

# Primena mašinskog učenja za prepoznavanje šablona pokreta oka

Stefan Ruvčeski

Fakultet tehničkih nauka, Novi Sad, Srbija

stefanruvceski@uns.ac.rs

**Apstrakt** — U ovom radu rešavan je problem prepoznavanja šablona pokreta oka u obrazovne svrhe, tačnije konzumiranja elektronskih testova. Za praćenje pokreta oka korišćen je GazePoint uređaj. Prikupljeni sirovi podaci u vidu X i Y koordinata su prevedeni u podatke sa više semantike, distancu pokreta oka od jedne do druge tačke i ugao koji prave dve distance u obliku prave. Dobijene sekvence su zatim mapirane na sekvence regija od interesa. Nakon završenog procesiranja podataka, izvršena je klasterizacija radi identifikovanja šablona direkcionihi pokreta oka, tačnije tok pokreta oka. Navedene funkcionalnosti su objedinjene u softversku biblioteku koja je finalni proizvod rada.

**Ključne reči** – mašinsko učenje, makro sekvenca, mikro sekvenca, klasterizacija, sakadna distanca, sakadni ugao, praćenje pokreta oka

## I. UVOD

Analiza pokreta oka se koristi da bi se shvatilo kako korisnici procesiraju informacije vizuelnog tipa. Utvrđeno je da korelacija između pokreta oka i procesiranja vizuelnih informacija postoji [1]. Iz toga proizilazi potreba da se pronađe semantička veza i unapredi konzumacija takvih informacija. Doprinos računarstva u pomenutoj oblasti je najviše iskorišćen u marketinške svrhe. Takođe se primenjuje u Interakcija čovek-računar (engl. Human-computer interaction, HCI) istraživanju.

Rad se bavi manje istraženim domenom primene pomenutih tehnika. Primena računarstva u edukativne svrhe bazirana je pretežno na materijale za predavanje u obliku prezentacija. Radovi na datu temu identifikuju i objašnjavaju kako studenti konzumiraju sadržaj, kako bi prezentacija trebala da bude koncipirana da zadrži studentovu pažnju što duže i kako je prilagoditi što većem spektru individua. Primena nad elektronskim testovima je manje istražen domen. Prednosti takvih testova nad standardim pristupom su prvenstveno lakša i jeftinija distribucija i prikupljanje [2]. Pored navedenog, ekologija je još jedan benefit ovog pristupa. Rezultat je isti – broj bodova. Standardno, u klasičnim i

elektronskim testovima konačan rezultat evidentira osvojeni broj bodova. Pokret oka je jedan od parametara kojeg je jednostavnije evidentirati na elektronskom testu nego na papirnom zbog položaja glave. Doprinos praćenja je velik, pored rezultata osvojenih bodova moguće je dobiti mnogo više informacija o studentovom procesiranju pitanja i odgovora. Može se utvrditi koliko je gledao u tačan odgovor a koliko u ostale, na koji način je konzumirao scenu, šta ga je najviše privlačilo i najbitnije za rad, koje su mu direkcije pokreta oka bile najizraženije.

Glavni cilj rada je identifikacija šablona pokreta oka radi lakšeg prepoznavanja grupa studenata prema direkciji pokreta oka tokom rešavanja elektronskog testa.

Rad je zasnovan na prikupljanju podataka pokreta oka tokom odgovaranja na dvadeset pitanja iz opšte kulture. Motivacija rada proizilazi iz potrebe prilagođenja vizuelne organizacije elektronskih testova onima koji ih polažu. Da bi se takav vid primenio u obrazovne svrhe i samo polaganje potrebno je prikupiti podatke metodom praćenja pokreta oka, analizirati dovoljan broj učenika i zatim ih grupisati u određene klase. Benefiti ovog pristupa su moguća personalizacija testova svakom tipu korisnika nakon prepoznavanja kojoj klasi isti pripada.

Finalni proizvod rada je softverska biblioteka implementirana u Python programskom jeziku. Biblioteka sadrži procesiranje podataka, klasterizaciju i vizuelizaciju dobijenih klastera. Namena biblioteke je da bude pomoć pri sastavljanju daljih testova na osnovu dobijenih rezultata prethodnih testova.

