualni fajlovi su organizovani u poddirektorijume čija imena predstavljaju klase. Takva organizacija fajlova je neophodna za uspešno korišćenje gore navedenog konvertera.



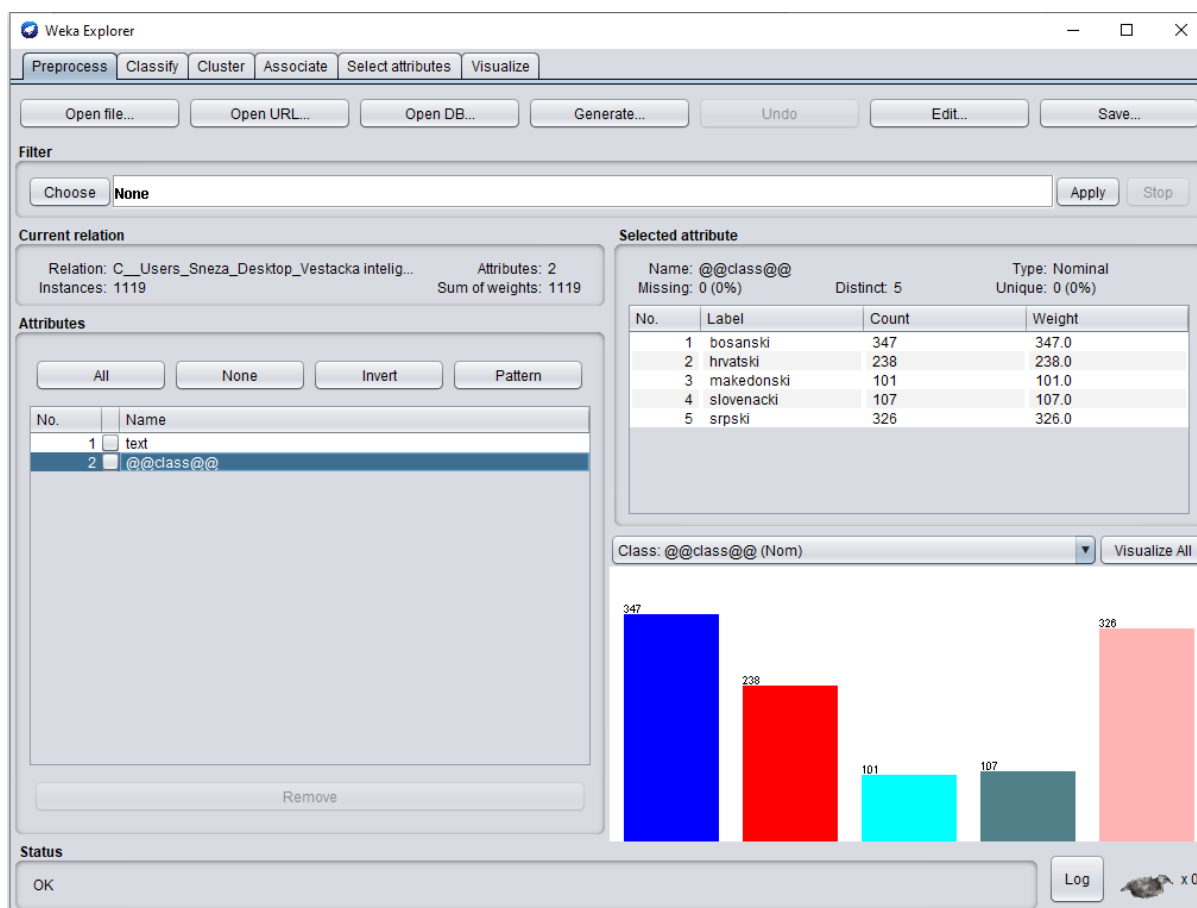
Slika 2.1

Nakon toga će se otvoriti prozor *weka.gui.GenericObjectEditor* (Slika 2.2). Klikom na dugme *Choose* otvoriće se padajući meni iz kog je potrebno odabrati konverter *TextDirectoryLoader*. U polju *directory* je potrebno odabrati folder sa podacima.



Slika 2.2

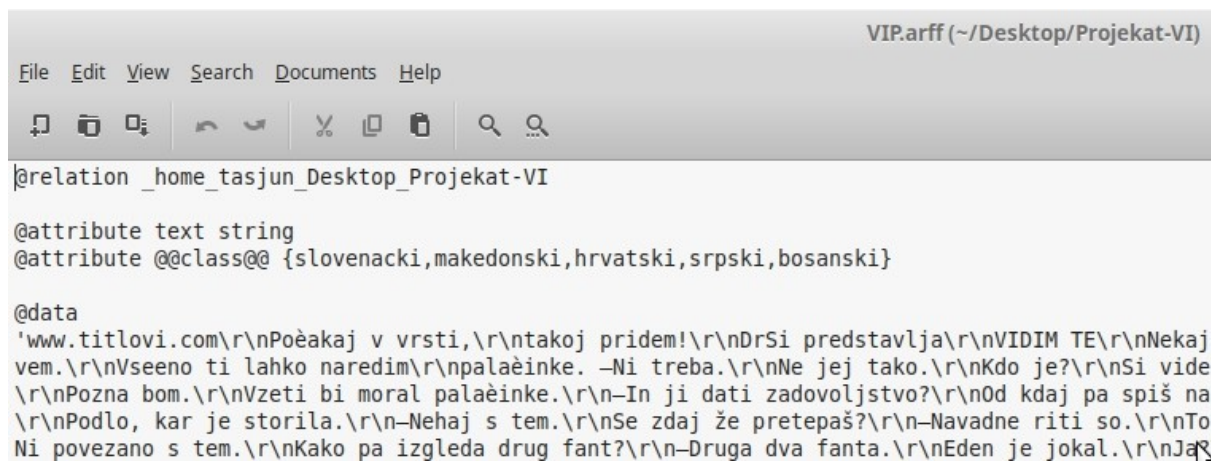
Nakon klika na dugme OK Weka će izgenerisati traženi *.arff* fajl i prikazati podatke (Slika 2.3).



