SVEUČILIŠTE U ZAGREBU

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1547

Primjena koncepta jezgrenih funkcija u dubinskoj analizi teksta

Dajana Buršić

Zagreb, srpanj 2005.

Zahvaljujem se sjajnoj mentorici i voditeljici

prof. dr.sc. Bojani Dalbelo-Bašić

i kolegi Domagoju Tomincu na uvijek spremnoj pomoći

SADRŽAJ

1 UVOD

Osnovni zadatak ovog diplomskog rada bio je proučiti primjenu koncepta jezgrenih funkcija u dubinskoj analizi teksta. Sastoji se od 2 dijela. U prvom dijelu proučava se mogućnost uvođenja koncepta jezgrenih funkcija u metode za dubinsku analizu teksta, posebno metode za redukciju dimenzionalnosti. Opisani su neki od osnovnih algoritma za rad u jezgrenom području – algoritmi za redukciju dimenzionalnosti i klasifikaciju. Algoritmi za redukciju dimenzionalnosti temelje se na općenitom problemu svojstvenih vrijednosti i za njihovo razumijevanje potrebno je osnovno znanje linearne algebre te se mogu lako riješiti ili aproksimirati koristeći dobro poznate tehnike numeričke algebre. Nadalje ovakvi problemi mogu se riješiti u dualnoj reprezentaciji tj. potrebne su nam samo informacije o skalarnom produktu između točaka. Općeniti problemi svojstvenih vrijednosti obuhvaćaju pronalaženje *k* smjerova u preslikanom prostoru koji sadrže maksimalnu vrijednost varijance u podacima (metoda glavnih komponenata) ili pronalaze korelacije između različitih reprezentacija istih podataka (kanonska korelacijska analiza). Algoritme redukcije dimenzionalnosti slijedi klasifikacija podataka jednom od najpoznatijih metoda, metodom potpornih vektora.

U drugom dijelu ispitana su svojstva i mogućnosti primjene metrika različitih od Euklidske na postupke grupiranja i klasificiranja podataka u dubinskoj analizi teksta za model vektorskog prostora. U zadnjem dijelu kao redukcija dimenzionalnosti odabrana je metoda konceptnog indeksiranja te je na temelju nje i metode potpornih vektora izrađen klasifikator. Klasifikacija je testirana na testnim skupovima dokumenata na hrvatskom jeziku.

2 JEZGRENE METODE

2.1 Uvod u jezgrene metode

Metode raspoznavanja uzoraka primjenjuju se u mnogo područja gdje je potrebno klasificirati uzorke u određen broj klasa. Primjeri klasičnih problema raspoznavanja uzoraka su prepoznavanje slova, prepoznavanje govora, medicinska dijagnostika… Novije primjene raspoznavanja uzoraka su u području dubinske analize podataka te u dubinskoj analizi teksta (eng. text mining), tj. radu na sintezi novih informacija iz podataka koji dolaze isključivo u tekstualnom obliku.