## II. PREGLED RELEVANTNE LITERATURE

Rešavani problem rada sastoji se iz prikupljanja podataka pokreta oka, prepoznavanja šablona pokreta oka i grupisanja istih. Rad [1] koji se bavi sličnim problemom, većinski je usmeren ka veb platformama, analiziranjem korisničkih sekvenci pokreta oka i izmenama grafičkog prikaza, tako da zadovolji što širi spektar istih. Istraživanja u edukativnom domenu su prikazana u radu [3] i tiču se predavanja, tačnije materijala koji se, na sličan način kao i veb platforme, adaptira nakon analize prikupljenih podataka.

Tema rada [4] je karakterizacija rezultata pokreta oka zasnovana na analizi direkcionih pokreta oka pri gledanju veb sadržaja. Sistematski je prikazana sumarijacija direkcionih šablona pokreta oka metodama na kojima se ovaj rad zasniva.

Rad teži ka tome da istraži deo domena koji nije izučen u potpunosti i oslanja se na rad [4] koji je baziran na sumarijaciji sekvenci po načinu konzumiranja veb sadržaja.

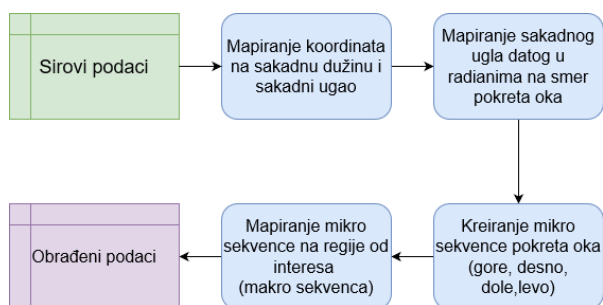
### III. SOFTVERSKA BIBILOTEKA ZA IDENTIFIKACIJU ŠABLONA DIREKCIONIH POKRETA OKA

#### A. Klijentska aplikacija

Postupak procesiranja podataka ne bi bio moguć bez prethodnog prikupljanja istih. Za potrebe prikupljanja podataka korišćena je aplikacija razvijena u Visual Studio okruženju i programskom jeziku C#, kao i WPF-u (*Windows Presentation Foundation*) za vizuelni prikaz samih pitanja na koje student odgovara. Aplikacija je istovremeno omogućila interakciju sa korisnikom, putem prikaza pitanja i mogućnosti odgovora na ista, i komunikaciju sa GazePoint uređajem putem XML komandi koje razmenjuje sa serverom pomenutog uređaja koji izlaže iste za prikupljanje sirovih podataka, u obliku X i Y koordinata pokreta oka, sa uređaja. Nakon završetka studentovog testiranja podaci dobijeni od servera su sačuvani u csv formatu spremni za dalje procesiranje.

#### B. Procesiranje podataka

Izlazni podaci klijentske aplikacije su takođe i već pomenuti sirovi podaci, naredna faza je procesiranje istih. Ulaz u pomenutu fazu su X i Y koordinate što nije dovoljno semantički bogato za klasterizaciju i identifikaciju šablona. Slika 1. prikazuje pomenuti tok procesiranja podataka.



Slika 1. Dijagram toka procesiranja

**Definicija 1 (sakada):** Obzirom na dati set tačaka pogleda oka  $P = \{p_1, p_2, \dots, p_n\}$ , sakada između bilo koja dva uzastopna pogleda  $p_p$  i  $p_q$  definisana je linijom koja povezuje prostorne koordinate  $(x_p, y_p)$  i  $(x_q, y_q)$  za svaku tačku gledanja.

**Definicija 2 (sakadna dužina):** Obzirom na sakadu koja se sastoji od linije koja povezuje prostorne koordinate  $(x_p, y_p)$  i  $(x_q, y_q)$  dve sekvencijalne tačke pokreta oka  $p_p$  i  $p_q$ , sakadna dužina  $sd$  je euklidska udaljenost između  $p_p$  i  $p_q$  definisana sledećom jednačinom:

$$sd = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2}$$

Jednačina 1. Sakadna distanca

**Definicija 3 (sakadni ugao):** Obzirom na sakadu koja se sastoji od linije koja povezuje prostorne koordinate  $(x_p, y_p)$  i  $(x_q, y_q)$  dve sekvencijalne tačke pokreta oka  $p_p$  i  $p_q$ , sakadni ugao  $sa$  je ugao ravne linije između  $p_p$  i  $p_q$ . Ugao se izračunava kao tangenta luka između razlike prostornih koordinata  $(x_p, y_p)$  i  $(x_q, y_q)$  i definisana je sledećom jednačinom:

