

Verifikacija govornika

Andrijana Brkić, Elio Bartoš

Sažetak—U ovom radu bavimo se problemom verifikacije govornika. Taj problem gledamo u kontekstu primjene za otključavanje uređaja, pa nam je stoga bitno da provalnik/varalica nikada ne uspije otključati uređaj, ali isto tako, radi kvalitete korištenja, želimo da korisnik ne mora jako puno puta izgovarati riječ kako bi otključao vlastiti uređaj. Pri tome se koncentriramo na tekstualno ovisno prepoznavanje, tj. za otključavanje je potrebno reći točno određenu riječ (lozinku).

Za dobivanje atributa iz zvučnih zapisa koristimo MFCC koeficijente, a za klasifikaciju stroj s potpornim vektorima (SVM) i neuronske mreže (NN). Algoritme testiramo na skupu podataka koje smo sami prikupili, a sastoji se od ukupno 136 zvučnih zapisa govora 30 različitih osoba. Za dvije osobe, jednu mušku i jednu žensku, pretpostavili smo da su korisnici i pokušali naći optimalne parametre za njihovo identificiranje.

Oba algoritma (SVM i NN) pokazuju se iznimno uspješnima pri rješavanju zadanog problema, pri tome treba uzeti u obzir da bi na većem skupu podataka greška ipak trebala biti veća, jer će algoritmi raditi gotovo isključivo na osobama koje nisu u skupu za učenje. Također, na našem skupu podataka teško je reći koji je od algoritama bolji. Ipak, obzirom da su oba dala prilično dobre rezultate, malu prednost dajemo SVM-u zbog manjeg vremena izvršavanja.

I. UVOD

Bavimo se problemom identifikacije osobe prema karakteristikama njenog glasa iz zvučnog zapisa. Glavni cilj ovog problema je otkriti je li osoba zaista ona kojom se predstavlja. Korisna primjena ovog problema bila bi otključavanje raznih uređaja poput vrata, sefova, automobila ili mobitela, računalni log-in ili korištenje bankarskih usluga.

Sličan problem je i prepoznavanje govornika gdje iz danog skupa osoba i zvučnog zapisa treba odrediti je li osoba sa zapisa u tom skupu i ako je, koja je to osoba. Identifikacija (prepoznavanje) govornika mogu biti ovisni ili neovisni o tekstu. Mi smo se u radu bazirali na tekstualno ovisno prepoznavanje, gdje svaka osoba govori istu riječ (lozinku), te smo bili motivirani problemom otključavanja nekog uređaja.

Problem ovakvog tipa sastoji se od dva dijela. Prvo trebamo iz zvučnog zapisa izvući attribute (*feature*) koji će sadržavati korisne informacije o zvučnom zapisu i predstavljati karakteristike glasa osobe sa zapisa. Drugi problem je problem klasifikacije na temelju izabranih atributa. Za biranje atributa tradicionalno se koriste MFCC (*Mel-frequency cepstral coefficients*) [1], ali i druge transformacije poput valić transformacije [2] i slično. Za problem klasifikacije najčešće se koriste neuronske mreže [2], a prije njihovog razvoja i neki vjerojatnosni modeli poput GMM-a [3].

U nastavku rada prvo ćemo opisati kako smo prikupili podatke, zatim objašnjavamo kako smo dobili attribute iz zvučnih zapisa, te na kraju objašnjavamo kako smo koristili SVM i neuronske mreže za klasifikaciju i prikazujemo dobivene rezultate.

Cjelokupni kod napisan je u Matlabu te su korištene gotove funkcije za MFCC koeficijente, SVM klasifikator te neuronske mreže.