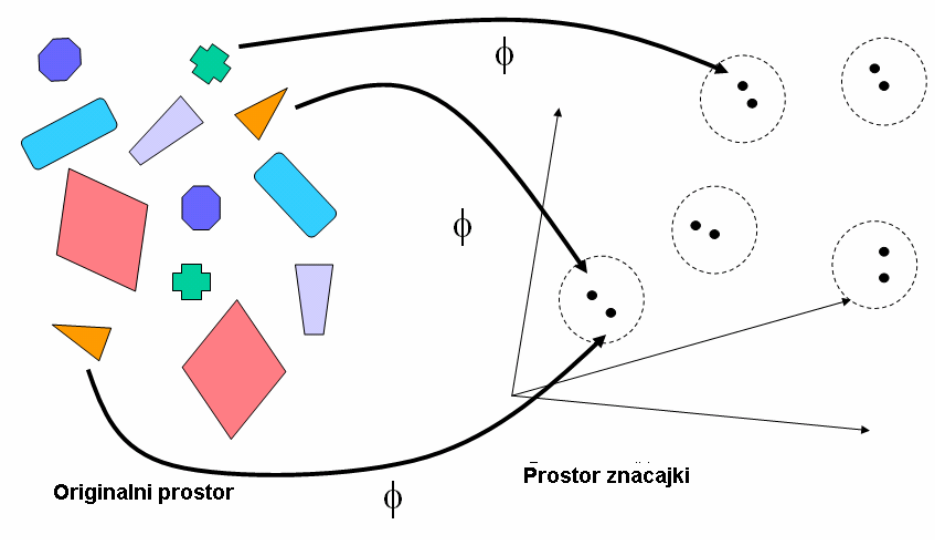
Evoluciju kroz koju su prolazili algoritmi raspoznavanja uzoraka možemo podijeliti u 3 etape. Sve je počelo 1960-ih otkrićem linearnih algoritama, među kojima je bitno spomenuti perceptron. Takvi algoritmi mogli su pronaći samo linearne relacije među uzorcima. Uslijedila je „nelinearna revolucija“ sredinom 1980-ih koja je dovela do razvoja raspoznavanja uzoraka i njegove primjene na do tada nezamislivim područjima. Iako je riješeno najvažnije pitanje postavljano do tada, korišteni su heuristički algoritmi i nedorađena statistička analiza, a algoritmi gradijentnog spusta i pohlepna heuristika doveli su do novih problema, među ostalim lokalnog minimuma i pretreniranosti. Treća etapa započinje sredinom 1990-ih uvođenjem novog pristupa poznatog kao jezgrene metode. Ovakav pristup omogućuje analizu nelinearnih relacija među uzorcima učinkovitošću prije rezerviranom samo za linearno odvojive uzorke, te nadalje, napredak u statističkoj analizi omogućio je analizu u višedimenzionalnom prostoru značajki. Prevladani su tipični problemi 3 etape, pretreniranost i lokalni minimum. Sa svih pogleda, računskog, statističkog i konceptualnog, ovi algoritmi predstavljaju novu stepenicu u razvoju raspoznavanja uzoraka.

Prvi put su uvedeni u metodi potpornih vektora, klasifikacijskom algoritmu koji je opravdao sve gore navedene prednosti jezgrenih metoda. Zatim su se proširili i na druga područja uz klasifikaciju uzoraka, ovdje možemo spomenuti još redukciju dimenzionalnosti i dvije metode kojima ćemo se baviti: jezgrena metoda glavnih komponenata i jezgrena kanonska korelacijska analiza.

Osnovni koncept jezgrenih metoda je preslikavanje uzoraka u novi pogodniji prostor u kojem se zatim koriste algoritmi temeljeni na linearnoj algebri, geometriji i statistici za pronalaženje relaciji među preslikanim uzorcima. Vidjet ćemo da je bit jezgrenih metoda

računski prečac za predstavljanje uzoraka u prostoru značajki koji zovemo jezgrena funkcija.

Zadani je skup S = {($\mathbf{x_1}$, $\mathbf{y_1}$), ($\mathbf{x_2}$, $\mathbf{y_2}$) ..., ($\mathbf{x_l}$, $\mathbf{y_l}$)} točaka $\mathbf{x_i} \subseteq \Re^N$ zajedno s odgovarajućim oznakama $\mathbf{y_i} \subseteq \Re$. U daljnjem tekstu ℓ će označavati broj zadanih točaka, dok je *N* njihova dimenzionalnost. Za taj skup definirano je preslikavanje $\phi : x \in \Re^N \rightarrow \phi(x) \in F$ na prostor značajki takvo da je



Slika 1. Preslikavanje točaka

rješavanje problema u novom prostoru jednostavnije npr. linearno. Ponekad nije potrebno eksplicitno znanje o preslikavanju ϕ, već je skalarni produkt u novom prostoru odabran kao potrebna mjera sličnosti između točaka. Jezgrena funkcija je ta mjera sličnosti i definirana je kao:

$$k(\mathbf{x},\mathbf{z}) = <\phi(x), \phi(z)>.$$

Moguće je, dok za sada izgleda nemoguće, koristiti prostor značajki sa eksponencijalnim ili beskonačnim brojem dimenzija. Krenimo sa jednostavnim primjerom jezgrene funkcije da shvatimo ključnu ideju ovakvog pristupa.