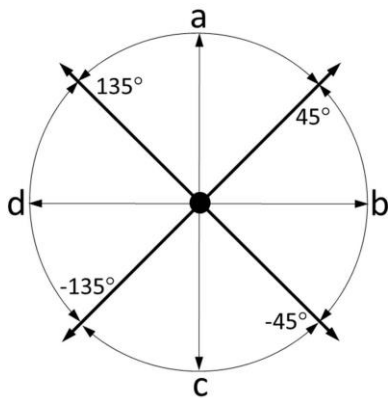
$$sa = \tan^{-1}(x_p - x_q, y_p - y_q)$$

Jednačina 2. Sakadni ugao

Generalni pristup rada je mapiranje sirovih podataka na sakadnu dužinu i sakadni ugao. Zatim se podaci sakadne dužine dobijene u radianima pretvaraju u stepene. Nakon toga se formira mikro sekvenca direkcionim kodiranjem sekvence. Poslednji korak procesiranja podataka je pretvaranje mikro sekvence na makro sekvencu mapiranjem koordinata na regije od interesa. Rad obrađuje promene toka pogleda oka. Šablon predstavlja najučestalije promene toka pogleda kod studenata tokom odgovaranja na pitanja.

Prvi korak procesiranja je računanje sakadne distance i sakadnog ugla sukcesivnih X i Y koordinata.

Direkciono kodiranje sekvence, tačnije drugi korak u procesiranju podataka je mapiranje sakadnog ugla u smer pokreta oka (gore – a, desno – b, dole – c, levo – d). Pošto je sakadna dužina izražena u radianima, prvo se mora pretvoriti u stepene zatim mapirati na karaktere prikazane na slici 2. Mapiranje uglova na karaktere smera se izvršava po sledećem šablonu, smer na gore zahvata ugao između  $45^\circ$  i  $135^\circ$ , smer na desno zahvata ugao između  $45^\circ$  i  $-45^\circ$ , smer na dole zahvata ugao između  $-45^\circ$  i  $-135^\circ$  i smer na levo zahvata ugao između  $135^\circ$  i  $-135^\circ$ . Nakon drugog koraka dobijena je mikro sekvenca karaktera koji predstavljaju smer pokreta oka studenta za jedno pitanje.

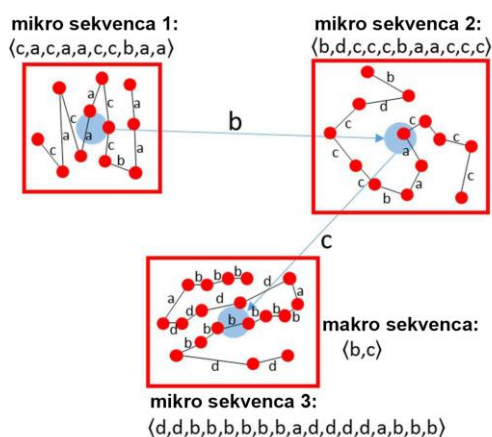


**Slika 2.** Mapiranje direkcije sekvence na karaktere

**Definicija 4 (mikro sekvenca):** Obzirom na skup sakada  $S = \{s_1, s_2, \dots, s_n\}$  gde je svaka sakada  $s_j$  predstavljena sakadnim uglom, sakadnom distancom i karakterom smjera, mikro sekvenca je podskup sakada za jedno pitanje.

**Definicija 5 (makro sekvenca):** Obzirom na skup sakada  $S = \{s_1, s_2, \dots, s_n\}$  gde je svaka sakada  $s_j$  predstavljena sakadnim uglom, sakadnom distancom i karakterom smjera, makro sekvenca je podskup sakada za jedno pitanje gde svaka sakada predstavlja pomeraj toka pogleda oka od jedne do druge regije od interesa.

