Pretraživanje, usporedba i klasifikacija

Kokor, Ana

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Ana Kokor

PRETRAŽIVANJE, USPOREDBA I KLASIFIKACIJA

Diplomski rad

Voditelj rada:
doc.dr.sc. Pavle Goldstein

Zagreb, rujan, 2019.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Veliko hvala A1 Hrvatska d.o.o., a posebno kolegici Andrei Pirši, na ustupljenim podacima i pruženoj potpori pri izradi rada.

Sadržaj

Sadržaj **iv**

Uvod **1**

1 Matematički pojmovi **2**
1.1 Linearna algebra 2
1.2 Statistika 6
1.3 Optimizacija 8

2 Opis problema **11**
2.1 Priprema teksta 11
2.2 Reprezentacija teksta 13
2.3 Klasifikacija 15

3 Detekcija fraza i ključnih riječi **24**
3.1 Faktor lokalnih izuzetaka 24
3.2 Pristupi računanju težina pridruženih pojmovima u dokumentu 25

4 Analiza rezultata **27**
4.1 Usporedba predstavljenih modela 27
4.2 Zaključak 30
4.3 Korišteni paketi 32

Bibliografija **33**

Uvod

U današnje vrijeme komunikacija sve više prelazi na digitalne kanale. Tako su i telekomunikacijske kompanije prepoznale koristi pružanja korisničke podrške putem digitalnih (chat) razgovora. U ovom ćemo se radu baviti analizom takvih razgovora između korisnika i agenata službe za korisnike A1 telekom operatera. A1 nudi brojne usluge iz područja telekomunikacija kao i razna druga digitalna rješenja. Podrška za veliki broj korisnika takvih usluga zahtijeva neprestani rad stručnjaka iz raznih područja. Želimo naći algoritam koji će biti sposoban prepoznati temu razgovora, što brže i točnije moguće. Tako ćemo moći pratiti razdiobu tema (koje ćemo još zvati klasa ili kategorija) razgovora i ovisno o učestalosti pojavljivanja svake teme odrediti broj potrebnih stručnjaka po područjima rada. Osim toga, nagle promjene u spomenutoj razdiobi će u pravo vrijeme ukazati na potencijalne probleme pa ćemo i tako moći prilagoditi podršku korisnicima.

Cilj ovog rada je ispitati i usporediti kako detekcija fraza i ključnih riječi te korištenje različitih mjera koje daju težinu pojmovima u tekstu utječu na rezultate klasifikacije - podjele razgovora po temama.

U ovom radu su sadržana 4 poglavlja. U prvom poglavlju ćemo navesti definicije i teoreme iz linearne algebre, statistike i optimizacije koje će biti potrebne za razumijevanje ostatka rada. U drugom poglavlju ćemo malo bolje opisati problem i objasniti kako tekst prilagoditi algoritmima klasifikacije i detekcije fraza i ključnih riječi. Predstavit ćemo 3 klasifikacijska algoritma i objasniti matematičku pozadinu iza njih. U trećem poglavlju govorimo o algoritmu kojim pronalazimo fraze i ključne riječi u cijeloj kolekciji dokumenata te predstavljamo nekoliko različitih pristupa računanju težina pridruženih pojmovima u dokumentu. Konačno, u četvrtom poglavlju analiziramo rezultate i biramo najbolji model.

Poglavlje 1

Matematički pojmovi

U ovom poglavlju dajemo neke definicije i tvrdnje iz linearne algebre, statistike i optimizacije. Pojmovi iz linearne algebre su velikim dijelom preuzeti iz [1]. Statistika je sažeta iz bilješki s kolegija Statistika, Matematička statistika i [8]. Optimizacija je preuzeta iz [5].

1.1 Linearna algebra

Definicija 1.1.1. *Neka je $\mathbb{F}$ neki skup na kojem su zadane binarne operacije zbrajanja*

$$+ : \mathbb{F} \times \mathbb{F} \to \mathbb{F}$$

i množenja

$$\cdot : \mathbb{F} \times \mathbb{F} \to \mathbb{F}$$

koje imaju sljedeća svojstva:

1. $\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma, \ \forall \alpha, \beta, \gamma \in \mathbb{F}$;
2. $\exists\, 0 \in \mathbb{F}$ *sa svojstvom* $\alpha + 0 = 0 + \alpha = \alpha, \ \forall \alpha \in \mathbb{F}$;
3. $\forall \alpha \in \mathbb{F}, \exists - \alpha \in \mathbb{F}$ *tako da je* $\alpha + (-\alpha) = (-\alpha) + \alpha = 0$;
4. $\alpha + \beta = \beta + \alpha, \ \forall \alpha, \beta \in \mathbb{F}$;
5. $\alpha(\beta\gamma) = (\alpha\beta)\gamma, \ \forall \alpha, \beta, \gamma \in \mathbb{F}$;
6. $\exists\, 1 \in \mathbb{F} \setminus \{0\}$ *sa svojstvom* $1 \cdot \alpha = \alpha \cdot 1 = \alpha, \ \forall \alpha \in \mathbb{F}$;
7. $\forall \alpha \in \mathbb{F}, \alpha \neq 0, \exists\, \alpha^{-1} \in \mathbb{F}$ *tako da je* $\alpha\alpha^{-1} = \alpha^{-1}\alpha = 1$;
8. $\alpha\beta = \beta\alpha, \ \forall \alpha, \beta \in \mathbb{F}$;

9. $\alpha(\beta+\gamma)=\alpha\beta+\alpha\gamma,\ \forall\alpha,\beta,\gamma\in\mathbb{F}$.

Tada kažemo da je $\mathbb{F}$ polje.

Definicija 1.1.2. *Neka je V neprazan skup na kojem su zadane binarne operacije zbrajanja $+ : V\times V\to V$ i operacija množenja skalarima iz polja $\mathbb{F}$, $\cdot : \mathbb{F}\times V\to V$. Kažemo da je uređena trojka $(V,+,\cdot)$ vektorski prostor nad poljem $\mathbb{F}$ ako vrijedi:*

1. $a+(b+c)=(a+b)+c,\ \forall a,b,c\in V$;
2. $\exists\, 0\in V$ *sa svojstvom* $a+0=0+a=a,\ \forall a\in V$;
3. $\forall a\in V,\exists -a\in V$ *tako da je* $a+(-a)=(-a)+a=0$;
4. $a+b=b+a,\ \forall a,b\in V$;
5. $\alpha(\beta a)=(\alpha\beta)a,\forall\alpha,\beta\in\mathbb{F},\ \forall a\in V$;
6. $(\alpha+\beta)a=\alpha a+\beta a,\forall\alpha,\beta\in\mathbb{F},\ \forall a\in V$;
7. $\alpha(a+b)=\alpha a+\alpha b,\ \forall\alpha\in\mathbb{F},\forall a,b\in V$;
8. $1\cdot a=a\cdot 1,\ \forall a\in V$.

Vektori su elementi vektorskog prostora.

Definicija 1.1.3. *Neka je V vektorski prostor nad $\mathbb{F}$. Izraz oblika*

$$\alpha_1a_1+\alpha_2a_2+...+\alpha_ka_k,$$

pri čemu je $a_1,a_2,...,a_k\in V$, $\alpha_1,\alpha_2,...,\alpha_k\in\mathbb{F}$ i $k\in\mathbb{N}$, naziva se linearna kombinacija vektora $a_1,a_2,...,a_k$ s koeficijentima $\alpha_1,\alpha_2,...,\alpha_k$.

Definicija 1.1.4. *Neka je V vektorski prostor nad $\mathbb{F}$ i*

$$S=\{a_1,a_2,...,a_k\},\quad k\in\mathbb{N},$$

konačan skup vektora iz V. Kažemo da je skup S linearno nezavisan ako vrijedi

$$\alpha_1,\alpha_2,...,\alpha_k\in\mathbb{F},\quad \sum_{i=1}^{k}\alpha_ia_i=0\Rightarrow\alpha_1=\alpha_2=...=\alpha_k=0.$$

U suprotnom kažemo da je skup S linearno zavisan.

Definicija 1.1.5. *Neka je V vektorski prostor nad poljem $\mathbb{F}$ i $S \subseteq V, S \neq \emptyset$. Linearna ljuska skupa S označava se simbolom $[S]$ i definira kao*

$$[S] = \{\sum_{i=1}^{k} \alpha_i a_i : \alpha_i \in \mathbb{F}, a_i \in S, k \in \mathbb{N}\}.$$

Dodatno, definira se $[\emptyset] = \{0\}$.

Definicija 1.1.6. *Neka je V vektorski prostor i $S \subseteq V$. Kaže se da je S sustav izvodnica za V ako vrijedi $[S] = V$.*

Definicija 1.1.7. *Konačan skup $B = \{b_1, b_2, ..., b_n\}$, $n \in \mathbb{N}$, u vektorskom prostoru V se naziva baza za V ako je B linearno nezavisan sustav izvodnica za V.*

Definicija 1.1.8. *Neka je V vektorski prostor nad poljem $\mathbb{F}$. Skalarni produkt na V je preslikavanje*

$$\langle \cdot, \cdot \rangle : V \times V \to \mathbb{F}$$

koje ima sljedeća svojstva:

1. $\langle x, x \rangle \geq 0, \forall x \in V$;
2. $\langle x, x \rangle = 0, \iff x = 0$;
3. $\langle x_1 + x_2, y \rangle = \langle x_1, y \rangle + \langle x_2, y \rangle, \ \forall x_1, x_2, y \in V$;
4. $\langle \alpha x, y \rangle = \alpha \langle x, y \rangle, \ \forall \alpha \in \mathbb{F}, \forall x, y \in V$;
5. $\langle x, y \rangle = \overline{\langle y, x \rangle}, \ \forall x, y \in V$.

Definicija 1.1.9. *Vektorski prostor na kojem je definiran skalarni produkt zovemo unitarnim prostorom.*

Definicija 1.1.10. *Neka je V unitaran prostor. 2-norma na V (ili samo norma) je funkcija*

$$\|\cdot\|_2 : V \to \mathbb{R}$$

definirana s

$$\|x\|_2 = \sqrt{\langle x, x \rangle}.$$

Propozicija 1.1.11. *Norma na unitarnom prostoru V ima sljedeća svojstva:*

1. $\|x\|_2 \geq 0, \ \forall x \in V$;
2. $\|x\|_2 = 0 \iff x = 0$;

3. $\|\alpha x\|_2 = |\alpha|\|x\|_2,\ \forall \alpha \in \mathbb{F}, \forall x \in V$;

4. $\|x + y\|_2 \leq \|x\|_2 + \|y\|_2,\ \forall x, y \in V$.