II. PRIKUPLJANJE PODATAKA

Bazu podataka smo prikupili sami. Uz pomoć mikrofona na dva mobitela snimali smo osobe kako govore riječ "otključaj" tri puta. Osobe su snimane u raznim zatvorenim prostorijama, pri čemu smo se trudili da u pozadini ne bude velike buke. Na taj način snimili smo 28 osoba, te prikupili 84 zapisa koji će nam predstavljati provalnike (osobe koje se lažno predstavljaju, kojima ne smije biti dopušteno korištenje uređaja). Kako smo motivirani problemom otključavanja nekog uređaja u primjenama će se dogoditi da se često čuje glas osobe koja je vlasnik uređaja i otključava ga, a ponekad će i provalnici pokušati otključati. Zato smo još snimili vlastite glasove (za osobu A i osobu B) 26 puta. Osoba A i osoba B će nam biti vlasnici uređaja, tj. oni kojima algoritam mora dopustiti korištenje uređaja. Dakle, ukupno imamo 30 osoba i 136 zapisa. Osoba A je žensko, a osoba B je muško. Među 28 provalnika 12 osoba je žensko, a 16 osoba muško. Kako bismo otežali prepoznavanje i dobili realističnije uvjete, zapise osobe A i osobe B snimali smo u različitim uvjetima kao što se može vidjeti u tablici I.

Tablica I
OKOLINA UZ SNIMANJE ZAPISA KOD OSOBE A I OSOBE B

Uz tišinu	13
Uz pozadinsku priču	4
Uz hodanje	4
Uz otvoren prozor	5
Ukupno	26

Nakon što smo snimili sve osobe, pomoću programa Audacity odrezali smo zapise tako da svaki zapis počne točno kada osoba počne govoriti riječ "otključaj" i završava odmah nakon.

Skup podataka podijelili smo na skup za treniranje, skup za validaciju, te skup za testiranje. Skup za treniranje koristimo za učenje parametara našeg klasifikatora, a sastoji se od 19 zapisa slučajno odabranih 19 osoba (osim one koju verificiramo), te 3 snimke osobe A (ili osobe B, ovisno koju verificiramo). Namjerno nismo sve osobe stavili u skup za treniranje, jer nije realno da ćemo imati sve osobe za koje algoritam treba raditi u skupu za treniranje. Skup za validaciju služi nam za izbor najboljih hiperparametara modela. U njega smo stavili 19 zapisa istih 19 osoba, po 1 snimku novih 3 osoba, te 7 snimki osobe A (ili osobe B). Potrebno nam je više stvarno pozitivnih primjera kako bismo dobili bolje procjene mjera kao što su preciznost (*precision*) i osjetljivost (*recall*), te iz njih izvedenu F_α mjeru, no o tome ćemo detaljnije kasnije. Konačno u test skup smo stavili po 1 zapis svih 29 osoba

(dakle 6 potpuno novih osoba), te sve preostale zapise osobe A (ili osobe B). Konačan broj snimki u svakom skupu može se vidjeti u tablici II. Napomenimo da su slučajne osobe različite za osobu A i osobu B što rezultira time da su i skupovi za treniranje, validaciju i testiranje različiti za osobu A i B, te da smo iskoristili svih 136 snimki.

Tablica II
BROJ ZAPISA PO SKUPOVIMA

Skup	Broj zapisa
Train	22
Validacijski	29
Test	45

III. IZVLAČENJE ATRIBUTA - MFCC

Jedan zvučni zapis sastoji se od vektora koji zapisuje pritisak zraka kroz vrijeme i broja f (*frequency sampled*) koji govori koliko smo točaka u jednoj sekundi zapisali u vektor. Naš cilj je iz tog vektora dobiti značajke koje će nam pomoći klasificirati govornika. Značajke koje smo odabrali su MFCC koeficijenti (*Mel-frequency cepstral coefficients*). Kako bismo ih dobili prvo moramo primijeniti filter koji može zanemariti nagle promjene signala (*pre-emphasizing filter*). Nakon toga zvučni zapis dijelimo na male dijelove (*frames*) duljine 25 ms sa pomakom 10 ms. To znači da je prvi frame 0.000 : 0.025 s, drugi 0.010 : 0.035 s, itd. Na kraju ćemo za svaki frame dobiti MFCC koeficijente tj. značajke. Kako bismo dobili značajke na svaki frame primijenjujemo Fourierovu transformaciju, zatim računamo mel-frequency spektar, te primijenimo logaritmiranje i diskretnu kosinus transformaciju. Detaljnije o tome kako dobiti MFCC koeficijente može se pročitati u [1]. Prilikom izvlačenja MFCC koeficijenata možemo birati broj koeficijenata koje želimo koristiti. To je jedan od hiperparametara našeg modela. Isprobavali smo broj MFCC koeficijenata u rasponu od 10 do 30 koeficijenata. U konačnici, za jedan zvučni zapis dobivamo puno podataka (onoliko koliko imamo frameova), te prilikom klasifikacije pokušavamo klasificirati svaki frame, a onda na temelju toga odlučujemo se za klasifikaciju govornika. Postupak se može vidjeti na slici (1).