Treći i poslednji korak podrazumeva pretvaranje mikro sekvence u makro sekvencu uz pomoć regija od interesa koje predstavljamo preko X i Y koordinata u dvodimenzionom prostoru jednog pitanja. Regije od interesa su uvedene da bismo šablon toka pogleda oka posmatrali na makro nivou, tačnije nivou regija od interesa. Ukoliko se pogled oka nalazi u jednoj regiji od interesa, karakteri mikro sekvence se zanemaruju, uzima se prvi karakter nakon kog je pogled oka prešao sa jedne regije na sledeću. Tako nastaje makro sekvenca direkcije pokreta oka, koja je agregacija podataka mikro sekvence na nivou regije od interesa a ne X i Y koordinata. Primer makro i mikro sekvence je dat na slici 3.



**Slika 3.** Primer mikro i makro sekvence

### C. Pronalaženje najučestalijih podsekvenci

Izlazni podaci iz obrade su makro sekvence direkcionih karaktera opisanih gore. Treće faza rada je izvlačenje najčešćih podsekvenci iz makro sekvence implementacijom top\_k algoritma [2] prikazanog na slici 4. Izvlačenje najčešćih podsekvenci se vrši radi klasterizacije studenata na osnovu najučestalijih promena toka pogleda, makro sekvenca je velika, zbog toga je potrebno dobiti k podsekvenci manje dimenzionalnosti.

```

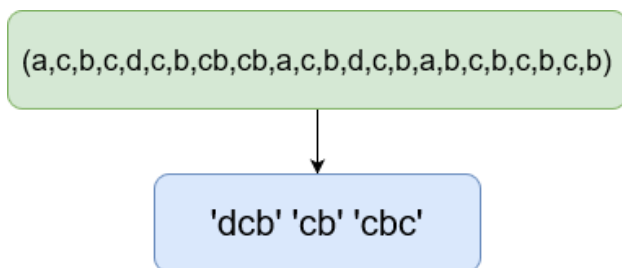
1: Input: A macrosequence  $\mathcal{MA}$ ,  $k$  value, maximum candidate subsequence length  $\alpha$ 
2: Output: Top- $k$  frequent macrosequences  $\mathcal{FMA}$ 
3:  $\mathcal{MA}_{sub} \leftarrow \emptyset$ ;
4:  $sequencelist \leftarrow \emptyset$ ;
5:  $\mathcal{FMA} \leftarrow \emptyset$ ;
6:  $cnt \leftarrow 1$ ;
7: //Sliding window to calculate frequency for each  $\mathcal{MA}_{sub} \in \mathcal{MA}$ ; loop until  $\alpha$ 
   threshold is met
8: for ( $i$  from 1 to  $\alpha$ ) do
9:   for ( $j$  from 1 to  $|\mathcal{MA}| - \alpha$ ) do
10:     $\mathcal{MA}_{sub} \leftarrow \mathcal{MA}(j : j + i)$ 
11:    if ( $\mathcal{MA}_{sub} \notin sequencelist$ ) then
12:       $sequencelist(cnt) \leftarrow \mathcal{MA}_{sub}$ 
13:       $\mathcal{MA}_{sub}^{freq} \leftarrow |\mathcal{MA}_{sub} \in \mathcal{MA}|$ 
14:       $cnt \leftarrow cnt + 1$ ;
15:    end if
16:   end for
17: end for
18: Top- $k$   $\mathcal{FMA} \in \mathcal{MA} \leftarrow \text{SORT}(\mathcal{MA}_{sub})(1 : k)$ 
19: Return Top- $k$  frequent  $\mathcal{FMA} \in \mathcal{MA}$ 

```

**Slika 4.** Top k algoritam

Ulazni parametri algoritma su makro sekvenca, maksimalna željena dužina podsekvenci i broj najčešćih sekvenci. Algoritam sadrži dve petlje, spoljašnja iterira kroz sve veličine podsekvenci od 1 do  $\alpha$ . Unutrašnja petlja iterira kroz makro sekvencu i traži sve podsekvence i broj njihovog ponavljanja. Nakon završetka spoljašnje petlje u nizu  $\mathcal{masub}$  se nalaze sve podsekvence makro sekvence a  $\mathcal{mafreqsub}$  niz sadrži broj ponavljanja svake podsekvence. Na kraju algoritma niz se sortira opadajuće, povratna vrednost algoritma je prvih  $k$  podsekvenci iz niza  $\mathcal{masub}$ .