Definicija 1.1.12. *Svaka funkcija $\|\cdot\| : V \to \mathbb{R}$ na vektorskom prostoru V sa svojstvima iz 1.1.11 naziva se norma. Tada $(V, \|\cdot\|)$ zovemo normirani prostor.*

Definicija 1.1.13. *Neka je V unitaran prostor. Na V definiramo preslikavanje*

$$d : V \times V \to \mathbb{R}$$

formulom

$$d(x, y) = \|x - y\|$$

i zovemo ga metrika ili udaljenost vektora x od vektora y.

Propozicija 1.1.14. *Metrika na unitranom prostoru V ima sljedeća svojstva:*

1. $d(x, y) \geq 0,\ \forall x, y \in V$;

2. $d(x, y) = 0 \iff x = y$

3. $d(x, y) = d(y, x),\ \forall x \in V$

4. $d(x, y) \leq d(x, z) + d(z, y),\ \forall x, y, z \in V$.

Definicija 1.1.15. *Neka je $X \neq \emptyset$. Svaka funkcija $d : X \times X \to \mathbb{R}$ sa svojstvima iz 1.1.14 naziva se metrika ili udaljenost. Tada (X, d) zovemo metrički prostor.*

Definicija 1.1.16. *Za prirodne brojeve m i n, preslikavanje*

$$A : \{1, 2, ..., m\} \times \{1, 2, ..., n\} \to \mathbb{F}$$

naziva se matrica tipa (m, n) s koeficijentima iz polja $\mathbb{F}$. Takve funkcije A pišemo tablično, u m redaka i n stupaca, gdje u i-tom retku i j-tom stupcu piše vrijednost $A(i, j)$, što ćemo jednostavnije označavati a_{ij}. Skup svih matrica s m redaka i n stupaca označavamo $M_{mn}(\mathbb{F})$.

Napomena 1.1.17. *Dalje ćemo u radu koristiti isključivo 2-normu i pisati $\|\cdot\|$.*

1.2 Statistika

Definicija 1.2.1. *Neka je* $(\Omega, \mathcal{F}, \mathbb{P})$ *vjerojatnosni prostor i* $A \in \mathcal{F}$ *takav da je* $\mathbb{P}(A) > 0$. *Definiramo funkciju* $\mathbb{P}_A : \mathcal{F} \to [0,1]$ *s:*

$$\mathbb{P}_A(B) = \mathbb{P}(B|A) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(A)}, \quad B \in \mathcal{F}.$$

$\mathbb{P}_A$ *je vjerojatnost na* $\mathcal{F}$ *i zovemo je uvjetna vjerojatnost uz uvjet A. Broj* $\mathbb{P}(B|A)$ *zovemo vjerojatnost od B uz uvjet da se A dogodio.*

Definicija 1.2.2. *Konačna ili prebrojiva familija* $(H_i, i = 1, 2...)$ *događaja u vjerojatnosnom prostoru* $(\Omega, \mathcal{F}, \mathbb{P})$ *jest potpun sistem događaja ako je* $H_i \neq \emptyset, \forall i$, $H_i \cap H_j = \emptyset$ *za* $i \neq j$ *i* $\bigcup_i H_i = \Omega$.

Teorem 1.2.3. *(Bayesov teorem) Neka je* $(H_i, i = 1, 2, ...)$ *potpun sistem događaja u vjerojatnosnom prostoru* $(\Omega, \mathcal{F}, \mathbb{P})$ *i* $A \in \mathcal{F}$ *takav da je* $\mathbb{P}(A) > 0$. *Tada za svako i vrijedi*

$$\mathbb{P}(H_i|A) = \frac{\mathbb{P}(H_i)\mathbb{P}(A|H_i)}{\sum_j \mathbb{P}(H_j)\mathbb{P}(A|H_j)}.$$

Neka je $(\Omega, \mathcal{F})$ proizvoljan izmjeriv prostor i neka je još dan izmjeriv prostor $(\mathbb{R}^m, \mathcal{B}(\mathbb{R}^m))$

Definicija 1.2.4. *Slučajna varijabla* $(m = 1)$ *ili slučajni vektor* $(m > 1)$ *je izmjerivo preslikavanje* $X : (\Omega, \mathcal{F}) \to (\mathbb{R}^m, \mathcal{B}(\mathbb{R}^m))$, *tj. preslikavanje za koje vrijedi*

$$X^{-1}(\langle -\infty, x]) \in \mathcal{F}, \quad \forall x \in \mathbb{R}^m.$$

Pritom je $x = (x_1, ..., x_m) \in \mathbb{R}^m, m \geq 2$ *tj.* $\langle -\infty, x] := \langle -\infty, x_1] \times ... \times \langle -\infty, x_m]$.

Neka je X slučajna varijabla na vjerojatnosnom prostoru $(\Omega, \mathcal{F}, \mathbb{P})$. Vrijedi $X = X^+ + X^-$, gdje je $X^+ = max\{X, 0\}$, $X^- = max\{-X, 0\}$.

Definicija 1.2.5. *Kažemo da slučajna varijabla X ima matematičko očekivanje ukoliko je barem jedan od integrala*

$$\mathbb{E}X^+ := \int_\Omega X^+ d\mathbb{P}, \quad \mathbb{E}X^- := \int_\Omega X^- d\mathbb{P}$$

konačan. U tom slučaju je matematičko očekivanje od X, u oznaci $\mathbb{E}X$, *jednako*

$$\mathbb{E}X := \mathbb{E}X^+ - \mathbb{E}X^- \in \overline{\mathbb{R}}.$$

Teorem 1.2.6. *Neka je* $(\Omega, \mathcal{F}, \mathbb{P})$ *vjerojatnosni prostor, X slučajna varijabla takva da* $\mathbb{E}|X| < \infty$ *te* $\mathcal{G}$ σ*-podalgebra od* $\mathcal{F}$. *Tada postoji slučajna varijabla Y koja ima sljedeća svojstva:*

1. *Y je $\mathcal{G}$-izmjeriva*
2. $\mathbb{E}|Y| < \infty$
3. $\mathbb{E}[Y\mathbb{1}_G] = \mathbb{E}[X\mathbb{1}_G], \forall G \in \mathcal{G}$

Ako je $\tilde{Y}$ neka druga slučajna varijabla sa svojstvima 1., 2. i 3., tada je $\tilde{Y}$=Y g.s.

Definicija 1.2.7. *Slučajna varijabla Y iz iskaza prethodnog teorema zove se verzija uvjetnog matematičkog očekivanja od X uz dano $\mathcal{G}$ i pišemo*

$$Y = \mathbb{E}[X|\mathcal{G}].$$

Definicija 1.2.8. *Neka je $(x_1, ..., x_n)$ opaženi uzorak za slučajnu varijablu X s gustoćom $f(x|\theta)$ gdje je $(\theta_1, ..., \theta_k) \in \Theta \subseteq \mathbb{R}^k$ nepoznati parametar. Definiramo funkciju vjerodostojnosti $L : \Theta \to \mathbb{R}$ sa*

$$L(\theta) := f(x_1|\theta) \cdots f(x_n|\theta), \quad \theta \in \Theta.$$

Vrijednost $\hat{\theta} = \hat{\theta}(x_1, ..., x_n) \in \Theta$ takva da vrijedi

$$\hat{\theta} = \max_{\theta \in \Theta} L(\theta)$$

zovemo procjena metodom maksimalne vjerodostojnosti. Statistika $\hat{\theta}(X_1, ..., X_n)$ je procjenitelj metodom maksimalne vjerodostojnosti ili kraće MLE.

Linearna regresija

Promatramo problem prilagodbe m-dimenzionalne hiperravnine točkama:

$$(x_1^{(i)}, x_2^{(i)}, ..., x_m^{(i)}, y_i), \quad i = 1, 2, ..., n$$

gdje su x_j, za $j = 1, ..., m$, nezavisne (neslučajne) varijable ili varijable poticaja dok je y zavisna slučajna varijabla (varijabla odziva).

Definicija 1.2.9. *Varijable $x = (x_1, x_2, ..., x_m)$ i y su u srednjem linearno povezane ako vrijedi:*

$$\mathbb{E}[y|x] = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + ... + \theta_m x_m.$$

Preciznije,

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + ... + \theta_m x_m + \varepsilon,$$

gdje su $\theta_0, \theta_1, \theta_2, ..., \theta_m$ parametri modela, ε je slučajna varijabla takva da je $\mathbb{E}[\varepsilon] = 0$ koju interpretiramo kao slučajnu grešku (šum).

Neka je $(x_1^{(1)}, x_2^{(1)}, ..., x_m^{(1)}, y_1), (x_1^{(2)}, x_2^{(2)}, ..., x_m^{(2)}, y_2), ..., (x_1^{(n)}, x_2^{(n)}, ..., x_m(n), y_n)$ slučajni uzorak i pretpostavljamo da su varijable $x = (x_1, x_2, ..., x_m)$ i y u srednjem linearno povezane. Problem možemo zapisati vektorski:

$$Y = X\theta + \varepsilon$$

gdje su $Y = (y_1, y_2, ..., y_n)^T$, $\varepsilon = (\varepsilon_1, \varepsilon_2, ..., \varepsilon_n)^T$ i $\theta = (\theta_0, \theta_1, \theta_2, ..., \theta_m)^T$ vektori stupci i X je matrica

$$X = (\mathbf{1}, x_1, x_2, ..., x_m) \in M_{n,m+1}(\mathbb{R})$$

kojoj su stupci: $\mathbf{1} = (1, 1, ..., 1)^T \in \mathbb{R}^n$, $x_j = (x_j^{(1)}, x_j^{(2)}, ..., x_j^{(n)})^T$, $j = 1, 2, ..., m$. Pretpostavljamo da je $n \geq m + 1$.

1.3 Optimizacija

Problem koji se sastoji od određivanja ekstrema funkcije, uz zadane uvjete, zvati ćemo problemom matematičkog programiranja. Posebno, ako je funkcija cilja linearna, govorimo o linearnom programiranju. Ako je ona kvadratna, tada govorimo o kvadratnom programiranju.