Slika 2.3

U odeljku *Preprocess* nalaze se osnovne informacije o *.arff* fajlu kao što su nazivi i broj klasa koje sadrži, kao i koliko instanci koja klasa sadrži. Ukupan broj instanci je 1119 od kojih klasi slovenački pripada 107, klasi makedonski 101, klasi hrvatski 238, klasi srpski 326 i klasi bosanski 347. Sa dijagrama se vidi da podaci nisu izbalansirani. Nebalansiranost bolje oslikava realnu situaciju.

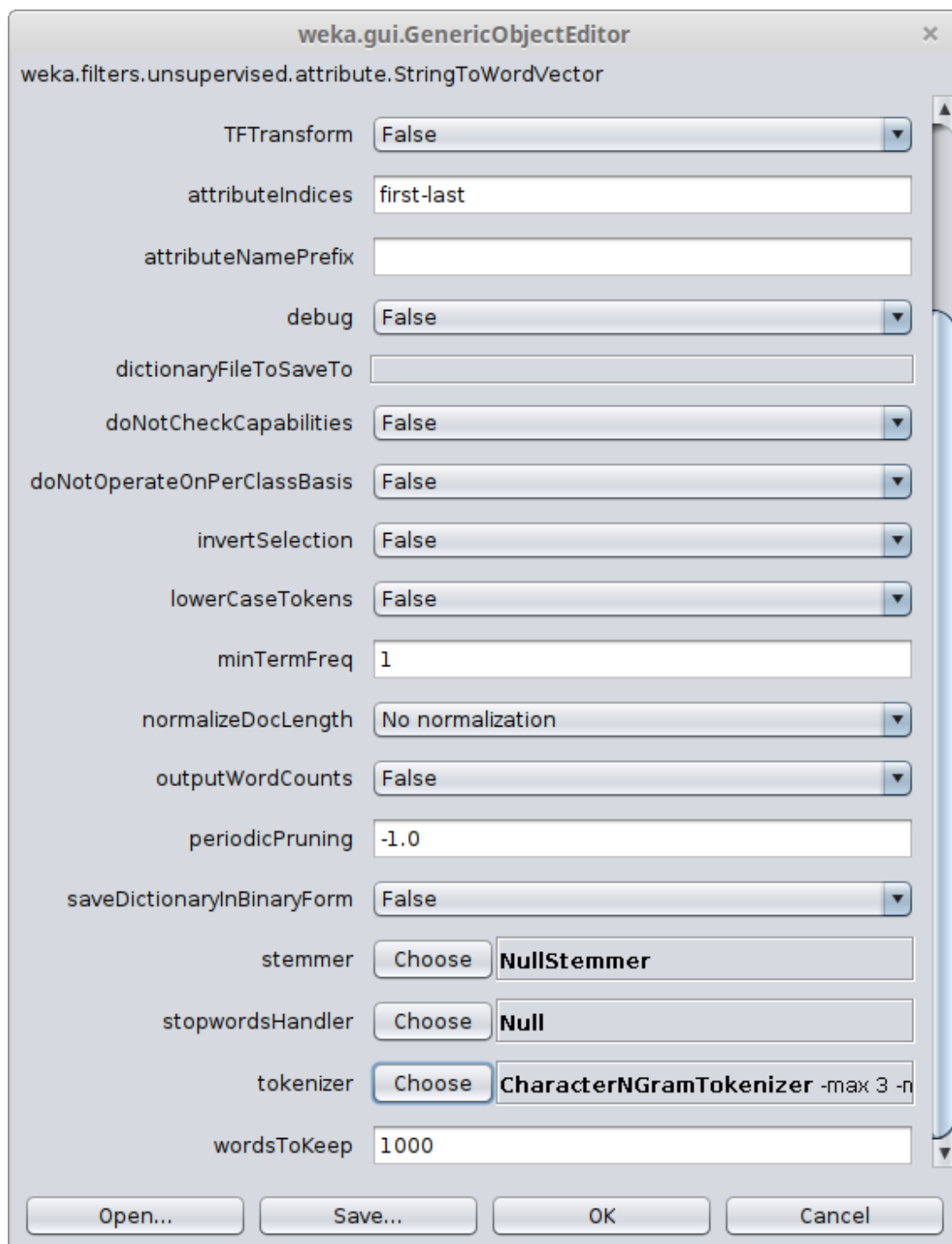
Klikom na dugme *Save* možemo sačuvati izgenerisani *.arff* fajl. Ovaj fajl (Slika 2.4) sadrži dva atributa. Prvi predstavlja tekst dokumenta tipa string, a drugi predstavlja klasu tj. jezik tog dokumenta i taj atribut je nominalni sa mogućim vrednostima: srpski, bosanski, hrvatski, makedonski, slovenački. Ispod ključne reči *@data* navedene su instance skupa podataka. Jedna instanca predstavlja vrednosti atributa odvojene zarezom.



```
VIP.arff (~/Desktop/Projekat-VI)
File Edit View Search Documents Help
[Icons]
@relation _home_tasjun_Desktop_Projekat-VI
@attribute text string
@attribute @@class@@ {slovenacki,makedonski,hrvatski,srpski,bosanski}
@data
'www.titlovi.com\r\nPoèakaj v vrsti,\r\ntakoj pridem!\r\nDrSi predstavlja\r\nVIDIM TE\r\nNekaj
vem.\r\nVseeno ti lahko naredim\r\npalaèinke. -Ni treba.\r\nNe jej tako.\r\nKdo je?\r\nSi vide
\r\nPozna bom.\r\nVzeti bi moral palaèinke.\r\n-In ji dati zadovoljstvo?\r\nOd kdaj pa spiš na
\r\nPodlo, kar je storila.\r\n-Nehaj s tem.\r\nSe zdaj že pretepaš?\r\n-Navadne riti so.\r\nTo
Ni povezano s tem.\r\nKako pa izgleda drug fant?\r\n-Druga dva fanta.\r\nEden je jokal.\r\nJaž
```