IV. KLASIFIKATORI

Nakon što smo snimke posložili u skupove za treniranje, validaciju i testiranje, te svaki zapis podijelili u frameove i dobili MFCC koeficijente, imamo podatke u matricnom obliku prikladnom za primjenu algoritama strojnog učenja. Svaki takav algoritam odnosno klasifikator pokušava za jedan frame odrediti je li pozitivan (1) ili negativan (0). Nakon što klasificiramo svaki frame, gledamo postotak frameova koji su bili pozitivni (za svaki zapis). Uvodimo hiperparametar rez (*cut*) koji određuje koliki postotak frameova u jednom zapisu mora biti pozitivan da bismo ga klasificirali kao pozitivan. Općenito u ovom radu hiperparametrima zovemo one parametre koje ćemo određivati uz pomoć mjera izmjerenih na validacijskom skupu. Najbolji cut određujemo pomoću F_α mjere koja je definirana sa:

$$F_\alpha = (1 + \alpha^2) \frac{\text{preciznost} \cdot \text{osjetljivost}}{(\alpha^2 \cdot \text{preciznost}) + \text{osjetljivost}}.$$

Za $\alpha = 1$ jednaka težina je dana i na preciznost i na osjetljivost. Za $\alpha < 1$ mjera preferira veću preciznost, a za $\alpha > 1$ mjera preferira veću osjetljivost. Primjetimo da je u našem problemu otključavanja uređaja bitnije da osoba koja je provalnik ne uspije otključati uređaj, nego da mi baš iz prve otključamo uređaj. Naravno loše je i ukoliko moramo puno puta reći lozinku kako bismo otključali uređaj, no smisleno je u ovom problemu dati veću težinu na preciznost (koja odgovara tome da provalnik rjeđe otključa mobitel) u odnosu na osjetljivost (koja odgovara tome da nam treba više pokušaja za otključati mobitel). Zato smo koristili $\alpha = 1$. Na validacijskom skupu za svaki postotak izračunamo F mjeru, te za svaki cut koji maksimizira F mjeru, uzimamo njihov prosjek kao konačni cut. Na primjer ako cut-ovi 0.25, 0.26, 0.27, 0.28, 0.29 svi daju $F_\alpha = 1$ (što odgovara tome da je u negativnim zapisima postotak pozitivnih frameova ispod 0.25, a u pozitivnima iznad 0.29) naš završni cut će biti njihov prosjek, tj. broj 0.27.

Još jedan hiperparametar je broj MFCC koeficijenata koje ćemo izvući za svaki frame iz zvučnog zapisa. Taj broj je zapravo broj značajki koje će koristiti naši klasifikatori. Od njega i ostalih specifičnih hiperparametara modela napravili smo sve kombinacije, te pomoću mjera na validacijskom skupu odredili najbolje. Detaljnije opis koju mjeru smo koristili slijedi u sljedećem podpoglavlju. Isprobavali smo broj koeficijenata 10, 13, 15, 20, 25, 30.