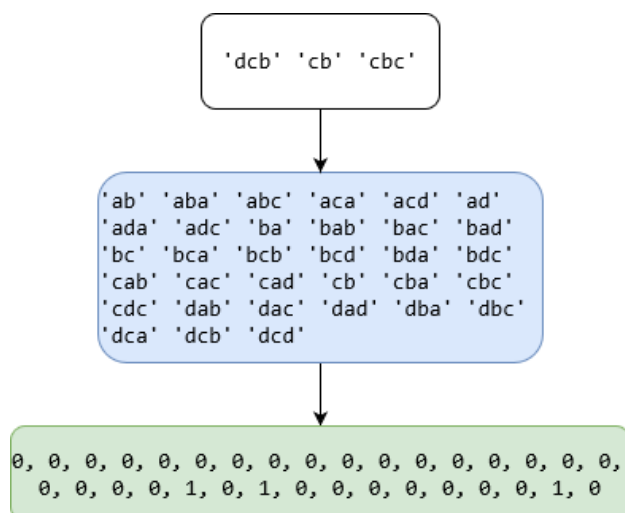
Rezultat gore pomenutog algoritma je niz od  $k$  najučestalijih podsekvenci dužine od jedan do  $\alpha$ , gde je algoritam modifikovan tako da daje na značaju dužim podsekvencama. Najučestalija podsekvenca predstavlja sekvencu unutar makro sekvence dužine do  $\alpha$  karaktera koja se najviše puta pojavila u makro sekvenci. Duže podsekvence dobijaju prednost tako što se broj ponavljanja množi sa dužinom podsekvence. U radu parametri su  $k=3$ ,  $\alpha=4$ . Biblioteka koja je proizvod rada ostavlja na korisniku da sam definiše pomenute parametre. Na slici 5 je prikazan primer makro sekvence i njene tri najučestalije podsekvence.



Slika 5. Primer makro sekvence i podsekvenci

#### D. Mapiranje sekvence na jedinstvenu sekvencu

Ulaz u četvrtu fazu rada jesu najučestalije podsekvence svih studenata za jedno pitanje. Problem koji rešava trenutna faza je nekonzistentnost dužine samih podsekvenci, koji je nastao iz razloga što *top k* algoritam pronalazi podsekvence dužine od 1 do  $\alpha$ . Takođe mora da reši i problem zanemarivanja redosleda samih sekvenci iz razloga što nije bitno na kom mestu u vektoru se data podsekvencu nalazi. Rešenjem ovog problema se omogućava pronalaženje sličnih šablona iako im redosled podsekvenci nije identičan jer to nije relevantno za dobijene podsekvence. Rešenje je pronalazak sekvence koja sadrži sve sekvence koje su dobijene od bilo kog studenta, takvu sekvencu nazivamo jedinstvena sekvenca. Nakon pronalaska sekvence kreira se vektor čiji je broj elemenata jednak svim mogućim sekvencama. Nakon toga jedinica se postavlja na mesto gde trenutni korisnik poseduje datu sekvencu, ostatak vektora je popunjen nulama. Primer trenutne faze se može videti na slici 6. Prvi pravougaonik predstavlja najučestalije podsekvence, drugi jedinstvenu sekvencu. Poslednji pravougaonik je rezultat celokupne faze, namapirane podsekvence prvog pravougaonika na podsekvence drugog u vidu jedinica krajnje sekvence.



Slika 6. Tok kreiranja jedinstvene sekvence

#### E. Redukcija dimenzionalnosti

Redukovanje dimenzionalnosti je potrebno zbog visoke dimenzionalnosti dobijenih jedinstvenih sekvenci. U radu je primenjen PCA (Principal Component Analysis). PCA je statistički postupak koji koristi ortogonalnu transformaciju da pretvori skup vrednosti eventualno korelisanih promenljivih u skup vrednosti linearno nekorelisanih promenljivih [5]. U radu dimenzionalnost je redukovana na tri dimenzije da bi se rezultat mogao vizuelno prikazati.

Vrednost varijanse varira između 0.5 i 0.65. Skup podataka dobijen ovom metodom je moguće vizuelizovati u trodimenzionalnom prostoru. Pretvaranjem visokodimenzionog problema u trodimenzioni, gubi se određena količina semantike koju su originalni podaci sadržali.