Definicija 1.3.1. *Neka je $\Omega \subseteq \mathbb{R}^n$ otvoren skup. Kažemo da funkcija $f : \Omega \to \mathbb{R}$ ima lokalni minimum u točki $P_0 \in \Omega$ ako postoji okolina $K(P_0, r) \subseteq \Omega$ takva da*

$$(\forall P \in \{K(P_0, r) \setminus P_0\}) \quad (f(P) \geq f(P_0),$$

odnosno funkcija f u $P_0 \in \Omega$ ima lokalni maksimum ako vrijedi

$$(\forall P \in \{K(P_0, r) \setminus P_0\}) \quad (f(P) \leq f(P_0).$$

Vrijednosti $f(P_0)$ zovemo minimumom, odnosno maksimumom funkcije f na skupu Ω. Ako vrijede stroge nejednakosti, radi se o strogom lokalnom minimumu, odnosno maksimumu. Ako nejednakosti vrijede za svaku točku $P \in \Omega$, tada funkcija f u točki P_0 ima globalni minimum, odnosno maksimum.

Definicija 1.3.2. *Neka je $\Omega \in \mathbb{R}^n$ otvoren skup i neka je $f : \Omega \to \mathbb{R}$ diferencijabilna funkcija. Za točku $P_0 \in \Omega$ kažemo da je stacionarna točka funkcije ako vrijedi:*

$$\partial_i f(P_0) = 0, \quad i = 1, 2, ..., n.$$

Teorem 1.3.3. *(Nužan uvjet za postojanje lokalnog ekstrema) Ako je $P_0 \in \Omega \subseteq \mathbb{R}^n$ točka lokalnog ekstrema diferencijabilne funkcije $f : \Omega \to \mathbb{R}$, onda je P_0 stacionarna točka funckije f, tj. vrijedi:*

$$\partial_i f(P_0) = 0, \quad i = 1, 2, ..., n.$$

Pretpostavimo da su zadane funkcije $f, g_i : \mathbb{R}^n \to \mathbb{R}, \quad i = 1, ..., m$, te promatrajmo sljedeći optimizacijski problem:

$$\min_{x \in \mathbb{R}^n} f(x).$$

Pri tome skup $U = \{x \in \mathbb{R}^n : g_i(x) \leq 0, \quad i = 1, ..., m\}$ zovemo dopustivo područje, a svaki $x \in U$ zovemo dopustivo rješenje. Dopustivo rješenje x^* sa svojstvom $f(x^*) \leq f(x)$ zovemo optimalno dopustivo rješenje. Gornjem problemu možemo pridružiti funkciju $L : \mathbb{R}^n \times \mathbb{R}^m_+ \to \mathbb{R}$ zadanu formulom

$$L(x, \alpha) = f(x) + \sum_{i=1}^{m} \alpha_i g_i(x).$$

Funkciju L zovemo Lagrangeova funkcija koja je pridružena problemu.

Teorem 1.3.4. *Problem*

$$\min_{x \in \mathbb{R}^n} f(x)$$
$$g_i(x) \leq 0, \quad i = 1, ..., m$$

ekvivalentan je problemu

$$\min_{x \in \mathbb{R}^n} \max_{\alpha \in \mathbb{R}^m_+} L(x, \alpha).$$

Dokaz. Označimo sa $g(x) := (g_1(x), ..., g_m(x))$. Uočimo da za fiksni $x \in \mathbb{R}^n$ vrijedi:

$$\max_{\alpha \in \mathbb{R}^m_+} L(x, \alpha) = \begin{cases} f(x) & g(x) \leq 0 \\ \infty & \text{inače.} \end{cases}$$

Naime, za $g(x) \leq 0$ maksimum funkcije L po varijabli $\alpha \geq 0$ se postiže za $\alpha = 0$. S druge strane, ako je $g_i(x) \geq 0$ za neki $i \in \{1, ..., m\}$ povećanjem vrijednosti komponenata vektora $\alpha \in \mathbb{R}^m_+$ funkciju $L(x, \alpha)$ možemo proizvoljno povećati. Minimizacijom po $x \in \mathbb{R}^n$ vidimo da se minimum od $\max L(x, \alpha)$ postiže za $g(x) \leq 0$ i da je on jednak minimumu funkcije na dopustivom skupu $U = \{x \in \mathbb{R}^n : g(x) \leq 0\}$ te su prema tome navedeni problemi usitinu ekvivalentni. □

Problem iz teorema zovemo primarni problem, a budući da je rješenje primarnog problema ujedno rješenje originalnog optimizacijskog problema, njega također zovemo primarni problem. Možemo promatrati sljedeći optimizacijski problem

$$\max_{\alpha \in \mathbb{R}^m_+} \min_{x \in \mathbb{R}^n} L(x, \alpha)$$

kojeg zovemo dualni problem. Nadalje, pretpostavimo da je zadan problem linearnog programiranja u sljedećem obliku

$$\begin{cases} f(x) = c^T x \to min_x, \\ Ax \geq b. \end{cases}$$

gdje su $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m, c \in \mathbb{R}^n$. Definiramo Lagrangeovu funkciju $L : \mathbb{R}^n \times \mathbb{R}^m_+ \to \mathbb{R}$ formulom

$$L(x, \alpha) = c^T x + \alpha^T (b - Ax).$$

Odgovarajući primarni i dualni problem tada glase

$$\min_{x \in \mathbb{R}^n} \max_{\alpha \in \mathbb{R}^m_+} L(x, \alpha).$$

$$\max_{\alpha \in \mathbb{R}^m_+} \min_{x \in \mathbb{R}^n} L(x, \alpha).$$

Poglavlje 2

Opis problema

2.1 Priprema teksta

Prije nego što krenemo analizirati tekst želimo ga dovesti do neke forme u kojoj će to biti smisleno. Primjer jednog razgovora na početku je na slici 2.1.

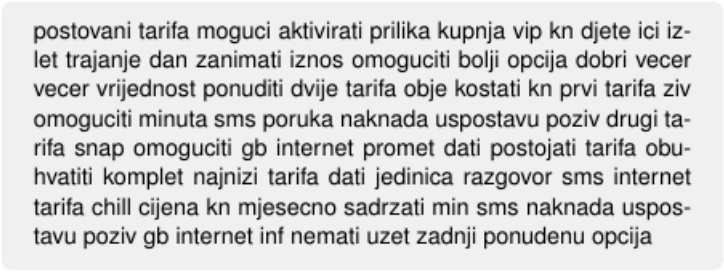
Riječi mogu biti pisane malim slovima, velikim slovima ili kombinacijom – najčešće početnim velikim slovom, a ostalim malim slovima. S obzirom da računalo takve riječi prepoznaje kao različite nizove znakova, izjednačit ćemo ih tako da svako slovo koji se pojavljuje u tekstu prebacimo na malo. S obzirom da se radi o digitalnim razgovorima, česta je pojava da korisnici potpuno preskaču korištenje dijakritičkih slova. U hrvatskom jeziku takva slova su č, ć, đ, š i ž i zamijeniti ćemo ih sa c, c, d, s i z respektivno.

Osim toga, hrvatski jezik je posebno složen jer ima deset vrsta riječi od kojih je čak pet promjenjivo. Želimo moći prepoznati istu riječ bez obzira na deklinaciju ili konjugaciju. Postupak prebacivanja bilo kojeg oblika riječi u izvorni (npr. za imenice je to padež nominativ, za glagole glagolsko vrijeme infinitiv) zovemo lematizacija. U tu svrhu koristimo rječnik [4].

U hrvatskom jeziku postoje riječi koje imaju različito značenje a u pisanom obliku izgledaju jednako (npr. kúpiti (dobiti u vlasništvo plaćanjem) i kùpiti (pribirati)). Uočavamo da ćemo i lematizacijom i uklanjanjem dijakritičkih znakova potencijalno neke različite riječi izjednačiti (npr. žuriti (ubrzano se kretati) i zuriti (gledati napadno u nešto)). Treba naglasiti da smo prije same analize teksta svjesni da računalo takve riječi neće znati razlikovati.

U pripremi teksta su se koristili i regularni izrazi kojima smo iz teksta uklonili određene vrste nizova znakova. S obzirom da će se u razgovorima koje analiziramo potencijalno pojaviti osjetljivi, privatni podaci korisnika A1 mreže, tekst ćemo anonimizirati na način da uklonimo sve brojeve telefona, prezimena, OIB-e, kućne i email adrese korisnika, kao i adrese web stranica.

Slika 2.1: Primjer razgovora prije čišćenja teksta



Slika 2.2: Primjer razgovora nakon čišćenja teksta

Uočit ćemo i veliki broj riječi koje se pojavljuju u gotovo svakom razgovoru ali ne daju nikakvu informaciju o temi razgovora, tj. neinformativne su. One su često veznici, ali mogu biti i bilo koji drugi tip riječi. Kreiramo popis takvih riječi i prije analize ih uklonimo iz teksta.

Točke, upitnike, uskličnike i oznake za novi redak ćemo iskoristiti kao informaciju o kraju rečenice kada nam takva informacija bude potrebna, a nakon toga ćemo ukloniti svu interpunkciju iz teksta.

U konačnici, razgovor s početka poglavlja prelazi u 2.2. Uočavamo da je za čovjeka, razgovor postao nečitljiv.

2.2 Reprezentacija teksta

Da bi mogli koristiti matematičke modele u analizi teksta, definirat ćemo strukturu tako što ćemo kreirati vektore koji odgovaraju tekstu. Jedinicu teksta, koju zovemo dokument, ćemo shvatiti kao vreću (skup s ponavljanjima) n-grama.

U našoj primjeni, definicija dokumenta nije uvijek ista. Imamo 3132 razgovora svaki od kojih želimo svrstati u jednu od unaprijed definiranih kategorija. U tu svrhu, dokument definiramo kao jedan chat razgovor. Međutim, jedan od ciljeva ovog rada je usporediti kako detekcija fraza i ključnih riječi utječe na točnost klasifikacije. Kod klasifikacije tekstualnih dokumenata, najčešće samo nekoliko ključnih pojmova daje odgovor na pitanje koja je tema dokumenta. Želimo pronaći ključne pojmove za svaki razgovor. Pretpostavljamo da promatranje fraza ima više smisla u kontekstu jedne rečenice, pa kada budemo radili detekciju fraza i ključnih riječi dokument će biti rečenica. Tada imamo ukupno 190099 dokumenata (rečenica). Dokument ćemo nekada zvati i primjerom.

N-gram je bilo koja uređena n-torka riječi ili znakova koji se pojavljuju u nekoj danoj kolekciji tekstualnih dokumenata. Mi ćemo promatrati n-grame riječi. Niz znakova prepoznajemo kao riječ ako je odvojen od ostatka razmakom. U ovom radu, frazom ćemo zvati uređeni par ili trojku riječi (bigram ili 3-gram) koje se često pojavljuju zajedno i zajedno možda poprimaju novo značenje.