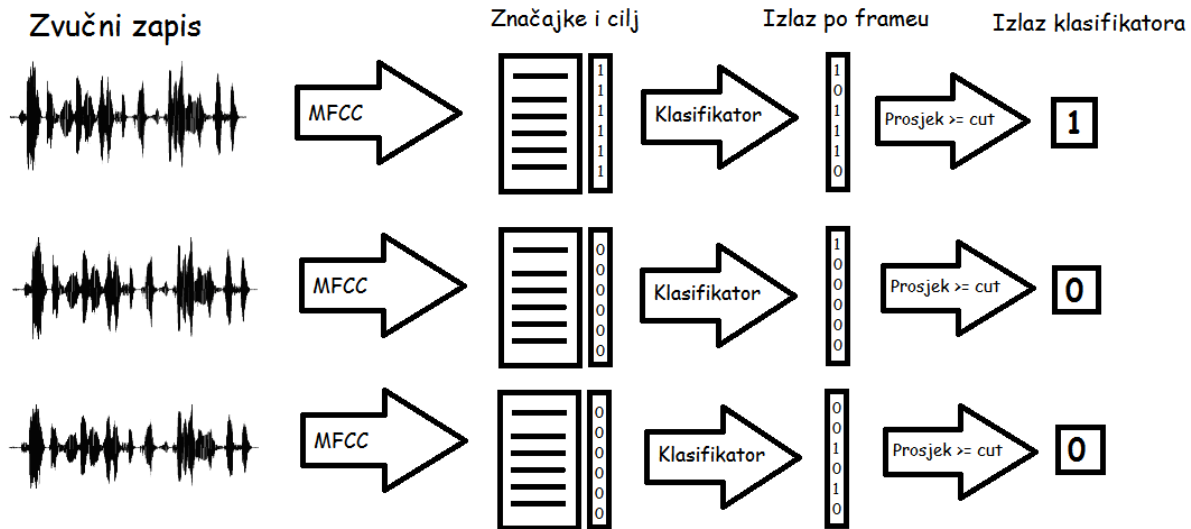
A. Stroj s potpornim vektorima (SVM)

Prvi klasifikator koji smo isprobali je stroj s potpornim vektorima ili skraćeno SVM. Koristili smo gaussovsku jezgru (*Gaussian kernel*, *radial basis kernel*) koja računa sličnost podataka x i y prema

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right).$$

Ovdje nam se javlja hiperparametar σ . Ako je sigma jako velik, sličnosti između podataka će biti glade (ne jako izražene, ne naglo promjenjive) što rezultira većom pristranošću modela i manjom varijansom. Ako je σ jako malen sličnosti su izražene što rezultira modelom sa manjom pristranošću, ali većom varijansom. Pošto je broj značajki (broj MFCC koeficijenata) između 10 i 30, a broj frameova u skupu za treniranje, validacijskom skupu i skupu za testiranje između 2000 i 5000, ima smisla koristiti gaussovsku jezgru. Broj značajki je premali da bi se sa linearnim SVM-om (bez jezgre) dobro razdvojili podaci, a opet podataka nema tako jako puno da bi gaussovska jezgra sporo radila. Također, kod problema verifikacije govornika često su se koristili GMM (*Gaussian mixture models*) i davali dobre rezultate pa smatramo da vektori značajki za svaki frame za jedan zapis slijede višedimenzionalnu normalnu razdiobu, pa ima smisla koristiti baš gaussovsu jezgru.

Kod SVM-a javlja se i hiperparametar C koji je obrnut od onoga što inače smatramo regularizacijskim faktorom. Ako je C jako malen to znači da veliku težinu ima regularizacijski faktor u funkciji gubika što rezultira modelom sa većom pristranošću, a manjom varijansom. Ako je C jako velik, mala težina ide na regularizaciju što rezultira modelom sa manjom pristranošću te većom varijansom. Kako bismo



Slika 1. Shematski prikaz postupka izvlačenja značajki i klasifikacije

izabrali najbolje hiperparametre σ i C automatizirali smo isprobavanje raznih kombinacija ovih parametara uz broj MFCC koeficijentata. Za svaku kombinaciju naučimo na skupu za treniranje parametre SVM-a, a zatim želimo pomoću neke mjere na validacijskom skupu vidjeti koja kombinacija ovih hiperparametara je najbolja. Prva ideja bila je opet gledati mjeru F_α , no kako nemamo puno stvarno pozitivnih snimki u validacijskom skupu (samo 7) često bi više kombinacija parametara dalo istu optimalnu vrijednost za F , pa ne bismo na jedinstven način mogli odrediti najbolje parametre. Trebali smo neku mjeru koja nije toliko diskretna, koja ne ovisi samo o brojevima iz matrice konfuzije. Kako za svaki zapis imamo postotak frameova koji su pozitivni odlučili smo se gledati mjeru entropije definiranu sa