#### F. Klasterizacija

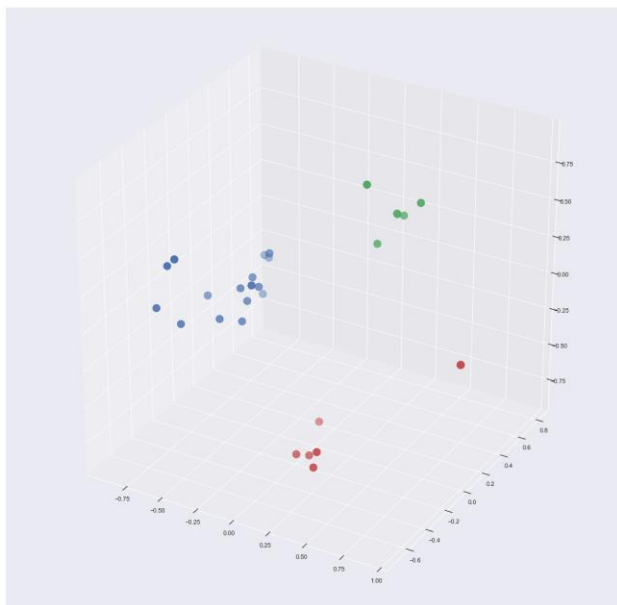
Klasterizacija je vid nenadgledanog mašinskog učenja, koji ima za cilj pronalaženje homogenih grupa podataka u datom skupu podataka. Svaka od ovih grupa naziva se klaster i može se definisati kao regija u kojoj je gustina objekata lokalno veća nego u ostalim regionima [6]. Popularni metod klasterizacije koji minimizuje grešku klasterovanja jeste k-means algoritam. U radu je predložen pomenuti algoritam koji predstavlja deterministički algoritam klasterizacije koji koristi k-means algoritam kao lokalni postupak pretraživanja.

Top k algoritmu potreban je unapred predefinisani broj klastera. Biblioteka koju rad izlaže korisniku iste omogućava izbor broja klastera kao parametar funkcije, dok podrazumevana vrednost posle analize dobijenih rezultata je tri ili četiri, u zavisnosti od tipa pitanja, pošto je ustanovljeno da je u većini slučajeva dati broj najoptimalniji za klasterizaciju dobijenih podataka. Rezultat funkcije je niz labela koje predstavljaju dobijene klasterne.

#### G. Vizuelizacija

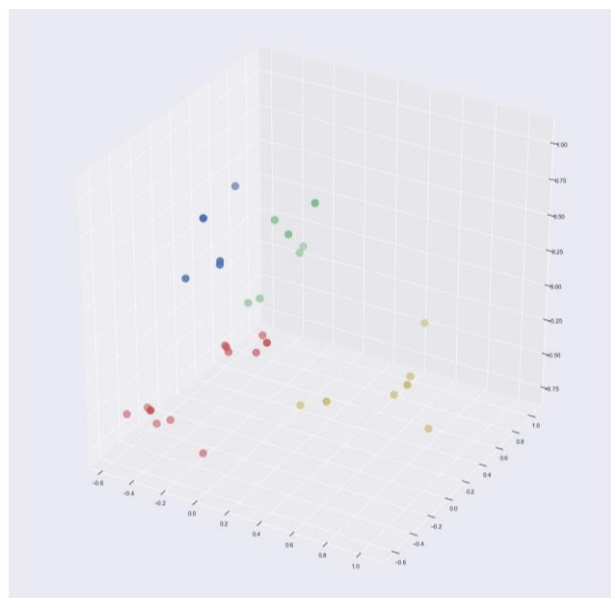
Nakon klasterizacije biblioteka izlaže funkcionalnost vizuelne reprezentacije dobijenih klastera u vidu trodimenzionalnog *scatter plot*-a. Primer rezultata vizuelizacije koji je primenjen nad pitanjem četiri, koje sadrži četiri slike i četiri ponuđena odgovora, za sve studente se može videti na slici 7.





**Slika 7.** Grupe studenata za prvi tip pitanja

Druga vrsta pitanja uzetih u obzir u radu je pitanje sa jednom slikom i četiri ponuđena odgovora. U ovom slučaju biblioteka za podrazumevanu vrednost broja klastera uzima četiri jer je ustanovljeno da je to najčešći broj klastera. Vizuelni prikaz pomenutog tipa pitanja se može videti na slici 8.