Slika 2.4

Pošto klasifikator *SMO* ne radi sa podacima tipa string potrebno je uz pomoć filtera *StringToWordVector* transformisati iste u vektore. Klikom na dugme *Choose* potrebno je odabrati *filters* zatim *unsupervised*, zatim *attributes*, a onda iz liste filtera odabrati *StringToWordVector*. Klikom na odabrani filter otvoriće se prozor na slici Slika 2.5. Ovaj filter konvertuje string attribute u skup atributa (numeričkog tipa) koji predstavljaju pojave reči (u zavisnosti od odabranog tokenizatora) u tekstu sadržanom u stringovima koji predstavljaju vrednosti atributa. Postoje četiri tokenizatora: *AlphabeticTokenizer*, *WordTokenizer*, *NGramTokenizer* i *CharacterNGramTokenizer*. Tokenizacija je izdvajanje tokena iz teksta. Svaki tokenizator ima svoj definisani način za određivanje tokena. Ukoliko se odabere *AlphabeticTokenizer* tokeni će biti samo neprekidne sekvence simbola engleskog alfabeta. Ukoliko se odabere *WordTokenizer* tokeni će biti reči iz teksta koje se prepoznaju na osnovu predefinisanih delimitera. Po default-u delimiteri su: „,;'"()?!. Ukoliko se odabere *NGramTokenizer* tokeni će biti konstruisani na osnovu kombinacija susednih reči, a ukoliko se odabere *CharacterNGramTokenizer* tokeni će biti konstruisani na osnovu kombinacija susednih karaktera. Pošto *NGramTokenizer* radi sa rečima tu je takođe moguće definisati delimitere. Za ova dva tokenizatora je moguće definisati koje dužine tokena će se razmatrati. Ta dužina se meri u gram-ima s tim što za *NGramTokenizer* gram predstavlja reč, dok za *CharacterNGramTokenizer* gram predstavlja jedan karakter. Moguće je podesiti *NGramMaxSize* (po default-u 3) i *NGramMinSize* (po default-u 1) koji predstavljaju maksimalan i minimalan broj gram-a od kojih token može biti konstruisan. Tokeni dužine jednog gram-a se nazivaju *unigram*, dva gram-a *bigram*, tri gram-a *trigram*, itd. Svaki od tokenizatora se može odabrati klikom na dugme *Choose* kod opcije *tokenizer* u prozoru na slici Slika 2.5. Pored opcije *tokenizer* moguće je podešavanje i nekih drugih parametara, ali koristiće se njihove default vrednosti u ovom projektu. Ukoliko se dva puta klikne na odabran tokenizator otvoriće se prozor u kom se mogu menjati parametri. Kada se odabere tokenizator, klikom na dugme *Apply* primenjuje se filter na skup podataka i shodno odabranom tokenizatoru, dobiće se novi atribute.



Slika 2.5

3. Analiza podataka

3.1 O SMO klasifikatoru

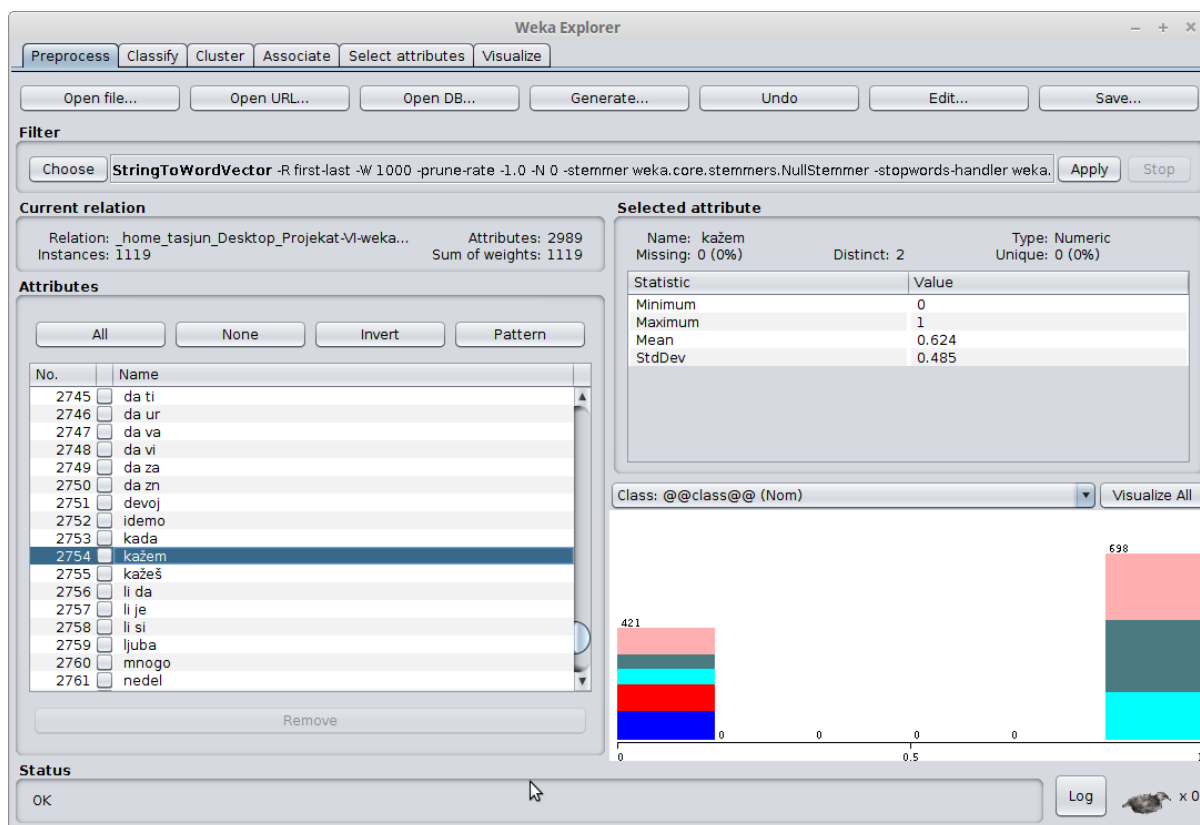
Support Vector Machines (SVM) razvijene su za probleme binarne klasifikacije, mada su proširenja tehnike napravljena da podrže probleme klasifikacije više klasa i problema regresije. SVM je razvijen za numeričke ulazne podatke, mada će automatski pretvarati nominalne vrednosti u numeričke vrednosti. Ulazni podaci se takođe normalizuju pre upotrebe. SVM radi tako što pronalazi liniju koja najbolje razdvaja podatke u dve grupe. Ovo se vrši pomoću procesa optimizacije koji uzima u obzir samo one podatke u skupu podataka za trening koji su najbliži liniji koja najbolje odvaja klase. U skoro svim problemima, ne može se nacrtati linija radi urednog odvajanja klasa, pa se oko linije dodaje margina kako bi se ublažilo ograničenje, dozvoljavajući da se neki slučajevi pogrešno klasifikuju, ali omogućuju bolji opšti rezultat. Konačno, nekoliko skupova podataka može se odvojiti samo ravnom linijom. Ponekad je potrebno obeležiti liniju sa zavojima ili čak poligonalnim oblastima. To se postiže SVM projekcijom podataka u prostor veće dimenzije kako bi se crtale linije i predviđale. Različita jezgra mogu se koristiti za kontrolu projekcije i količine fleksibilnosti u odvajanju klasa. U ovom projektu će se koristiti SMO algoritam koji predstavlja specifični algoritam efikasne optimizacije koji se koristi u okviru SVM implementacije, a koji se zalaže za sekvencijalnu minimalnu optimizaciju. SMO koristi heuristiku da bi podelio problem treniranja podataka na manje probleme koji se mogu analitički rešiti. Da li dobro funkcioniše ili ne, u velikoj meri zavisi od pretpostavki koje stoje iza heuristike. Obično u značajnoj meri ubrzava treniranje podataka.