$$E(y_s, y_p) = \sum_i y_s(i) \log(1 - y_p(i)) + (1 - y_s(i)) \log(1 - y_p(i))$$

pri čemu je $y_s(i) \in \{0, 1\}$ stvarna klasa zapisa, a $y_p(i)$ postotak pozitivnih frameova za taj zapis. Ako je $y_s(i) = 0$ ova mjeru kažnjava kada je $y_p(i)$ daleko od 0 i obrnuto. Kako su $y_p(i)$ postotci, pri čemu za jedan zvučni zapis ima od 50 – 100 frameova, tu ćemo dobiti različite brojeve za različite kombinacije hiperparametara, pa ovim postupkom očekujemo dobiti jedinstveno najbolje hiperparametre. Nakon određivanja najboljih C i σ postupkom kao što smo već objasnili tražimo najbolji cut. Na kraju treniramo ovako dobiven model na skupu za testiranje. Za moguće σ i C koristili smo: 1, 3, 10, 30, 100. Detaljnije objašnjenje SVM-a sa sintaksom kao i u našem radu može se naći na [4].

B. Neuronske mreže

Idući klasifikacijski algoritam koji smo koristili su neuronske mreže. Kod njih kao hiperparametar shvaćamo broj skrivenih slojeva i broj čvorova u skrivenim slojevima (arhitekturu mreže). Broj ulaznih čvorova nam je jednak broju MFCC koeficijentata, a broj izlaznih čvorova je jedan. Kao aktivacijsku

funkciju koristimo sigmoidu. Za svaki frame dobivamo broj $y \in [0, 1]$. Dakle, za razliku od SVM-a koji za svaki frame glasa 0 ili 1, neuronska mreža nam daje vjerojatnost da je frame pozitivan ili negativan. Zato kao konačan postotak pozitivnih uzimamo prosjek tih vjerojatnosti. Zatim postupamo na isti način kao u SVM-u, ukoliko je taj postotak veći od cut-a, klasificiramo zapis kao pozitivan. Da bismo odredili najbolje hiperparametre, za razne njihove kombinacije naučili smo mrežu na skupu za treniranje. Zatim smo izračunali mjeru entropije na validacijskom skupu. Uzeli smo onu kombinaciju hiperparametara koja minimizira entropiju. Zatim smo izračunali cut pomoću F mjere kao što je već spomenuto. Kao moguće arhitekture isprobali smo:

- jedan skriveni sloj: 3, 6, 10, 15, 20 čvorova.
- dva skrivena sloja: (10, 10), (7, 7), (7, 3), (10, 3), (10, 7).
- tri skrivena sloja: (7, 7, 4), (8, 4, 2), (10, 5, 3), (8, 5, 5), (5, 5, 5), (7, 5, 3).

Vrlo važno je naglasiti da je cilj bio da je izbor hiperparametara potpuno automatiziran. Općenito, ne možemo očekivati da će jedna kombinacija hiperparametara biti dobra za verifikaciju svake osobe. Cilj nam je bio napraviti program koji će automatski od skupa podataka za treniranje i validaciju naći najbolji SVM model i najbolju neuronsku mrežu za verifikaciju baš te određene osobe. Kod smo implementirali u Matlab-u, a algoritme strojnog učenja smo učili pomoću Matlab-ovih ugrađenih funkcija. U sljedećem odlomku donosimo dobivene rezultate.

V. REZULTATI

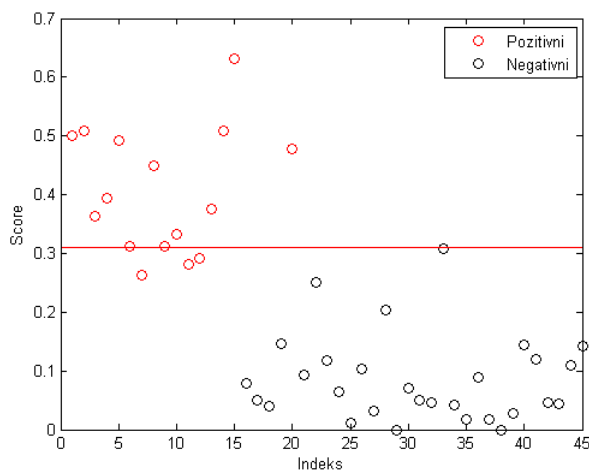
A. SVM - osoba A

Konačne parametre SVM modela za osobu A možemo vidjeti u tablici III.