**Slika 8.** Grupe studenata za drugi tip pitanja

Pored osnovne vizuelizacije grupa studenata nakon klasterizacije, biblioteka izlaže i funkcionalnost vizuelizacije direkcije pokreta oka tokom konzumacije jednog pitanja u vidu dvodimenzionalnog plot-a toka studentovog pokreta oka.

#### IV. STUDIJA SLUČAJA

Da bi postojali sirovi podaci koji se kasnije procesiraju potrebno je prikupiti podatke. Za evaluaciju biblioteke izvršen je eksperiment u kojem je učestvovao pedeset i jedan student. Svaki student je odgovorio na 20 istih pitanja, time je dobijeno 1020 sekvenci sirovih podataka u obliku X i Y koordinata. Ispitanici su u velikom broju studenti Fakulteta tehničkih nauka i velik broj se bavi računarstvom što nije uticalo na podatke jer su pitanja opšte kulture, nevezana za pomenutu oblast. Test iz opšte kulture je isključivo pravljen za potrebe rada.

Svaki ispitanik je imao identične uslove, u prostoriji pored opreme koja je bila potrebna se nalazio jedan izvršitelj eksperimenta kako bi objasnio i nadzirao validnost eksperimenta. Pomenuta potrebna oprema uključuje računar na kom student rešava test, uređaj za praćenje pokreta oka i periferije računara za interakciju sa datim testom.

Sam test je sačinjen od dva tipa pitanja, prvi tip podrazumeva pitanje, četiri ponuđena odgovora i četiri slike koje prate četiri ponuđena odgovora. Time zbir regija od interesa je devet. Primer pitanja prvog tipa se može videti na slici 9.



**Slika 9.** Primer slike prvog tipa pitanja

Drugi tip pitanja podrazumeva pitanje, sliku koja dopunjuje postavljeno pitanje i četiri ponuđena odgovora. Time zbir regija od interesa je šest. Primer pitanja drugog tipa se može videti na slici 10.



**Slika 10.** Primer slike drugog tipa pitanja

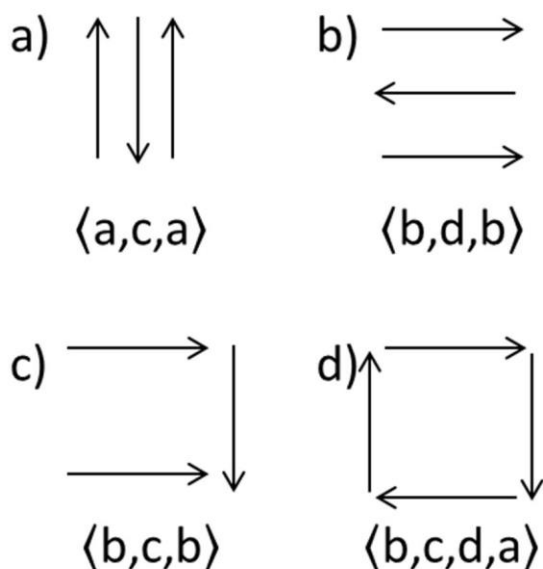
Nakon završetka eksperimenta, praćenje pokreta oka svih ispitanika rezultovalo je sa 51 CSV (*Comma Separated Values*) datotekom. Sirovi podaci dobijeni sa *GazePoint* uređaja su ulaz za narednu fazu, procesiranje podataka.

Procesiranje podataka korišćenjem sirovih podataka računa semantički značajnije podatke u vidu sakadne distance (*sd*), sakadnog ugla (*sa*). Poslednji podatak koji će se na dalje koristiti je direkcija pogleda oka (*dir*). Kolona *dir* predstavlja ulaznu sekvencu nazvanu mikro sekvenca od koje se dobija makro sekvenca mapirana na regije od interesa. Deo izlaznih podataka iz faze procesiranja se može videti u tabeli 1.