U alatu Weka, u odeljku *Classify*, klikom na dugme *Choose* prikazaće se klasifikatori organizovani u podfoldere foldera *classifiers*. SMO klasifikator može se naći u podfolderu *functions*. Način testiranja koji će se koristiti prilikom učenja SMO klasifikatora je kros-validacija (cross-validation). Način testiranja se takođe može odabrati u odeljku *Classify* pod delom *Test options*. Broj foldova koji će se koristiti je broj po default-u tj. 10.

Kros-validacija u prvom koraku deli skup podataka u k podskupova približno iste veličine. U drugom koraku, jedan podskup se izdvoji za testiranje, ostali za obučavanje. Postupak se ponavlja za svih k podskupova. Standardni metod je stratifikovana validacija u 10 foldova. U praksi se pokazalo da ovaj izbor daje najbolje rezultate i za to postoje neki teorijski dokazi. Stratifikovana kros-validacija obezbeđuje da se u svakom delu (fold) nalazi odgovarajuća proporcija podataka.

3.2 Primena SMO klasifikatora uz korišćenje CharacterNGramTokenizer-a

CharacterNGramTokenizer radi tako što se zadaju NGramMinSize i NGramMaxSize parametri. Na osnovu vrednosti tih parametara se konstruišu tokeni dužina koje su u tom opsegu. Na primer, ukoliko imamo string "Quick Fox" i odaberemo parametre NGramMinSize=1 i NGramMaxSize=2 tokeni će biti: [Q, Qu, u, ui, i, ic, c, ck, k, "k ", " ", " F", F, Fo, o, ox, x]. CharacterNGramTokenizer daje najbolje rezultate za parametre NGramMinSize=6 i NGramMaxSize=6. Na slici (Slika 3.2.1) prikazan je rezultat primene ovih parametara. Filtriranjem od jednog string atributa dobijeno je 2989 numerickih atributa. U delu *Attributes* može se videti lista svih tih atributa. Ukoliko se klikne na neki od njih, u delu *Selected attribute* mogu se videti minimalna (Minimum), maksimalna (Maximum), srednja (Mean) vrednost i standardna devijacija (StdDev) tog atributa. Sa dijagrama se može primetiti da je token ' kažem' zastupljen u 698 instanci (u jezicima bosanski, hrvatski i srpski), dok se ne pojavljuje u 421 instanci (u jezicima slovenački, makedonski, bosanski, hrvatski i srpski).



Slika 3.2.1

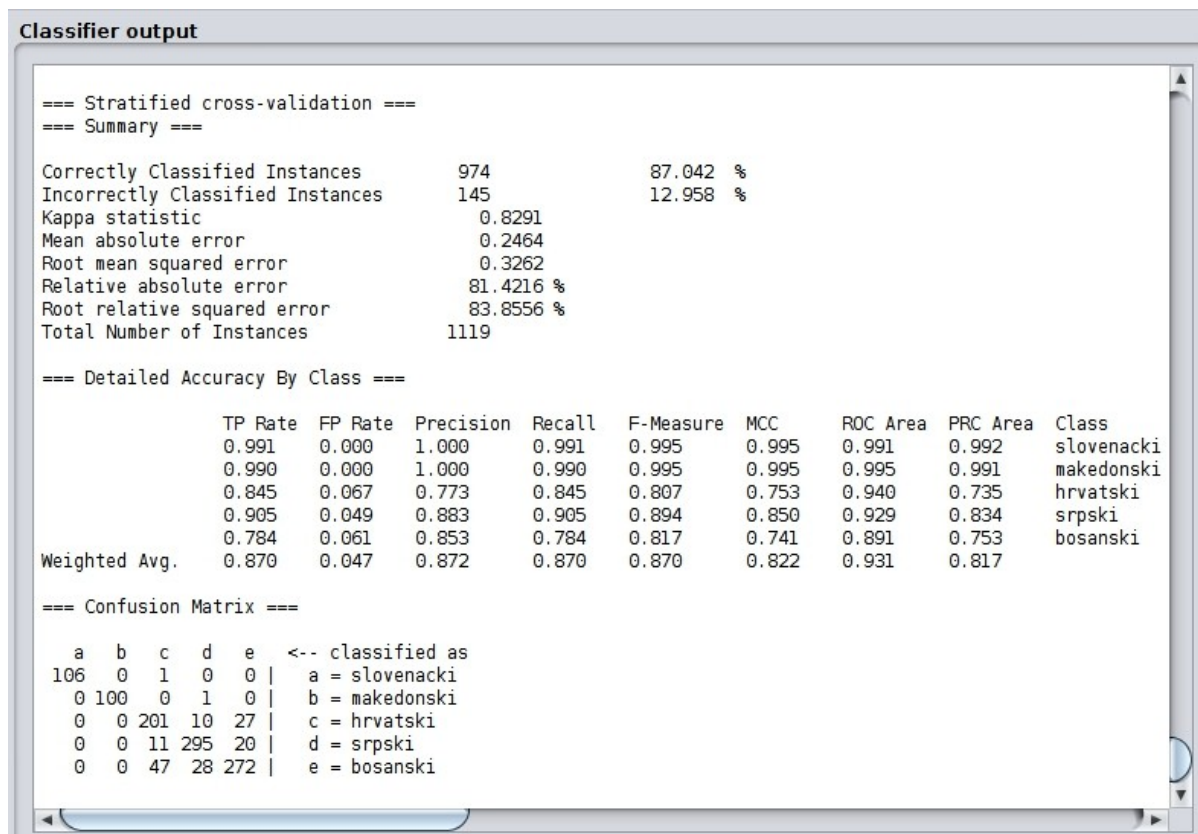
Nakon ovakvog filtriranja podataka, nad njima se poziva SMO klasifikator sa default parametrima koji daje rezultate na slici Slika 3.2.2. Tu se vidi da je procenat tačno klasifikovanih instanci (**Correctly Classified Instances**) 87,042%, što znači da je 974 od 1119 instanci klasifikovano tačno u onu klasu kojoj i pripada. Netačno klasifikovanih instanci (**Incorrectly Classified Instances**) ima 145 (12,958%). Vrednost **Kappa statistic** je 0,829. To je metrika koja upoređuje tačnost korišćenog klasifikatora i novog klasifikatora generisanog na osnovu random raspoređenih vrednosti u matrici konfuzije. U matrici konfuzije novog klasifikatora mora se sačuvati odnos vrednosti zbira kolona i redova iz matrice konfuzije korišćenog klasifikatora. Srednja apsolutna greška (**Mean absolute error**) iznosi 0,2464. Ona predstavlja prosek apsolutnih grešaka svih instanci. Koren srednje kvadratne greške (**Root mean squared error**) je 0,3262. Ova greška predstavlja standardnu devijaciju uzorka razlika između predviđenih vrednosti i posmatranih vrednosti. Relativna apsolutna greška (**Relative absolute error**) je 81,4216%. Računa se tako što se Srednja apsolutna greška podeli sa greškom ZeroR klasifikatora (klasifikator koji ignoriše sva predviđanja i jednostavno odabere najčešću vrednost klase). Koren relativne kvadratne greške (**Root relative squared error**) je 83,8556% i predstavlja uzimanje kvadratnog korena relativne kvadratne greške i smanjivanje greške na istu dimenziju kao i količina koja se predviđa.