Svaki dokument odgovara vektoru frekvencija n-grama i riječi – n-grame i riječi ćemo zajedno nekada zvati pojmovima. Bazu vektorskog prostora čine vektori koji odgovaraju tim pojmovima. To su vektori čija je dimenzija jednaka ukupnom broju dokumenata u kolekciji, a na čijoj i-toj poziciji piše frekvencija pojavljivanja danog pojma u i-tom dokumentu. Dimenzija svakog od vektora koji odgovaraju dokumentima je zbog toga jednaka ukupnom broju pojmova u kolekciji.

U našoj kolekciji pronalazimo ukupno 8085 različitih samostalnih riječi. Osim toga, promatramo n-grame od 2 ili 3 riječi. Broj n-grama koje ćemo promatrati u modelima ovisi o definiciji dokumenta. Kada jedan dokument bude odgovarao jednom cijelom razgovoru, bigrama će biti 70730, a 3-grama 112412, što zajedno sa samostalnim riječima daje ukupno 191227 pojmova. Jasno je da ćemo kod detekcije fraza, gdje je dokument rečenica, imati manji broj n-grama. Naime, u ovom slučaju kao n-grame nećemo prepoznati nizove riječi koje jesu uzastupne, ali su sada odvojene u dva različita dokumenta – rečenice. Imamo 61333 bigrama (2-grama) i 85554 3-grama, ukupno 154972 pojmova.

Opisane vektore zajedno promatramo u matrici u kojoj svaki redak odgovara jednom dokumentu, a stupac pojmu. Na mjestu (i, j) piše frekvencija pojavljivanja j-tog pojma u i-tom dokumentu.

Dakle, matrica koju ćemo koristiti za klasifikaciju je matrica X_1 tipa 3132×191227, a ona za pronalazak fraza, X_2, tipa je 190099×154972. Označimo $n_1 = 3132$, $n_2 = 190099$, $m_1 = 191227$, $m_2 = 154972$. Retke u matrici ćemo označavati $x^{(i)}$, stupce x_j, za $i = 1, ..., n_1$ ili $i = 1, ..., n_2$ odnosno $j = 1, ..., m_1$ ili $j = 1, ..., m_2$, ovisno o definiciji dokumenta.

Možemo uočiti i da su naše matrice tzv. rijetke matrice. Rijetke matrice su matrice koje su većinom ispunjene nulama. U slučaju matrice X_1, na 99.94% mjesta je 0, a u X_2 na čak 99.99% mjesta.

Promotrimo sada sljedeće 4 rečenice koje ćemo shvatiti kao dokumente.

1. Uređaj se može kupiti preko webshopa.
2. Htio bih preko webshopa kupiti osnovnu mobilnu tarifu.
3. Koju tarifu je moguce aktivirati prilikom kupnje vip bona od 50 kn?
4. Tarifa Za Ziv bi omogućila 500 minuta/sms poruka, a tarifa Za Snap 2 GB internet prometa.

Nakon opisane pripreme teksta one će izgledati ovako:

1. uredaj kupiti preko webshopa
2. preko webshopa kupiti osnovni mobilan tarifa
3. tarifa moguci aktivirati prilika kupnja vip kn
4. tarifa ziv omoguciti minuta sms poruka tarifa snap gb internet promet

U ovoj kolekciji dokumenata pronalazimo ukupno 23 riječi. Radi primjera, sada ćemo promatrati samo samostalne riječi, dakle, bigrame i 3-grame nećemo uzimati u obzir. Dobivena matrica frekvencija je:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 2 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Stupci redom odgovaraju pojmovima *aktivirati, bon, gb, internet, kn, kupiti, kupnja, minuta, mobilan, moguci, omoguciti, osnovni, poruka, preko, prilika, promet, sms, snap, tarifa, uredaj, vip, webshopa, ziv*, a reci navedenim rečenicama.

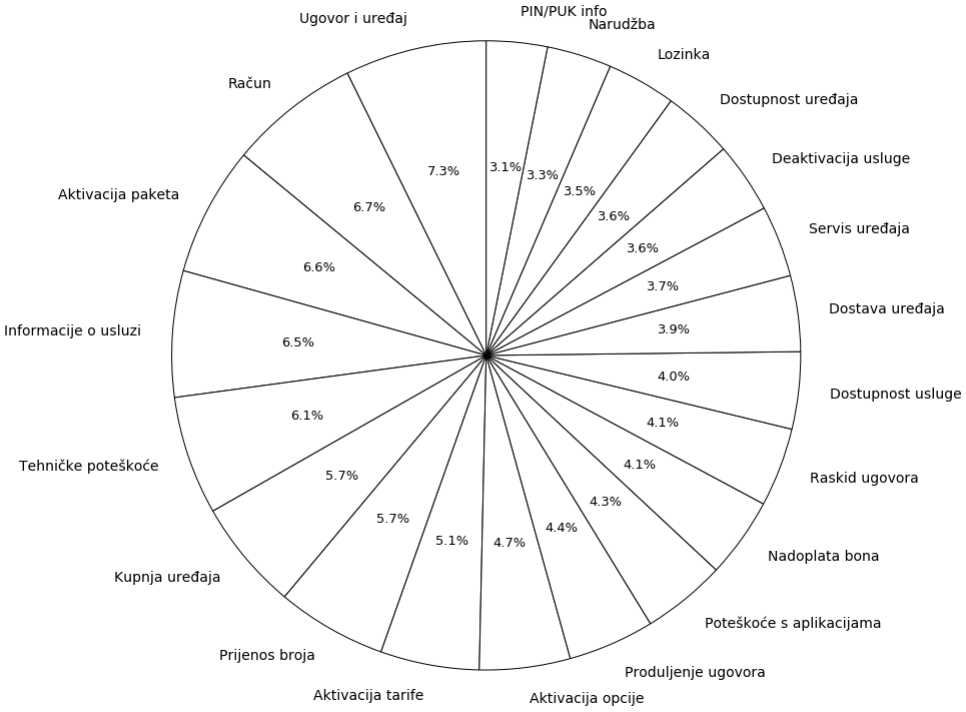
2.3 Klasifikacija

Želimo razgovore podijeliti u dvadeset i jednu unaprijed definiranu temu:

- Aktivacija opcije
- Aktivacija paketa
- Aktivacija tarife
- Deaktivacija usluge
- Dostava uređaja
- Dostupnost uređaja
- Dostupnost usluge
- Informacije o usluzi
- Kupnja uređaja
- Lozinka
- Nadoplata bona
- Narudžba
- PIN/PUK info
- Poteškoće s aplikacijama
- Prijenos broja
- Produljenje ugovora
- Raskid ugovora
- Račun
- Servis uređaja
- Tehničke poteškoće
- Ugovor i uređaj

Algoritmi nadziranog strojnog učenja (*engl. supervised machine learning*) promatraju skup označenih primjera i iz njega uče generalizirati. Takav skup podataka sadrži ulazne (nezavisne) varijable, čije vrijednosti ćemo zvati značajkama (*engl. features*) i izlaznu (zavisnu) varijablu čija je vrijednost oznaka (*engl. label*). Problem prepoznajemo kao klasifikacijski kada je zavisna varijabla kategorijska. Kod nadziranog strojnog učenja, prvo ćemo podijeliti dani skup podataka u nekom omjeru na skup podataka za učenje (treniranje) i skup podataka za validaciju. Biramo omjer 3:1, pa tako 2505 razgovora smjestimo u skup za učenje, a 627 u skup za validaciju. Distribucija tema prije podjele je na slici 2.3. Distribucija tema u skupu podataka za učenje je na slici 2.4, a na skupu za validaciju na slici 2.5. Nakon što model naučimo na skupu podataka za učenje, provjerit ćemo kako se taj model ponaša na novim podacima, tj. skupu za validaciju. Podsjetimo se da je u ovom slučaju dokument jedan cijeli razgovor.

U ovom radu ćemo promatrati tri različita klasifikacijska modela: logističku regresiju, multinomijalni naivni Bayesov klasifikator i stroj potpornih vektora. U objašnjenju ovih klasifikatora koristili smo definicije i tvrdnje iz [3] te bilješke s kolegija Strojno učenje na Fakultetu elektrotehnike i računarstva.



Slika 2.3: Distribucija tema razgovora

PIN/PUK info	2.8%	Dostupnost usluge	4.0%	Prijenos broja	5.6%
Narudžba	3.2%	Raskid ugovora	4.1%	Kupnja uređaja	5.8%
Lozinka	3.4%	Nadoplata bona	4.2%	Tehničke poteškoće	6.3%
Dostupnost uređaja	3.8%	Poteškoće s aplikacijama	4.5%	Informacije o usluzi	6.4%
Deaktivacija usluge	3.8%	Produljenje ugovora	4.6%	Aktivacija paketa	6.5%
Servis uređaja	3.8%	Aktivacija opcije	4.6%	Račun	6.6%
Dostava uređaja	3.9%	Aktivacija tarife	4.8%	Ugovor i uređaj	7.4%

Slika 2.4: Distribucija tema razgovora - učenje

PIN/PUK info	2.2%	Dostupnost usluge	4.0%	Prijenos broja	5.3%
Narudžba	2.9%	Raskid ugovora	4.3%	Kupnja uređaja	5.9%
Lozinka	2.9%	Nadoplata bona	4.3%	Tehničke poteškoće	6.1%
Dostupnost uređaja	3.5%	Poteškoće s aplikacijama	4.6%	Informacije o usluzi	6.2%
Deaktivacija usluge	3.7%	Produljenje ugovora	4.6%	Aktivacija paketa	6.9%
Servis uređaja	3.7%	Aktivacija opcije	4.8%	Račun	7.2%
Dostava uređaja	3.8%	Aktivacija tarife	5.1%	Ugovor i uređaj	8.1%

Slika 2.5: Distribucija tema razgovora - validacija

Označimo sa $\mathcal{X}$ skup svih primjera za učenje: $\mathcal{X} = \{x^{(i)} : i = 1, ..., n\}$.