Na slici ispod možemo vidjeti predikcije optimalnog modela na skupu za testiranje.

Tablica III
OPTIMALNI PARAMETRI

Sigma	3
C	1
mfcc	30
cut	0.31



Crvenom bojom označeni su stvarno pozitivni podaci (TP), a crnom stvarno negativni (TN). Također, horizontalna crvena linija predstavlja cut iz optimalnog modela. Sa slike je vidljivo da smo pri biranju cut-a preferirali veliku preciznost u odnosu na osjetljivost. Naime, vidimo da bi spuštanjem cut-a do najnižeg crvenog podatka dobili osjetljivost 1, ali bi također jedan stvarno negativan podatak klasificirali kao točan, čime bi pala preciznost. Naravno, treba uzeti u obzir da je cut biran na validacijskom setu, a ne na testnom kojeg vidimo ovdje.

Matricu konfuzije možemo vidjeti u tablici IV, a osnovne statistike u tablici V.

Tablica IV
MATRICA KONFUZIJE

	Stvarno	
	1	0
	1	0
Predviđeno	1	13
	0	3
		29

Tablica V
STATISTIKE

Preciznost	1
Osjetljivost	0.8125
Točnost	0.9333
F	0.9559

Preciznost je 1 što znači da niti jedna osoba nije uspjela provaliti, a osjetljivost od oko 80% znači da u 5 pokušaja korisnik uspije otključati uređaj 4 puta.

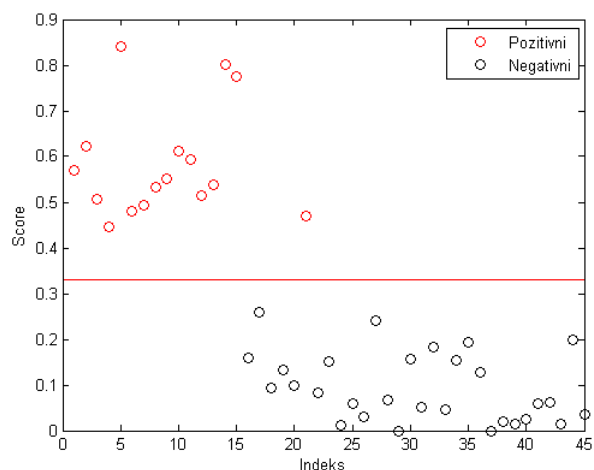
B. SVM - osoba B

Konačne parametre SVM modela za osobu B možemo vidjeti u tablici VI.

Tablica VI
OPTIMALNI PARAMETRI

Sigma	3
C	10
mfcc	13
cut	0.33

Na slici ispod možemo vidjeti predikcije optimalnog modela na skupu za testiranje.



Kao i prije, crvenom bojom označeni su stvarno pozitivni podaci (TP), a crnom stvarno negativni (TN). Također, horizontalna crvena linija predstavlja cut iz optimalnog modela.

Matricu konfuzije možemo vidjeti u tablici VII, a osnovne statistike u tablici VIII.

Sa slike i iz statistika vidimo da cut savršeno razdjeljuje podatke. Iako to možda djeluje preoptimistično, primjetimo da to ne znači da naš klasifikator savršeno klasificira sve podatke. Naime, on klasificira svaki frame zasebno te u ponekim griješi, ali u prosjeku klasificira točno. Zato je moguće na malom skupu ljudi dobiti savršenu klasifikaciju.