Question	X	Y	sd	sa	dir
1	869.54	191.21	19.86	179.72	d
1	759.11	189.45	0.45	116.97	a
1	566.43	19.78	6.56	-40.94	b
1	759.51	189.24	0.41	118.05	a
1	759.87	189.05	0.37	168.18	d

**Tabela 1.** Deo podataka nakon procesiranja

Naredna faza podrazumeva pronalazak najučestalijih podsekvenci iz makro sekvence *top k* algoritmom. Primeri najučestalijih podsekvenci se mogu videti na slici 11.



**Slika 11.** Primeri najučestalijih podsekvenci

Dobijene podsekvence su zatim transformisane u jedinstvenu sekvencu popisanu nulama i jedinicama. Postupak podrazumeva pronalazak jedinstvene sekvence iz svih dobijenih podsekvenci za određeno pitanje, zatim za svakog ispitanika kreira zasebna sekvenca jedinstvenih vrednosti sa jedinicama na pozicijama gde se podsekvenca poklapa sa njegovom. Ostatak sekvence je popisan nulama.

Studija slučaja se završava finalnim proizvodom biblioteke, prikazane u radu, u vidu vizuelizacije dobijenih klastera studenata u zavisnosti od njihovog toka pomeraja oka tokom rešavanja elektronskog testa.

## V. ZAKLJUČAK

U radu je predstavljena primena mašinskog učenja za prepoznavanje šablona direkcionog pokreta oka. Ispitanici su rešavali test opšte kulture dok je uređaj pratio pokrete oka. Eksperiment je sproveden nad više od 50 ispitanika, većinom studenata.

Finalni proizvod rada jeste biblioteka implementirana u programskom jeziku Python. Funkcionalnosti biblioteke su procesiranje, klasterizacija i vizuelizacija podataka.

U toku implementacije biblioteke, analizirana je razlika rezultata kada se obrađuje mikro i makro sekvenca. Nakon dobijenih podsekvenci ustanovljeno je da su razlike u podsekvencama minimalne. Zaključak je da nad testnim podacima postoje minimalne razlike prilikom rada sa mikro ili makro sekvencama.

Rezultate treba uzeti sa rezervom zbog sledećih faktora: Prvi faktor koji se uzima u obzir je posmatranje samo toka promena smera pogleda umesto celokupnog sadržaja. Dobijeni rezultati pokazuju da i pored pomenutog ograničenja postoje šabloni toka pogleda oka nezavisno od sadržaja već samo smera gledanja. Drugi faktor je sam uređaj koji nije u potpunosti precizan, sklon je greškama koje utiču na podatke. Student je u obavezi da sedi mirno, ne pomera vrat i glavu, time utiče na validnost dobijenih podataka.

Klasterizacija koja je implementirana u krajnjoj verziji biblioteke je pokazala da postoje tri do četiri grupe osoba kroz sva pitanja, time možemo pretpostaviti da postoji neki definisani šabloni direkcije pokreta oka za pitanja kreirana za ovaj eksperiment.

Bitno je napomenuti da rad nije dao semantičko značenje klasterima koji su proizašli kao rezultat. Za takav postupak su potrebni domenski stručnjaci koji bi utvrdili semantiku iza svake grupe studenata. Biblioteka je namenjena predavačima da služi samo kao alat pri sastavljanju testova, za analizu direkcionih šablona i većeg broja informacija o studentima osim broja bodova.

Rad je osnova za dalja istraživanja u domenu unapređenja digitalnih testova, pronalazaka šablona direkcionih pokreta oka u raznim domenima i sličnih biblioteka za klasterizaciju pogleda. Takođe unapređenje rada bi moglo izvršiti uz domenske stručnjake koji bi dali semantičko značenje klasterima i time kompletirali istraživački rad.

# REFERENCE

- [1] M. Schiessl, S. Duda, A. Tholke i R. Fischer, Eye tracking and its application in usability and media research, Berlin, 2003.
- [2] Samson Ejim, An Overview of Computer Based Test.
- [3] S. Parikh i H. Kalva, Eye Gaze Feature Classification for Predicting Levels of Learning, 2018.
- [4] 2017M. P. McGuire i J. Chakraborty, Directional Scat Path Characterization of Eye Tracking Sequences: A Multi-Scale Approach, San Francisco, 2016.
- [5] H. Abdi and L. J. Williams, Principal Component Analysis, 2010.
- [6] Aristidis Likas,Nikos Vlassis i Jakob J. Verbeek, The global k-means clustering algorithm, Greece, Netherlands,2002.