Logistička regresija

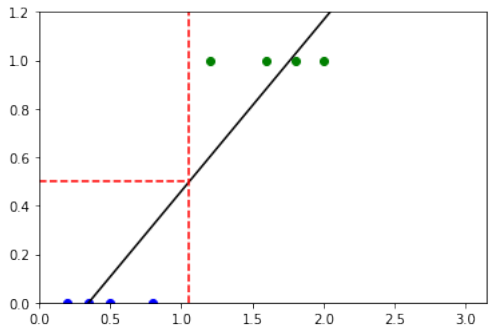
Za početak, promatramo problem binarne klasifikacije. Želimo primjere klasificirati u jednu od dvije klase, 0 ili 1, te za to koristiti linearni model. Najčešće korišten linearni model jest model linearne regresije 1.2 koji hiperravninu

$$h_1((x_1, x_2, ..., x_m); (\theta_0, \theta_1, ..., \theta_m)) = \theta_0 + \theta_1 x_1 + ... + \theta_m x_m \tag{2.1}$$

prilagođava zadanim točkama. Ovdje su $(\theta_1, ..., \theta_m) = \theta$ i θ_0 parametri modela. Linearnu regresiju možemo iskoristiti za klasifikaciju tako da granicu između klasa definiramo kao hiperravninu $\theta_0 + \theta_1 x_1 + ... + \theta_m x_m = 0.5$. Tada je klasifikacijski model

$$h_2((x_1, x_2, ..., x_m); (\theta, \theta_0)) = \mathbb{1}_{\{\theta_0 + \theta_1 x_1 + ... + \theta_m x_m \geq 0.5\}}.$$

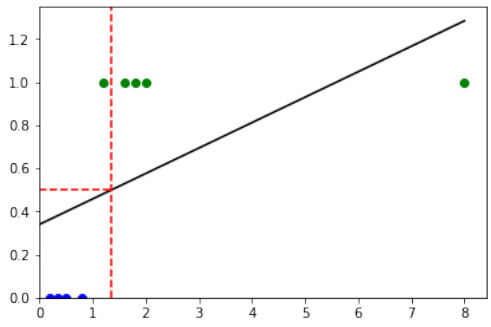
Međutim, takav model ima više nedostataka.



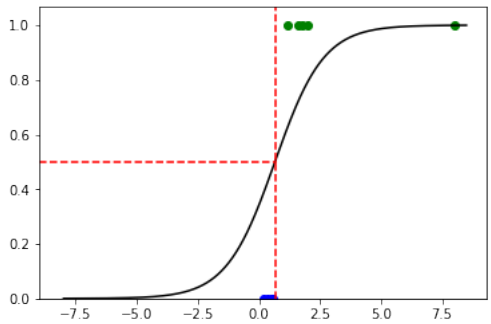
Slika 2.6: Klasifikacija linearnom regresijom

Primjer za $m = 1$ prikazan je na slici 2.6. Vertikalna iscrtkana linija predstavlja granicu između klasa, tako da svi primjeri s njene desne strane budu klasificirani u klasu 1, dok primjere s lijeve strane kalsificiramo s 0. Na slici 2.7 uočavamo da kod podataka koji udstupaju može nastati problem. Samo jedan primjer koji odstupa od ostalih je uveo velike promjene u model i radi toga dolazi do greške. Drugi problem je što h_1 može poprimati točke iz cijelog $\mathbb{R}$ pa nema probabilističku interpretaciju. Zbog toga ćemo uvesti nelinaernu aktivacijsku funkciju

$$\sigma(\alpha) = \frac{1}{1 + exp(-\alpha)}.$$



Slika 2.7: Klasifikacija linearnom regresijom



Slika 2.8: Klasifikacija logističkom regresijom

Funkciju σ zovemo logistička (sigmoidalna) funkcija. Model logističke regresije glasi:

$$h((x_1, x_2, ..., x_m); \theta) = \sigma(\theta_0 + \theta_1 x_1 + ... + \theta_m x_m) =$$

$$= \frac{1}{1 + exp(-\theta_0 - \theta_1 x_1 - ... - \theta_m x_m)} = \mathbb{P}(y = 1|(x_1, x_2, ..., x_m))$$

Dakle, izlaz modela interpretiramo kao vjerojatnost da primjer pripada klasi 1. Klasifikacija primjera sa slike 2.7 logističkom regresijom je na slici 2.8.

Definirati ćemo funkciju pogreške ovog modela kao:

$$E((\theta_0, \theta_1, ..., \theta_m)|(x_1, x_2, ..., x_m)) = \sum_{i=1}^{n} \ln(\exp(-y_i(x^{(i)})^T \theta + \theta_0)) + 1) \tag{2.2}$$

Želimo pronaći parametre θ i θ_0 koji minimiziraju 2.2. Međutim, kod algoritama strojnog učenja lako može doći do problema prenaučenosti (*engl. overfitting*). Složeniji model

će uvijek bolje opisati podatke na kojima je učio ali neće dobro generalizirati. Zbog toga se najčešće neće tako dobro ponašati na novim podacima. Uvodimo regularizaciju kojom kažnjavamo složenije modele. Regularizacijskim parametrom C određujemo koliko ih kažnjavamo. Funkciju pogreške 2.2 množimo sa C i dodajemo joj kvadrat norme vektora parametara. Sada funkciju pogreške definiramo s:

$$E((\theta_0, \theta_1, ..., \theta_m)|(x_1, x_2, ..., x_m)) = \frac{1}{2}||\theta||^2 + C\sum_{i=1}^{n} \ln(\exp(-y_i(x^{(i)})^T\theta + \theta_0)) + 1) \qquad (2.3)$$

Primjetimo da će regularizacija biti jača što je C manji, jer tako smanjujemo "doprinos" funkcije 2.2. Nije moguće pronaći formulu za minimum ove funkcije u zatvorenoj formi. Zbog toga ćemo optimizaciju provesti iterativno. Međutim, naš problem nije binaran. Zavisna varijabla je tema razgovora i ukupno ima dvadeset i jednu kategoriju. Koristiti ćemo shemu jedan-naspram-ostali (*engl. one-vs-rest*), što znači da ćemo napraviti binarnu logističku klasifikaciju dvadeset i jedan put. Svaki od tih puta ćemo jednu klasu označiti sa 1, dok će sve ostale imati oznaku 0. Treba naglasiti da prije korištenja ovog modela pretpostavljamo da su primjeri (dokumenti) međusobno nezavisni, te da nema visoke korelacije među značajkama.
Logistička regresija je linearni diskriminativni model jer modelira granicu između klasa. S druge strane, model može biti generativan tj. može modelirati nastajanje podataka iz zajedničke distribucije. Jedan takav model je naivni Bayesov klasifikator.

Multinomijalni naivni Bayesov klasifikator

Naivni Bayesov klasifikator je probabilistički model zasnovan na Bayesovom teoremu 1.2.3. Bayesov teorem daje formulu za računanje vjerojatnosti događaja H_i uz uvjet da se dogodio događaj A, gdje je $(H_i, i = 1, 2, ..., n)$ potpun sistem događaja. U našem slučaju, za svaki primjer (dokument) računat ćemo vjerojatnost klase uz uvjet da se dogodio taj primjer. Neka je $H_i = x^{(i)}$, vektor koji odgovara jednom primjeru, dok je $A = Y$ varijabla klase. Multinomijalni naivni Bayesov klasifikator dodatno pretpostavlja da podaci dolaze iz multinomijalne razdiobe. Multinomijalna razdioba je poopćenje Bernoullijeve. $Y = (Y_1, ..., Y_{21})^T$ može poprimiti jednu od 21 klasa. Prikazujemo je kao vektor indikatorskih varijabli $y = (y_1, ..., y_{21})^T$, gdje je $y_k = 1$ kada je ishod k-ta klasa, inače 0. Uz to je $\sum_k y_k = 1$. Ako označimo vjerojatnosti $\mathbb{P}(Y_k = 1) = \theta_k$ tada je razdioba dana s

$$\mathbb{P}(Y = y|(\theta_1, ..., \theta_{21})^T) = \prod_{k=1}^{21} \theta_k^{y_k}.$$

Ovaj klasifikator dodatno pretpostavlja uvjetnu nezavisnost događaja $x_j^{(i)}$ uz uvjet Y:

$$P(x_j^{(i)}|Y, x_1^{(i)}, \ldots, x_{j-1}^{(i)}, x_{j+1}^{(i)}, \ldots, x_m^{(i)}) = P(x_j^{(i)}|Y).$$

Ovaj uvjet najčešće (pa tako ni u našoj primjeni) nije ispunjen, ali unatoč tome u praksi model dobro funkcionira. Sada imamo:

$$\mathbb{P}(Y|x^{(i)}) = \frac{\mathbb{P}(Y)\mathbb{P}(x^{(i)}|Y)}{\mathbb{P}(x^{(i)})} = \frac{\mathbb{P}(Y)\prod_{j=1}^{m}\mathbb{P}(x_j^{(i)}|Y)}{\mathbb{P}(x^{(i)})}.$$

S obzirom da je vjerojatnost $\mathbb{P}(x^{(i)})$ jednaka neovisno o klasi y, klasifikaciju će raditi sljedeća formula:

$$\hat{y} = \arg\max_y \mathbb{P}(Y = y)\prod_{j=1}^{m}\mathbb{P}(x_j^{(i)}|Y = y).$$

Dodatno ćemo uvesti Laplaceovo zaglađivanje, tj. za svaki $k = 1, ..., 21$, distribuciju parametriziramo vektorom $\theta_{Y_k} = (\theta_{Y_k 1}, \theta_{Y_k 2}, ..., \theta_{Y_k m})$ kojeg procjenjujemo zaglađenom verzijom maksimalne vjerodostojnosti:

$$\theta_{Y_k i} = \frac{N_{Y_k i} + \alpha}{N_{Y_k} + \alpha m},\ \forall i \in \{1, 2, ..., m\}.$$

, gdje $N_{Y_k i}$ broj pojavljivanja i-tog pojma u primjeru iz k-te klase, a N_{Y_k} ukupan broj pojavljivanja svih pojmova u primjeru iz k-te klase, dakle $N_{Y_k} = \sum_{i=1}^{m} N_{Y_k i}$. Parametar $\alpha \geq 0$ služi za tzv. zaglađivanje i spriječava nul vjerojatnosti kod pojmova koji se nisu pojavili u primjeru. Na ovaj način smo procijenili vjerojatnosti $\mathbb{P}(x_j^{(i)}|Y)$, tj. $\mathbb{P}(x_j^{(i)}|Y) = \theta_{Yi}$. $\mathbb{P}(Y)$ procjenjujemo maksimalnim aposteriornim (MAP) procjeniteljem.

Stroj potpornih vektora (*engl. Support vector machine, SVM*)

Ponovno promatramo slučaj binarne klasifikacije, sada klase označimo -1 i 1. Stroj potpornih vektora je u svojoj osnovnoj formi linearan model. Bavi se problemom pronalaska hiperravnine (kao u 2.1) koja će podijeliti prostor primjera na najbolji mogući način. Definiramo marginu kao udaljenost između te hiperravnine i najbližih primjera. SVM pronalazi maksimalnu marginu. Geometrijski, hiperravnina će biti simetrala spojnice konveksnih ljusaka dviju klasa. Te, hiperravnini najbliže, primjere ćemo zvati potpornim vektorima.