U delu *Detailed Accuracy By Class* mogu se videti vrednosti za Precision i Recall, kao i za TP Rate i FP Rate za svaku klasu posebno, kao i za ceo skup podataka. **Precision** predstavlja preciznost tj. koji deo od svih instanci koje su prepoznate kao određena klasa stvarno pripadaju toj klasi ($TP/(TP+FP)$). **Recall** predstavlja koji deo od svih instanci koje pripadaju nekoj klasi su prepoznate kao ta klasa ($TP/(TP+FN)$). **TP Rate** predstavlja vrednost koja oslikava količinu instanci određene klase koje su tačno klasifikovane. **FP Rate** predstavlja količinu instanci neke klase koje su pogrešno klasifikovane.

Na osnovu ovih vrednosti vidi se da su instance klase makedonski i slovenački bolje klasifikovane u odnosu na ostale, dok su instance klase hrvatski najlošije klasifikovane.

U delu *Confusion Matrix* nalazi se Matrica konfuzije koja prikazuje kako je klasifikator rasporedio instance klase, tj. koliko instanci je tačno rasporedio, a koliko je smestio u pogrešne klase. Svaki red označava kojoj klasi instance pripadaju, dok kolona označava u koju klasu je klasifikator smestio koliko instanci. Sa matrice vidimo da je klasifikator od 107 instanci klase slovenački 106 klasifikovao tačno (u

klasu slovenacki), dok je jednu instancu klasifikovao pogrešno, u klasu hrvatski. Time vidimo da je ovu klasu skoro savršeno klasifikovao, kao i klasu makedonski, od 101 instance, tačno je klasifikovao 100 dok je jednu klasifikovao kao srpski. Može se primetiti da su lošije klasifikovane klase hrvatski, srpski i bosanski. Od 238 instance klase hrvatski 201 je klasifikovano tačno, a 37 netačno (10 kao srpski i 27 kao bosanski). Od 326 instanci klase srpski 295 je klasifikovano tačno, dok je 31 klasifikovano netačno (11 kao hrvatski, i 20 kao bosanski). Od 347 instanci klase bosanski 272 je klasifikovano tačno, dok je 75 klasifikovano netačno (47 kao hrvatski i 28 kao srpski). Iako u skupu podataka ima manje instanci klasa slovenacki i makedonski u odnosu na ostale klase, one imaju veću preciznost jer se radi o jezicima koji imaju manji stepen sličnosti u odnosu na ostale klase (jezike). Iako jezici srpski, bosanski i hrvatski imaju više instanci u skupu podataka pokazalo se da su lošije klasifikovani zato što su ti jezici dosta slični pa ih je klasifikator u određenim instancama “mešao”. Iz ovoga se može zaključiti da veliki broj instanci klase ne garantuje uvek dobre rezultate.



Slika 3.2.2

U narednim tabelama prikazani su rezultati prilikom testiranja SMO klasifikatora sa različitim parametrima za CharacterNGramTokenizer. **min-max** predstavlja NGramMinSize i NGramMaxSize parametre tokenizatora, a **CCI** predstavlja Correctly Classified Instances. Najbolji rezultati su dobijeni korišćenjem vrednosti 6-6, 5-5 i 4-4 za min-max.

min-max	1-1	1-2	1-3	1-4	1-5	1-6	2-2
CCI (u %)	65.8624	82.7525	81.1439	80.8758	82.8418	82.6631	83.9142

min-max	2-3	2-4	2-5	2-6	3-3	3-4	3-5
CCI (u %)	78.9991	82.0375	81.5013	81.9482	82.5737	81.5907	81.1439

min-max	3-6	4-4	4-5	4-6	5-5	5-6	6-6
CCI (u %)	81.9482	86.5952	83.6461	83.9142	86.5058	85.2547	87.042

3.3 Primena SMO klasifikatora uz korišćenje NGramTokenizer-a

NGramTokenizer radi tako što se zadaju NGramMinSize i NGramMaxSize parametri. Na osnovu vrednosti tih parametara se konstruišu tokeni dužina koje su u tom opsegu. Za razliku od CharacterNGramTokenizer-a gde se u obzir uzimaju karakteri, NGramTokenizer će posmatrati reči. Kao parametar se mogu navesti i delimiteri. Na primer, ukoliko imamo string "Quick Fox" i odaberemo parametre NGramMinSize=1 i NGramMaxSize=2 tokeni će biti: [Quick, Fox, Quick Fox]. U ovom projektu procenat tačno klasifikovanih instanci koji se dobio na osnovu parametara NGramMinSize=1 i NGramMaxSize=3 je jednak procentu tačno klasifikovanih instanci dobijenom korišćenjem CharacterNGramTokenizer-a za vrednosti NGramMinSize=6 i NGramMaxSize=6 što su najbolji rezultati za korišćeni skup podataka. Na slici (Slika 3.3.1) prikazan je rezultat primene ovih parametara.