Tablica VII
MATRICA KONFUZIJE

	Stvarno	
	1	0
	1	0
Predviđeno	1	16
	0	0
		29

Tablica VIII
STATISTIKE

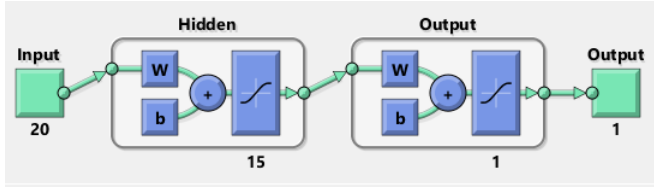
Preciznost	1
Osjetljivost	1
Točnost	1
F	1

Primjetimo kako je klasifikacija za osobu B puno efikasnija od one za osobu A iako ukupno ima više snimljenih muških nego ženskih osoba, što nam govori da kvaliteta algoritma subjektivno ovisi i o glasu. Možemo također vidjeti da je broj

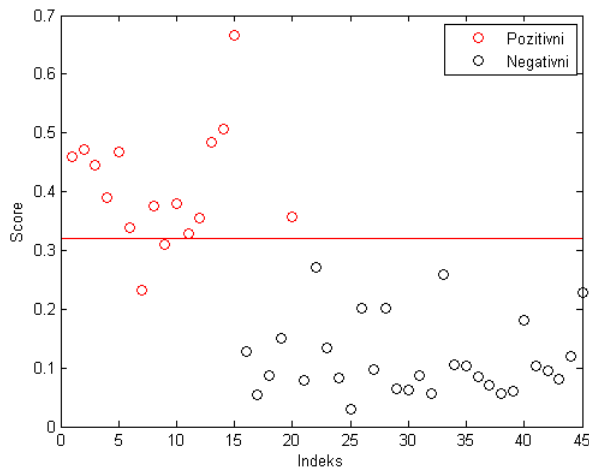
potrebnih ulaznih parametara za osobu A veći nego za osobu B, što pokazuje da je ta klasifikacija teža.

C. Neuronske mreže - osoba A

Na slici ispod možemo vidjeti shematski prikaz dobivene optimalne neuronske mreže.



Slika prikazuje neuronsku mrežu sa 20 ulaznih (MFCC) parametara te jednim skrivenim slojem sa 15 čvorova. Cut i klasifikaciju test seta možemo vidjeti na sljedećoj slici.



Konfuzijska matrice te statistike su u tablicama IX i X.

Tablica IX
MATRICA KONFUZIJE

	Stvarno		
		1	0
	Predviđeno	1	0
		1	14
		0	2
			29

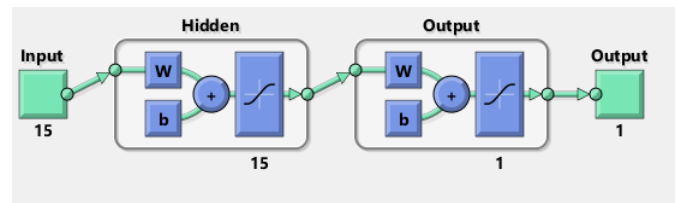
Tablica X
STATISTIKE

Preciznost	1
Osjetljivost	0.8750
Točnost	0.9556
F	0.9722

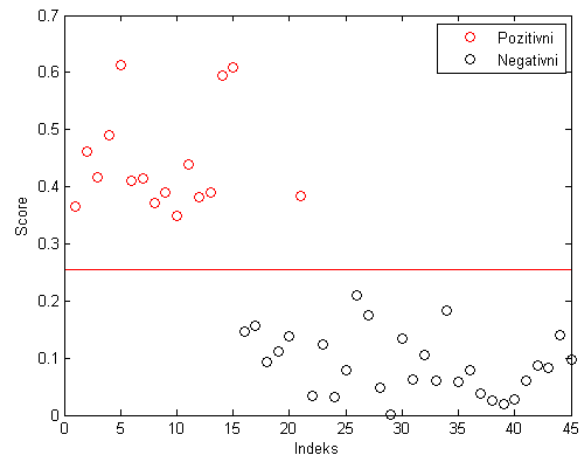
Primjetimo da su rezultati neznatno bolji od onih koje smo dobili sa SVM-om.

D. Neuronske mreže - osoba B

Na slici ispod možemo vidjeti shematski prikaz dobivene optimalne neuronske mreže.



Slika prikazuje neuronsku mrežu sa 15 ulaznih (MFCC) parametara te jednim skrivenim slojem sa 15 čvorova. Cut i klasifikaciju test seta možemo vidjeti na sljedećoj slici.