Označimo $\theta = (\theta_1, \theta_2, ..., \theta_m)$. Udaljenost primjera $x^{(i)}$ od hiperravnine je $\frac{1}{||\theta||} y_i(\theta^T x^{(i)} + \theta_0)$. Tražimo hiperravninu maksimalne margine:

$$\arg\max_{\theta,\theta_0} \{\frac{1}{||\theta||} \min_i \{y_i \theta^T x^{(i)} + \theta_0\}\}. \tag{2.4}$$

Vektor (θ, θ_0) ćemo skalirati tako da za potporne vektore vrijedi $y_i(\theta^T x^{(i)} + \theta_0) = 1$. Ako pretpostavimo linearnu odvojivost primjera, pričamo o tvrdoj margini. Tada ovaj model ne dopušta ulazak u marginu ni pogrešnu klasifikaciju, tj. mora biti ispunjeno:

$$y_i(\theta^T x^{(i)} + \theta_0) \geq 1, \quad i = 1, ..., n. \tag{2.5}$$

Problem optimiziranja svodi se na:

$$\arg\max_{\theta,\theta_0} \frac{1}{||\theta||} \iff \arg\min_{\theta,\theta_0} \frac{1}{2}||\theta||^2 \tag{2.6}$$

uz ograničenja

$$y_i(\theta^T x^{(i)} + \theta_0) \geq 1, \quad i = 1, ..., n \tag{2.7}$$

Ograničenja ubacujemo u Lagrangeovu funkciju (definiranu u 1.3) i dobivamo:

$$L(\theta, \theta_0, \alpha) = \frac{1}{2}||\theta||^2 - \sum_{i=1}^{n} \alpha_i(y_i(\theta^T x^{(i)} + \theta_0) - 1), \tag{2.8}$$

gdje je $\alpha = (\alpha_1, ..., \alpha_n)$ vektor Lagrangeovih multiplikatora i vrijedi $\alpha_i \geq 0, i = 1, ..., n$. Minimizaciju funkcije 2.8 zovemo primarnim problemom. Međutim, problem možemo zapisati i na drugi način. Deriviranjem funkcije 2.8 po parametrima θ i θ_0 i izjednačavanjem s nulom dobivamo još dva uvjeta:

$$\theta = \sum_{i=1}^{n} \alpha_i y_i x^{(i)} \tag{2.9}$$

$$0 = \sum_{i=1}^{n} \alpha_i y_i.$$

Njihovim korištenjem iz 2.8 možemo ukloniti θ i θ_0 i tako dobiti dualnu Lagrangeovu funkciju:

$$\tilde{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (x^{(i)})^T x^{(j)} \tag{2.10}$$

uz ograničenja

$$\alpha_i \geq 0, \quad i = 1, ..., n \tag{2.11}$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

Dualni problem je maksimizacija dualne Lagrangeove funkcije 2.10 uz ograničenja 2.11.

U slučaju kada primjeri nisu odvojivi, ovako definiran problem nema rješenje. Kada ga i ima, lako dolazi do prenaučenosti. Zato želimo dopustiti ulazak u marginu i pogrešnu klasifikaciju, pa uvodimo pojam meke margine. Ograničenje 2.7 prelazi u:

$$y_i(\theta^T x^{(i)} + \theta_0) \geq 1 - \xi_i, \quad i = 1, ..., n \tag{2.12}$$

gdje $\xi_i \geq 0$ govori koliko je primjer ušao u marginu. Primarna Lagrangeova funkcija prelazi u:

$$L(\theta, \theta_0, \alpha, \beta, \xi) = \frac{1}{2}\|\theta\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i(y_i(\theta^T x^{(i)} + \theta_0) - 1 + \xi_i) - \sum_{i=1}^{n}\beta_i\xi_i, \tag{2.13}$$

gdje su $\alpha_i \geq 0$ i $\beta_i \geq 0$ Lagrangeovi multiplikatori. Dualnu Lagrangeovu funkciju dobivamo kada deriviramo 2.13 po θ, θ_0 i ξ_i i izjednačimo s nulom, te uvrstimo. Rezultat je funkcija koja je jednaka 2.10, ali ograničenja su drugačija:

$$\begin{gathered} 0 \leq \alpha_i \leq C, \quad i = 1, ..., n \\ \sum_{i=1}^{n}\alpha_i y_i = 0 \end{gathered} \tag{2.14}$$

Dualni zapis stroja potpornih vektora možemo definirati i kao nelinearan problem. Uočimo da se u 2.10 pojavljuje skalarni produkt vektora $x^{(i)}$ i $x^{(j)}$, koji predstavljaju i-ti odnosno j-ti primjer. Sličnost dva primjera možemo računati i koristeći neku nelinearnu funkciju. Mi ćemo koristiti tzv. jezgrenu funkciju $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, gdje je $\mathcal{X}$ prostor primjera. Tako ćemo umjesto da računamo težine uz vektor značajki, računati sličnost dvaju primjera. Jezgrena funkcija je i mjera sličnosti ako zadovoljava:

1. $K(x, x) = 1, \forall x \in \mathcal{X}$;
2. $0 \leq K(x, x') \leq 1, \forall x, x' \in \mathcal{X}$;
3. $K(x, x') = K(x', x), \forall x, x' \in \mathcal{X}$.

Jezgrena funkcija se zove radijalna bazna funkcija (i tada pričamo o RBF jezgri) ako je ona definirana kao funkcija neke norme. Mi ćemo koristiti Gaussovu RBF jezgru:

$$K(x, x') = \exp(-\frac{\|x - x'\|^2}{2\sigma^2}), \tag{2.15}$$

gdje su x, x' primjeri, a σ^2 je parametar koji predstavlja širinu pojasa. Definiramo preciznost $\gamma = \frac{1}{2\sigma^2}$ pa (2.15) postaje

$$K(x, x') = \exp(-\gamma\|x - x'\|^2) \tag{2.16}$$

Konačni dualni optimizacijski problem je:

$$\tilde{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x^{(i)}, x^{(j)}). \tag{2.17}$$

uz ograničenja 2.14.

Kao i u slučaju logističke regresije, koristimo shemu jedan-naspram-ostali.

Poglavlje 3

Detekcija fraza i ključnih riječi

3.1 Faktor lokalnih izuzetaka

S obzirom da nemamo označenu bazu fraza hrvatskog jezika, problemu detekcije pristupamo kao problemu nenadziranog strojnog učenja. Nenadzirano strojno učenje (*engl. unsupervised machine learning*) promatra neoznačeni skup podataka. Nisu nam poznate vrijednosti zavisne varijable po kojoj bi algoritam mogao naučiti pravilo generaliziranja, pa pokušava grupirati dane podatke promatrajući sličnosti i uzorke bez nadzora. Kod ovakvih modela nema smisla dijeliti podatke na skupove za učenje i validaciju jer ne znamo računati točnost dobivenog.

Izuzetak (*engl. outlier*) je podatak koji se značajno razlikuje od ostatka. Problemu detekcije ključnih riječi i fraza pristupamo kao problemu detekcije lokalnih izuzetaka. Faktor lokalnih izuzetaka (*engl. local outlier factor, LOF*) računa odstupanje gustoće danog primjera s obzirom na zadani broj susjeda $k \in \mathbb{N}$. Gustoća se procjenjuje računajući udaljenosti od k najbližih susjeda a primjer proglašavamo izuzetkom ako ima značajno manju gustoću od svojih najbližih susjeda. Slijede formalne definicije (iz [2]).

Sada ćemo vektore koji odgovaraju pojmovima shvatiti kao primjere, a one koji odgovaraju dokumentima kao značajke. Drugim riječima, promatramo matricu X_2^T. Neka je $\mathcal{X}$ skup svih primjera.

Definicija 3.1.1. *Neka je p primjer. Za $k \in \mathbb{N}$, neka su $p_1, p_2, ..., p_k \in \mathcal{X}$ redom najbliži primjeri primjeru p, tj. $d(p_1, p) \leq d(p_2, p) \leq ... \leq d(p_k, p)$.* ***k-udaljenost*** *od p, u oznaci $d_k(p)$, definiramo kao udaljenost $d(p_k, p)$.*

Definicija 3.1.2. *Za danu k-udaljenost d_k primjera p,* ***k-udaljeno susjedstvo*** *od p sadrži svaki primjer $q \in \mathcal{X}$ čija udaljenost do p nije veća od k-udaljenosti tj.*

$$S_{d_k(p)}(p) = \{q \in \mathcal{X} \backslash \{p\} : d(p, q) \leq d_k(p)\}$$

Primjere q zovemo k-najbližim susjedima od p.

Definicija 3.1.3. *Neka je* $k \in \mathbb{N}$. ***Duljina dohvativosti*** *primjera p s obzirom na primjer o definirana je s* $dd_k(p, o) = max\{d_k(o), d(p, o)\}$.

Definicija 3.1.4. ***Gustoća lokalne dohvatljivosti*** *od p definirana je kao*

$$g_k(p) = 1/\frac{\sum_{o \in S_k(p)} dd_k(p, o)}{|S_k(p)|}$$

gdje je k parametar koji određuje minimalan broj primjera u skupu.

Definicija 3.1.5. ***Faktor lokalnih izuzetaka*** *primjera p definiran je kao*

$$LOF_k(p) = \frac{\sum_{o \in S_k(p)} \frac{g_k(o)}{g_k(p)}}{|S_k(p)|}$$

Faktor lokalnih izuzetaka je dakle aritmetička sredina omjera lokalne dohvatljivosti primjera p i njegovih k-najbližih susjeda. Njegova vrijednost nam daje informaciju o razini kojom možemo primjer p smatrati izuzetkom.

3.2 Pristupi računanju težina pridruženih pojmovima u dokumentu

Frekvencije pojavljivanja pojma ne daju uvijek dovoljno dobru informaciju o važnosti tog pojma u dokumentu. Do problema može doći u situaciji kada se jedan pojam pojavljuje u zadanom dokumentu ali je čest i u cijeloj kolekciji. Tada bi se promatranjem same frekvencije dalo naslutiti da je pojam ključan za dokument. Međutim, s obzirom da se pojavljuje i u većini ostalih dokumenata, zaključujemo da ne može biti toliko informativan za jedan dani dokument.