Classifier output

```

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      974           87.042 %
Incorrectly Classified Instances    145           12.958 %
Kappa statistic                    0.829
Mean absolute error                 0.2463
Root mean squared error             0.3259
Relative absolute error             81.3626 %
Root relative squared error         83.7766 %
Total Number of Instances          1119

=== Detailed Accuracy By Class ===
```

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.991	0.000	1.000	0.991	0.995	0.995	0.991	0.992	slovenacki
	0.990	0.000	1.000	0.990	0.995	0.995	1.000	0.999	makedonski
	0.840	0.065	0.778	0.840	0.808	0.755	0.924	0.725	hrvatski
	0.905	0.052	0.878	0.905	0.891	0.846	0.941	0.835	srpski
	0.787	0.061	0.853	0.787	0.819	0.743	0.897	0.757	bosanski
Weighted Avg.	0.870	0.048	0.872	0.870	0.870	0.822	0.934	0.817	

```

=== Confusion Matrix ===

 a  b  c  d  e  <-- classified as
106  0  0  1  0 | a = slovenacki
  0 100  0  1  0 | b = makedonski
  0  0 200  8 30 | c = hrvatski
  0  0 14 295 17 | d = srpski
  0  0 43 31 273 | e = bosanski

```

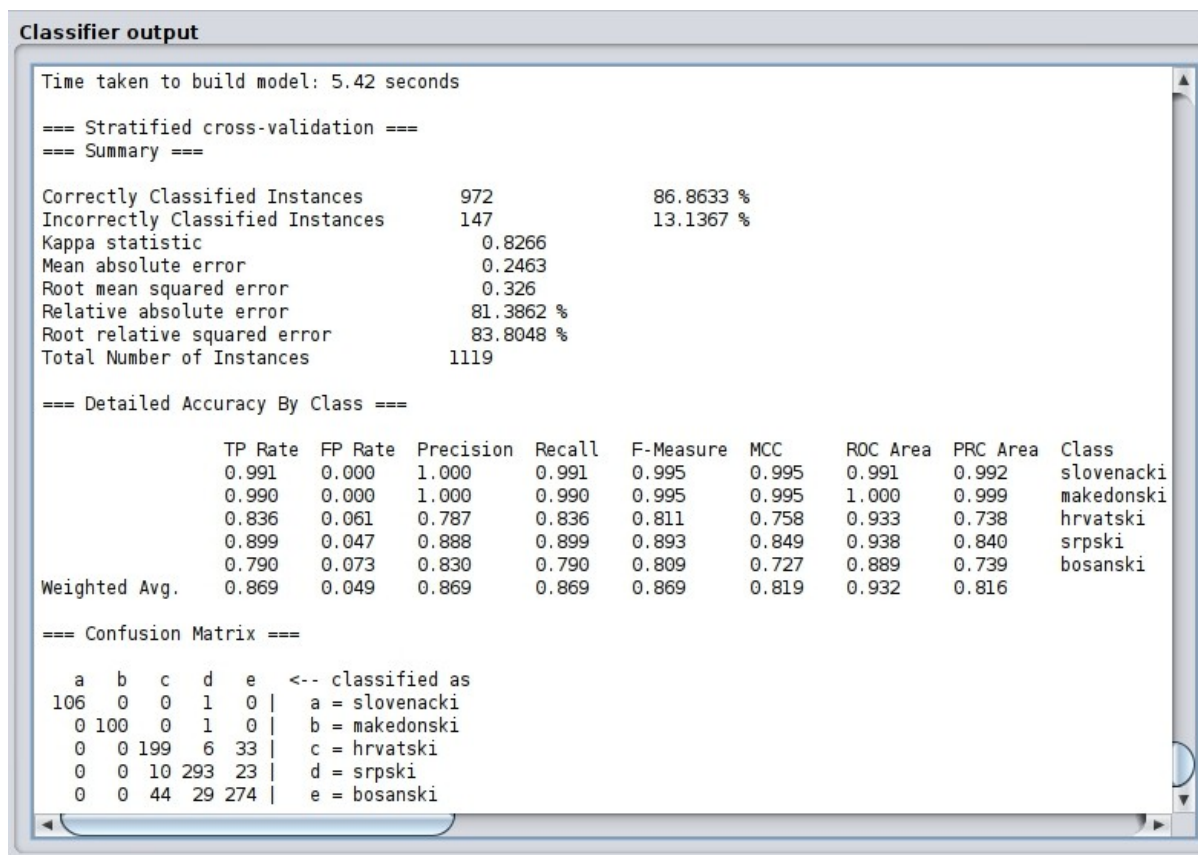
Slika 3.3.1

Tačno klasifikovanih instanci ima 974 (87,042%), dok je netačno klasifikovano 145 (12,958%) instanci. Kappa statistic iznosi 0,829, Mean absolute error je 0,2463, Root mean squared error je 0,3259, Relative absolute error je 81,3626% i Root relative squared error je 83,7766%. Detailed Accuracy By Class deo je sličan kao kod korišćenja CharacterNGramTokenizer-a. U matrici konfuzije, od 107 instanci klase slovenački 106 klasifikovao tačno (u klasu slovenački), dok je jednu instancu klasifikovao pogrešno, u klasu srpski. Time vidimo da je ovu klasu skoro savršeno klasifikovao, kao i klasu makedonski, od 101 instance, tačno je klasifikovao 100 dok je jednu klasifikovao kao srpski. Od 238 instanci klase hrvatski 200 je klasifikovano tačno, a 38 netačno (8 kao srpski i 30 kao bosanski). Od 326 instanci klase srpski 295 je klasifikovano tačno, dok je 31 klasifikovano netačno (14 kao hrvatski, i 17 kao bosanski). Od 347 instanci klase bosanski 273 je

klasifikovano tačno, dok je 74 klasifikovano netačno (43 kao hrvatski i 31 kao srpski). Zaključci su slični kao za rezultate kod korišćenja CharacterNGramTokenizer-a.