Vidimo sa slike te iz tablica XI i XII da i neuronska mreža, kao i SVM, u ovom slučaju postiže maksimalan učinak u klasifikaciji.

Tablica XI
MATRICA KONFUZIJE

	Stvarno		
		1	0
	Predviđeno	1	0
		1	16
		0	0
			29

Tablica XII
STATISTIKE

Preciznost	1
Osjetljivost	1
Točnost	1
F	1

VI. ZAKLJUČAK

U ovom radu testirali smo SVM algoritam i neuronske mreže za problem verifikacije govornika koristeći MFCC koeficijente kao značajke. Motivacija nam je bila otključavanje uređaja. Uz ručno skupljene podatke snimane u raznim, ne idealnim uvjetima, ovi algoritmi su dali poprilično dobre rezultate. Također, zbog toga što smo u testni skup dali i zapise osoba koje nismo stavili niti u skup za treniranje, niti u skup za validaciju smatramo da dobiveni rezultati imaju određenu težinu. Naime, dodavanje do tada nepoznatih ljudi u test set je nužno za dobru procjenu greške, obzirom da će u stvarnosti upravo to biti najčešći slučaj pokušaja provaljivanja.

Verificiranje osobe B je uspjelo savršeno, dok smo kod osobe A imali preciznost 100%, a osjetljivost veću od 80% što ugrubo znači da niti jedan provalnik nije uspio otključati uređaj osobe A, a osoba A je od 5 pokušaja otključavanja uspjela otključati mobitel 4 puta. Naravno, mana naših rezultata je veličina skupa podataka. Kada bismo imali puno više osoba i zapisa mogli bismo u testni skup staviti sve sasvim nove ljude koji se ne pojavljuju niti u skupu za treniranje niti u skupu za validaciju, te dobiti realističnije procjene greške.

U radu smo uočili i da verifikacija ovisi o osobi koju želimo verificirati. Za osobu B algoritmi su savršeno radili, dok su za osobu A malo lošije radili. To nas vodi na zaključak da postoje osobe sa glasom koji je "prepoznatljiviji" pa će klasifikatori imati lakši posao. Ipak klasifikatori su solidno radili za obje osobe što nas navodi da bi mogli solidno raditi za većinu osoba. Za provjeru te tvrdnje i detaljniju analizu potrebno bi nam bilo još više zapisa istih osoba kako bi one mogle biti osoba koju verificiramo.

Dobra stvar je što nismo koristili puno pozitivnih zapisa za treniranje (samo tri) što osoba može jako brzo snimiti. Naravno treba snimiti i par zapisa za validacijski skup (mi smo koristili 7 zapisa), no već to je dovoljno za uspješan rad algoritma, a pri tome korisnik ne gubi puno vremena. Pozitivna stvar je i automatizirano traženje najboljeg modela, što je nužno kako bi se ovakav model uopće mogao koristiti u vidu neke aplikacije.

Ovim radom smo potvrdili i opravdanost korištenja MFCC koeficijenata kao atributa u problemu verifikacije govornika.

Na temelju malog uzorka kojeg smo imali u skupu za testiranje ne možemo tvrditi da je neki od algoritama, SVM ili neuronske mreže, bolji od drugog. Oba algoritma su dali vrlo dobre rezultate i smatramo da se podjednako dobro mogu koristiti u svrhu verificiranja govornika. Malu prednost bismo ipak dali SVM-u zbog bržeg učenja (manjeg vremena izvršavanja) u odnosu na neuronske mreže.

LITERATURA

- [1] Lindsalwa Muda, Mumtaj Begam, and I Elamvazuthi. Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques. *arXiv preprint arXiv:1003.4083*, 2010.
- [2] V Srinivas, T Madhu, et al. Neural network based classification for speaker identification. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 7(1):109–120, 2014.
- [3] Douglas A Reynolds. Automatic speaker recognition using gaussian mixture speaker models. In *The Lincoln Laboratory Journal*. Citeseer, 1995.
- [4] Andrew Ng. Coursera, Stanford: Machine Learning, <https://www.coursera.org/learn/machine-learning/home/welcome>.