Do drugog problema može doći zbog različite norme vektora. Naime, vektori koji predstavljaju duže razgovore će imati veće norme od kraćih razgovora. Kod učenja modela logističke regresije, stroja potpornih vektora te faktora lokalnih izuzetaka moglo bi biti ključno normalizirati vektore. Naime, takvi modeli bi u slučaju bez normalizacije veću važnost dodijelili većim vektorima.

Množenjem tri komponente, frekvencije pojma u dokumentu, inverzne frekvencije dokumenta i normalizacije, dobit ćemo novu vrijednost. Zanima nas hoće li klasifikacija u konačnici biti bolja s ovakvim novim vrijednostima vektora.

Računanju svake od navedene tri komponente možemo pristupiti na više načina koji su predstavljeni u [7]. Frekvenciju pojma u dokumentu možemo promatrati binarno, 1

ako se pojam pojavio, 0 inače. Matricu dobivenu takvom mjerom ćemo označavati bxx. Naravno, možemo i promatrati standardne frekvencije tj. broj puta koji se pojam pojavio u zadanom dokumentu, u oznaci txx. Označimo s tf_i, $i = 1, ..., n$ vektor frekvencija i-tog dokumenta. Uočimo da je, u slučaju kada problemu pristupamo kao problemu same klasifikacije, $tf_i = x^{(i)}$ i-ti redak matrice X_1. Definirat ćemo i tzv. proširenu normaliziranu frekvenciju formulom:

$$nxx_i = 0.5 + 0.5\frac{tf_i}{\max(tf_i)},$$

gdje je nxx matrica, a nxx_i njen i-ti redak. Nadalje, inverzna frekvencija dokumenta se najčešće računa kao logaritam omjera ukupnog broja dokumenata n i broja dokumenata u kojima se j-ti pojam pojavio n'_j za $j = 1, .., m$:

$$\log(\frac{n}{n'_j}).$$

Množenjem po točkama svakog retka matrice bxx sa tako dobivenom vrijednosti dobit ćemo matricu koju ćemo označavati s bfx Normalizaciji ćemo pristupiti tako da vektor po točkama dijelimo njegovom 2-normom. Matricu normaliziranih frekvencija iz txx ćemo označavati txc:

$$txc_i = \frac{tf_i}{||tf_i||}.$$

Konačno, najsloženija mjera koju ćemo koristiti (u literaturi se često zove TF-IDF) dobije se kada u obzir uzmemo sve tri komponente - frekvenciju, inverznu frekvenciju i normalizaciju. Matricu ćemo označavati tfc, a njen i-ti redak se dobije formulom:

$$tfc_i = \frac{tf_i \cdot \log(\frac{n}{n'})}{||tf_i||}.$$

Ovdje s $\log(\frac{n}{n'})$ označavamo vektor redak na čijoj je j-toj poziciji $\log(\frac{n}{n'_j})$ za $j = 1, ..., m$, a množenje vektora u brojniku je po točkama.

Poglavlje 4

Analiza rezultata

4.1 Usporedba predstavljenih modela

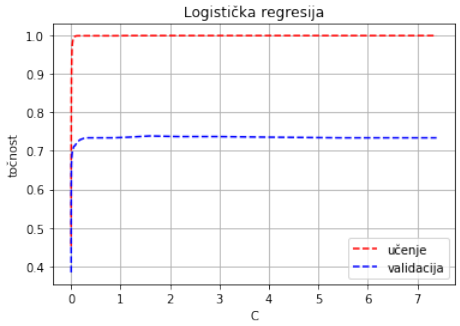
Predstavili smo više različitih pristupa klasifikaciji kao i pripremi podataka iz kojih će klasifikator učiti. Preostaje odabrati kombinaciju modela koja u konačnici daje najbolji rezultat. Točnost klasifikacije definiramo kao omjer broja točno klasificiranih dokumenata i ukupnog broja dokumenata.

U strojnom učenju, hiperparametar modela je parametar koji biramo prije učenja modela. Odabir hiperparametara je jedan od ključnih koraka kod korištenja ovih modela i rezultati se mogu bitno razlikovati ovisno o njemu. Konkretno u našem slučaju, logistička regresija prima parametar regularizacije, stroj potpornih vektora uz regularizaciju prima i preciznost, multinomijalni naivni Bayesov klasifikator prima parametar zaglađivanja, a faktor lokalnih izuzetaka broj susjeda koje želimo promatrati. Odabir hiperparametara ćemo napraviti postupkom unakrsne provjere (*engl. cross-validation*). Provjerit ćemo sve potencijalne vrijednosti parametra regularizacije, preciznosti, zaglađivanja i broja susjeda i naučiti svaki od modela dobivenih kombinacijama tih vrijednosti. Kao što smo već spomenuli, složeniji model će uvijek bolje opisati podatke koje je koristio za učenje, ali često neće dobro generalizirati. Zbog toga postupkom unakrsne provjere biramo onaj model koji daje najveću točnost na skupu podataka za validaciju. Ti podaci su do sada modelu bili nepoznati pa se možemo bolje pouzdati u takvu procjenu točnosti.

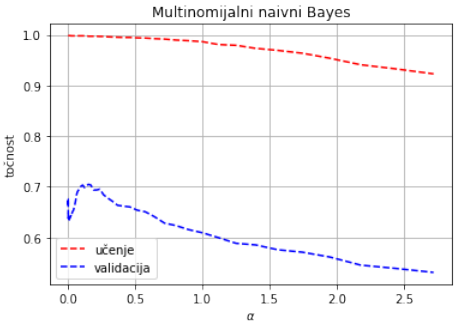
Krećemo s osnovnim pristupom, zanima nas koliko su točni klasifikacijski algoritmi kada ih učimo na matrici običnih frekvencija.

Unakrsnom provjerom kod logističke regresije pronalazimo optimalni parametar regularizacije. Kod multinomijalnog naivnog Bayesovog klasifikatora, tražimo parametar zaglađivanja. Unakrsnom provjerom po rešetci (*engl. grid search*) tražimo optimalnu kombinaciju parametara regularizacije i preciznosti za stroj potpornih vektora.

Na slici 4.1 vidimo točnost modela logističke regresije ovisno o parametru C i skupu



Slika 4.1: Točnost logističke regresije



Slika 4.2: Točnost Bayesovog klasifikatora

podataka na koji taj model primjenjujemo. Ta točnost brzo raste na oba skupa podataka. Vidimo da model na skupu podataka za učenje brzo dođe do potpune točnosti. Veća vrijednost parametra C daje manju regularizaciju i složeniji model. Najveća točnost ovog modela je 73.84% i ostvaruje se za C=1.59. Točnost naivnog Bayesovog klasifikatora ovisno o parametru α je na slici 4.2. Bez zaglađivanja, kada je $\alpha = 0$, model na podacima za učenje postiže potpunu točnost. Ovakvi modeli su jednostavniji što je α veći. Najveća postignuta točnost je 70.49% za α=0.15. Stroj potpornih vektora dolazi do 69.06% za C=238.34 i γ=0.001.

Sljedeće što nas zanima je kako će se klasifikatori ponašati ako ih naučimo na primjerima čije su značajke dobivene nekom od mjera predstavljenih u 3.2. Kao i u prethodnom slučaju, tražimo optimalne hiperparametre. Rezultati su na slici 4.3.

	LOGISTIČKA REGRESIJA	MULTINOMIJALNI NAIVNI BAYES	STROJ POTPORNIH VEKTORA
bxx	73.37% ($C = 0.19$)	68.58% ($\alpha = 0.14$)	68.10% ($C = 614.65, \gamma = 0.0002$)
txx	73.84% ($C = 1.59$)	70.49% ($\alpha = 0.15$)	69.06% ($C = 238.34, \gamma = 0.001$)
nxx		25.35% ($\alpha = 0.00$)	
bfx	73.84% ($C = 0.01$)	70.02% ($\alpha = 4.15$)	
tfc	73.04% ($C = 14.39$)	71.77% ($\alpha = 0.15$)	69.22% ($C = 238.34, \gamma = 0.002$)

Slika 4.3: Točnost klasifikacije

Primjećujemo da je najtočniji klasifikator logistička regresija i to kada koristimo matricu $txx = X_1$ ili matricu bfx. Dakle, nismo uspjeli povećati točnost klasifikacije u odnosu na klasifikaciju kada koristimo samo obične frekvencije iz X_1.

Jedan od najvećih problema kod naše analize je visoka dimenzionalnost matrica na kojima vršimo operacije. Iz tog razloga, neki se od modela nisu uspjeli izvršiti u razumno vrijeme. Uočimo da na primjer nedostaju rezultati klasifikacije logističkom regresijom i strojem potpornih vektora u slučaju korištenja težina dobivenih proširenom normaliziranom frekvencijom. Takve vrijednosti se bitno razlikuju od ostalih definiranih mjera jer su jedine koje daju matricu koja nije rijetka. Na većini pozicija u ovakvoj matrici više nisu nule, sada imamo vrijednosti 0.5. Treniranje modela na takvoj matrici računalno je zahtjevnije i sporije nego što je bilo do sada. Multinomijalni Bayesov klasifikator se pokazao kao najjednostavniji model klasifikacije koji smo koristili i jedini se u razumno vrijeme uspio naučiti na ovakvoj matrici, iako daje znatno manju točnost nego ostali pristupi.

Zanima nas kako detekcija fraza utječe na rezultate. Koristeći faktor lokalnih izuzetaka, pronalazimo pojmove koji odstupaju od ostatka i proglašavamo ih frazama. Pokušavat ćemo za razne vrijednosti broja susjeda k i odabrati onaj koji daje najveću točnost klasifikacije kada promatramo samo značajke koje su indeksirane pronađenim frazama. Nemamo način na koji bi provjerili točnost detekcije fraza, ali nakon učenja modela možemo pogledati koje n-grame smo proglasili frazama. Neki od primjera za broj susjeda $k = 25$ su: *dostupnost, pojasniti, prijavljen, informacija dati, postojeci operater, mogucnost provjeriti, obiteljski tarifa cijena, dobiti sms poruka, tjedan prije istek,....* Ukupan broj detektiranih fraza za $k = 25$ je 2538, pa dalje radimo s matricom tipa 3132×2538, što znači da smo broj stupaca matrice smanjili za čak 87%, a time smo značajno ubrzali učenje svih predstavljenih klasifikatora.

Rezultati ovakve klasifikacije su na slici 4.4.