3.4 Primena SMO klasifikatora uz korišćenje WordTokenizer-a

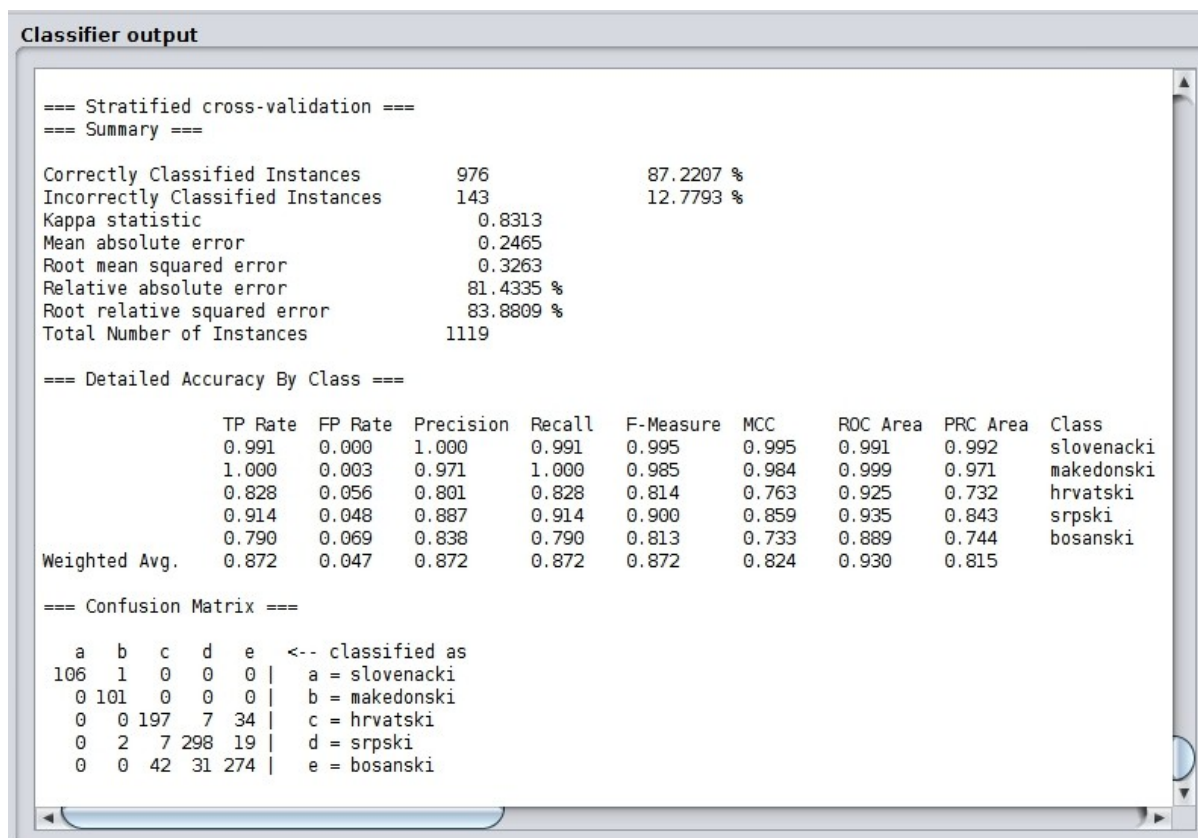
WordTokenizer radi tako što konstruiše tokene na osnovu reči iz teksta koje prepoznaje na osnovu predefinisanih delimitera. Po default-u delimiteri su: „;'"()?! To nam daje rezultate koji su prikazani na slici Slika 3.4.1.



Slika 3.4.1

Tačno klasifikovanih instanci ima 972 (86,8633%), dok je netačno klasifikovano 147 (13,1367%) instanci. Kappa statistic iznosi 0,8266, Mean absolute error je 0,2463, Root mean squared error je 0,326, Relative absolute error je 81,3862% i Root relative squared error je 83,8048%. U matrici konfuzije, od 107 instanci klase slovenački 106 klasifikovao tačno (u klasu slovenački), dok je jednu instancu klasifikovao pogrešno, u klasu srpski. Time vidimo da je ovu klasu skoro savršeno klasifikovao, kao i klasu makedonski, od 101 instance, tačno je klasifikovao 100 dok je jednu klasifikovao kao srpski. Od 238 instanci klase hrvatski 199 je klasifikovano tačno, a 39 netačno (6 kao srpski i 33 kao bosanski). Od 326 instanci klase srpski 293 je klasifikovano tačno, dok je 33 klasifikovano netačno (10 kao hrvatski, i 23 kao bosanski). Od 347 instanci klase bosanski 274 je klasifikovano tačno, dok je 73 klasifikovano netačno (44 kao hrvatski i 29 kao srpski).

3.5 Primena SMO klasifikatora uz korišćenje AlphabeticTokenizer-a



Slika 3.5.1

Na slici Slika 3.5.1 prikazani su statistički podaci o rezultatima testiranja SMO klasifikatora kada je za tokenizaciju korišćen AlphabeticTokenizer.

Tačno klasifikovanih instanci ima 976 (87,2207%), dok je netačno klasifikovano 143 (12,7793%) instanci. Kappa statistic iznosi 0,8313, Mean absolute error je 0,2465, Root mean squared error je 0,3263, Relative absolute error je 81,4335% i Root relative squared error je 83,8809%. U matrici konfuzije, od 107 instanci klase slovenački 106 klasifikovao tačno (u klasu slovenački), dok je jednu instancu klasifikovao pogrešno, u klasu makedonski. Time vidimo da je ovu klasu skoro savršeno klasifikovao, kao i klasu makedonski, od 101 instance, tačno je klasifikovao 101 što je savršen scenario. Od 238 instanci klase hrvatski 197 je klasifikovano tačno, a 41 netačno (7 kao srpski i 34 kao bosanski). Od 326 instanci klase srpski 298 je klasifikovano tačno, dok je 28 klasifikovano netačno (7 kao hrvatski, 19 kao bosanski i 2 kao makedonski). Od 347 instanci klase bosanski 273 je klasifikovano tačno, dok je 73 klasifikovano netačno (42 kao hrvatski i 31 kao srpski).