	LOGISTIČKA REGRESIJA	MULTINOMIJALNI NAIVNI BAYES	STROJ POTPORNIH VEKTORA
bxx	71.45% ($k = 25, C = 0.26$)	68.74% ($k = 25, \alpha = 0.35$)	70.49% ($C = 92.42, \gamma = 0.001$)
txx	72.89% ($k = 23, C = 1.2$)	71.61% ($k = 26, \alpha = 0.19$)	69.38% ($C = 35.84, \gamma = 0.002$)
nxx	71.45% ($k = 27, C = 3.54$)	42.74% ($k = 25, \alpha = 3.91$)	
bfx	74.80% ($k = 25, C = 0.01$)	68.89% ($k = 29, \alpha = 3.36$)	72.89% ($C = 126.74, \gamma = 0.001$)
tfc	72.57% ($k = 25, C = 5.11$)	68.26% ($k = 25, \alpha = 0.05$)	

Slika 4.4: Točnost klasifikacije nakon detekcije fraza

Sada smo postigli nešto veću točnost u slučaju kada smo koristili logističku regresiju i težine pridružene pojmovima u dokumentu računali koristeći formulu

$$\log(\frac{n}{n'_j}) \cdot \mathbb{1}_{\{j\text{-ti pojam se pojavljuje u } i\text{-tom dokumentu}\}}, \tag{4.1}$$

gdje je $n = n_2 = 190099$, a n'_j broj dokumenata u kojima se j-ti pojam pojavljuje.

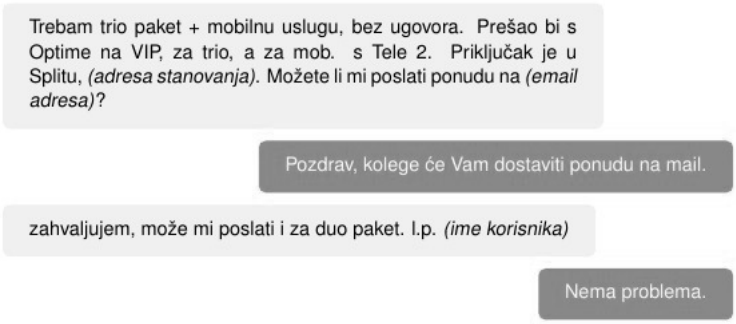
Uočimo i da se generalno logistička regresija pokazala kao najtočniji klasifikator za naš problem. Stroj potpornih vektora, koji je najsloženiji model od predstavljenih, za ovaj problem nije isplativo koristiti.

4.2 Zaključak

Konačno, možemo reći da je najbolji pristup klasifikaciji naših podataka sljedeći:

1: nauči faktor lokalnih izuzetaka na matrici X_2^T, gdje je dokument jedna rečenica i predstavlja ga stupac, i detektiraj ključne pojmove (fraze ili samostalne riječi)
2: iz matrice X_2 ukloni sve stupce osim onih koji su indeksirani detektiranim frazama ili ključnim riječima
3: sumiraj retke koji predstavljaju dokumente (rečenice) iz istog razgovora - sada je dokument jedan razgovor
4: transformiraj novodobivenu matricu tako da vrijednost na poziciji (i, j) računaš formulom 4.1
5: podijeli primjere na skupove za učenje i validaciju u omjeru 3:1
6: nauči logisitčku regresiju na matrici u kojoj su retci primjeri za učenje i testiraj na primjerima za validaciju

Vidjeli smo da ovakav algoritam na skupu podataka za validaciju daje točnost od 74.80%, što je najveća točnost koju smo postigli. Osim te činjenice, ovaj je algoritam zbog velikog smanjenja dimenzionalnosti detekcijom fraza i ključnih riječi među najefikasnijim predstavljenim algoritmima. Međutim, može se primijetiti da nismo puno povećali samu točnost u odnosu na prvi, najjednostavniji pristup klasifikaciji. Na slici 4.5 je primjer krivo klasificiranog razgovora. Oznake klasa iz našeg skupa podataka su pridružene razgovorima čitanjem istih. Čovjek je ovaj razgovor označio temom "Prijenos broja", jer je to prepoznao kao najrelevantniju temu, dok ga je naš model smjestio u klasu "Aktivacija paketa". Međutim, kada pročitamo razgovor koji je u pitanju, jasno nam je zašto je došlo do greške. Definirane klase nisu međusobno disjunktne i sami razgovori mogu imati više tema.



Slika 4.5: Krivo klasificirani razgovor

Na slikama 2.3, 2.4 i 2.5 smo redom prikazali udio svake od klasa u skupu svih podataka, te skupovima za učenje i validaciju. Pri tome smo pazili da iz svake klase u svim skupovima imamo dovoljno primjera. Posložimo li teme po učestalosti, uočavamo da je redoslijed isti neovisno koji od skupova promatramo. Točnost modela ovisno o klasi je u tablici na slici 4.6. Zadržavamo spomenuti redoslijed tema, čitajući prvo po stupcima, a zatim po recima, teme idu od najrjeđih prema najčešćima.

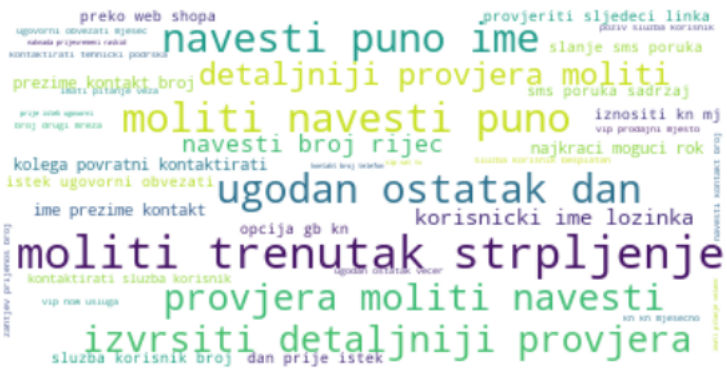
PIN/PUK info	72.41%	Dostupnost usluge	77.27%	Prijenos broja	64.86%
Narudžba	91.67%	Raskid ugovora	62.96%	Kupnja uređaja	53.13%
Lozinka	71.43%	Nadoplata bona	100%	Tehničke poteškoće	69.70%
Dostupnost uređaja	83.33%	Poteškoće s aplikacijama	52.17%	Informacije o usluzi	34.21%
Deaktivacija usluge	66.67%	Produljenje ugovora	48%	Aktivacija paketa	75.56%
Servis uređaja	75.86%	Aktivacija opcije	66.67%	Račun	80.39%
Dostava uređaja	66.67%	Aktivacija tarife	51.28%	Ugovor i uređaj	62.79%

Slika 4.6: Točnost modela ovisno o klasi

Na slikama 4.7, 4.8 i 4.9 su dane neke od najčešće detektiranih riječi i fraza.

Slika 4.7: Detektirane riječi

Slika 4.8: Detektirani bigrami



Slika 4.9: Detektirani 3-grami

4.3 Korišteni paketi

Za analizu teksta i implementaciju predstavljenih matematičkih modela koristio se programski jezik Python. Kod učitavanja i pripreme teksta koristila se velika količina Pythonovih paketa, kao što su pandas, numpy, string, re i drugi. S obzirom da su matrice koje predstavljaju tekst rijetke matrice velikih dimenzija, za njihovo definiranje koristimo sparse paket koji omogućuje spremanje podataka na efikasan način. Iz paketa scikit-learn [6] korišteni su klasifikatori (logistička regresija, multinomijalni naivni Bayesov klasifikator, stroj potpornih vektora) i faktor lokalnih izuzetaka. Grafovi su nacrtani koristeći matplotlib paket, a vizualizacije riječi na slikama 4.7, 4.8 i 4.9 paketom wordcloud.

Bibliografija

[1] D. Bakić, *Linearna algebra*, Sveučilište u Zagrebu, Matematički odsjek PMF-a, 2008.

[2] M. M. Breunig, H. Kriegel, R. T. Ng i J. Sander, *LOF: identifying density-based local outliers*, ACM sigmod record, ACM, 2000, str. 93–104.

[3] R. E. Fan, K.W. Chang, C. J. Hsieh, X. R. Wang i C. J. Lin, *LIBLINEAR: A Library for Large Linear Classification*, Journal of Machine Learning Research **9** (2008), 1871–1874.

[4] G. Igaly, *Rječnik hrvatskih jezika*, `https://github.com/gigaly/rjecnik-hrvatskih-jezika`, Datum pristupanja: ožujak 2019.

[5] F. Janjić, *Semantičko indeksiranje i klasifikacija dokumenata*, Sveučilište u Zagrebu, Matematički odsjek PMF-a, 2019., Diplomski rad.

[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot i E. Duchesnay, *Scikit-learn: Machine Learning in Python*, Journal of Machine Learning Research **12** (2011), 2825–2830.

[7] G. Salton i C. Buckley, *Term-Weighting Approaches in Automatic Text Retrieval.*, Information Processing and Management **24** (1988), br. 5, 513–523, `http://www.doc.ic.ac.uk/~jmag/classic/1988.Term-weighting%20approaches%20in%20automatic%20text%20retrieval.pdf`.

[8] N. Sarapa, *Teorija vjerojatnosti*, Školska knjiga, Zagreb, 2002.

Sažetak

Ovaj rad uspoređuje neke pristupe klasifikaciji digitalnih razgovora. Na početku su dani osnovni matematički pojmovi potrebni za razumijevanje tema obrađenih u radu. Nakon toga, objašnjavamo prilagodbu teksta matematičkim modelima te predstavljamo modele za klasifikaciju. Zatim tražimo ključne riječi i fraze u tekstu faktorom lokalnih izuzetaka i pokušavamo efikasnije prilagoditi težine pojmovima u dokumentu. Konačno, analiziramo dobivene rezultate.

Summary

In this work, we present and compare some approaches to classifying text, particularly digital conversations. The first part is dedicated to explaining the basic mathematical concepts necessary for the understanding of the topics explored in the paper. Next, we explain the text parsing techniques and introduce 3 classification models. We search for key words and phrases throughout the text using local outlier factor and we aim to assign weights more efficiently to the terms in the document. Finally, we analyze the results obtained.

Životopis

Rođena sam u Kiseljaku u Bosni i Hercegovini 07. studenog 1993. godine. Odrasla u Orebiću na poluotoku Pelješcu, gdje sam 2008. godine završila Osnovnu školu Petra Šegedina. Do 2012. godine pohađam opću gimnaziju u Srednjoj školi Petra Šegedina u Korčuli. Potom upisujem Preddiplomski studij matematike na Prirodoslovno-matematičkom fakultetu Sveučilišta u Zagrebu, kojeg završavam 2017. godine. Po završetku preddiplomskog studija, upisujem Diplomski sveučilišni studij Matematička statistika na istom fakultetu.