4. Zaključak

Na osnovu dobijenih rezultata može se zaključiti da sva četiri korišćena tokenizatora daju slične rezultate, a AlphabeticTokenizer se pokazao kao najbolji izbor za ovaj skup podataka.

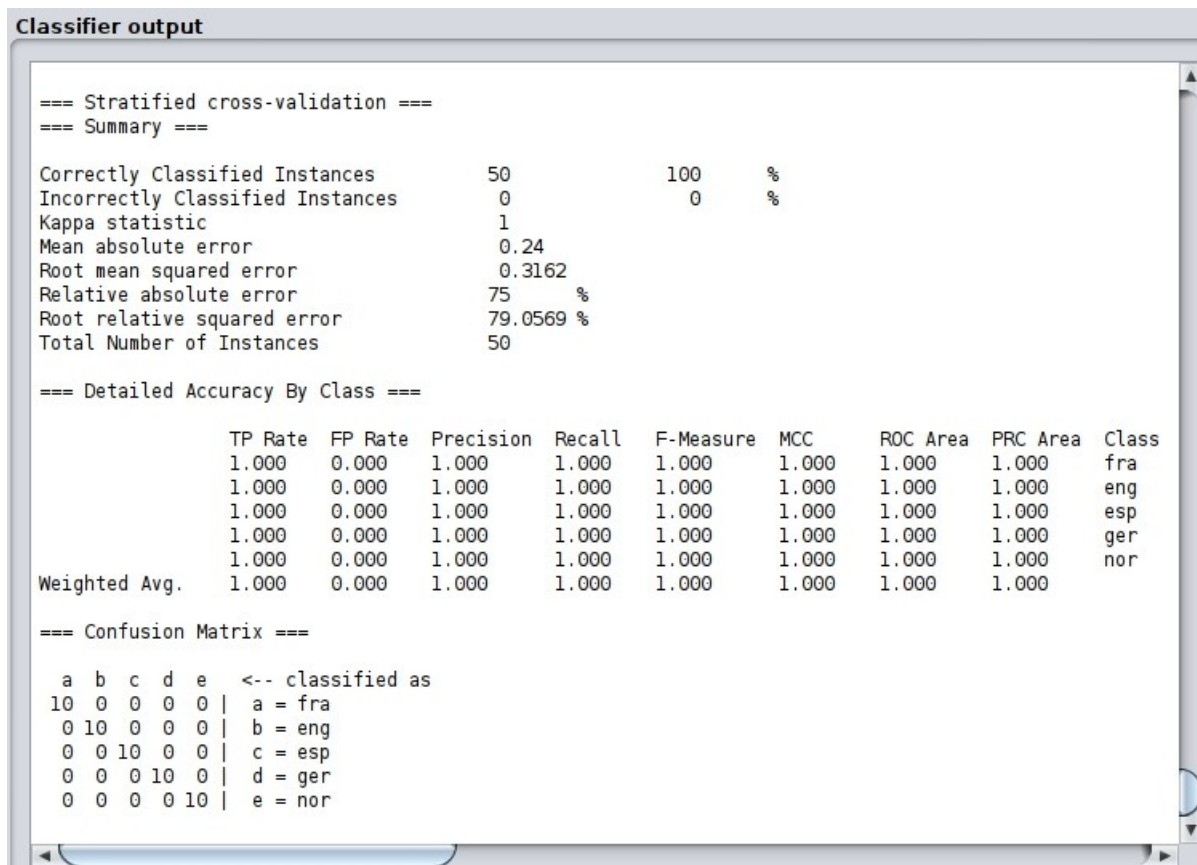
Tokenizator	Tačno klasifikovane instance (%)
CharacterNGramTokenizer	87,042%
NGramTokenizer	87,042%
WordTokenizer	86,8633%
AlphabeticTokenizer	87,2207%

U praksi se uglavnom CharacterNGramTokenizer pokazao kao najbolji kada se radi sa tekstualnim podacima. Iz tog razloga je u ovom projektu najveća pažnja poklonjena radu sa n-gram znacima.

Pored toga što se klasifikacija obavljala nad podacima koji predstavljaju srodne jezike, bolju klasifikaciju je otežalo i to sa kakvim podacima se radi. Naime, radi se o titlovima koji često nisu dobro formatirani. U smislu da su neki karakteri poput č,ć,š,ž predstavljeni pogrešnim karakterima poput æ ili œ što je čest slučaj kada se radi o titlovima. Ovo je doprinelo drugačijoj tokenizaciji u odnosu na onu koja bi se dobila da se radi o savršenim podacima i verovatno bi klasifikator dao bolje rezultate. Ovo nije imalo uticaj kod korišćenja AlphabeticTokenizer-a jer on radi samo sa karakterima engleskog alfabeta, dok je kod svih ostalih tokenizatora uticalo na rezultat. To je jedan od razloga zašto se AlphabeticTokenizer pokazao kao najbolji za rešavanje ovog problema. Pretpostavka je da bi se za veći skup podataka dobili bolji rezultati.

U ovom projektu fokus je bio na sagledavanju realne situacije, stoga nisu korišćene dodatne metode filtriranja atributa poput selekcije atributa koja se često koristi kada se radi o podacima tekstualnog sadržaja.

Takođe bi rezultati bili mnogo bolji, da su se koristili jezici koji nisu srodni. To je reprezentovano primerom na slici Slika 4.1. Korišćeni su podaci takođe na 5 različitih jezika (francuski, nemački, engleski, španski i norveški) sa po 10 instanci za svaku klasu. Iako se radi o veoma malom skupu podataka, biće 100% tačno klasifikovanih instanci uz korišćenje CharacterNGramTokenizer-a za default vrednosti NGramMinSize=1 i NGramMaxSize=3. Takođe, u ovom slučaju su podaci dosta čistiji i balansirani.



Slika 4.1