Za dvodimenzionalni ulazni prostor $X \subseteq \Re^2$ zadano je preslikavanje

$$\phi : x = (x_1, x_2) \rightarrow \phi(x) = (x_1^{\,2}, x_2^{\,2}, \sqrt{2}x_1x_2) \in F = \Re^3.$$

podaci se preslikavaju iz dvodimenzionalnog prostora u trodimenzionalni na taj način da linearne relacije u prostoru značajki odgovaraju kvadratnim relacijama na ulaznom prostoru. Skalarni produkt u prostoru značajki se može izračunati kao

$$\langle \phi(x), \phi(z) \rangle = \left\langle ({x_1}^2, {x_2}^2, \sqrt{2}x_1x_2), ({z_1}^2, {z_2}^2, \sqrt{2}z_1z_2) \right\rangle$$
$$= {x_1}^2{z_1}^2 + {x_2}^2{z_2}^2 + 2x_1x_2z_1z_2$$
$$= (x_1z_1 + x_2z_2)^2 = \langle x, z \rangle^2$$

Dakle $k(x,z) = \langle x, z \rangle^2$ je jezgrena funkcija. Vidimo da možemo izračunati skalarni produkt između projekcija bez eksplicitnog znanja njihovih koordinata. Međutim nisu sve funkcije jezgrene funkcije.

Gram matrica je definirana kao ℓ *x* ℓ matrica sa elementima $G_{ij} = \langle \phi(x), \phi(z) \rangle = k(x,z)$. Češće se ta matrica naziva jezgrenom matricom. Definirajmo još i matricu **X** dimenzija ℓ *x N*

$$\mathbf{X} = [\phi(\mathbf{x}_1),\ \phi(\mathbf{x}_2),\ldots,\ \phi(\mathbf{x}_l)]'.$$

Jezgrene matrice definirane su elementima koji su skalarni produkt točaka u prostoru značajki. Iz gornje definicije proizlazi da takve matrice moraju biti simetrične i pozitivne definitne tj. sve svojstvene vrijednosti matrice su veće od nule. Tako funkcija $k(x,z)$ zadovoljava svojstvo pozitivnosti i definitnosti ako je simetrična funkcija za koju i konstruirana jezgrena matrica zadovoljava ovo svojstvo. Prema Merceru jezgrene funkcije moraju zadovoljavati sljedeći uvjet:

$$\int_{XxX} k(x,z)f(x)f(z)dxdz \geq 0 \forall f \in L_2(X)$$

Napomenimo još jedanput da rad u jezgrenom području znači da ne možemo eksplicitno prikazati preslikane točke. Npr. slika ulazne točke **x** je $\phi(\mathbf{x})$, ali nemamo pristupa komponentama ovog vektora već samo vrijednosti skalarnog produkta između slika točaka. Unatoč ovakvoj smetnji začuđujuće je koliko se informacija može dobiti o $\phi(\mathbf{x})$ koristeći jezgrene funkcije.

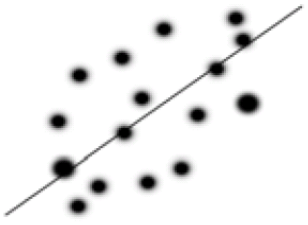
2.2 Redukcija dimenzionalnosti jezgrenim metodama

Sljedeća poglavlja analiziraju dvije metode redukcije dimenzionalnosti, tj. nastoje smisleno smanjiti dimenzionalnost *N* koje smo definirali u skupu S.

Analizirane redukcije dimenzionalnosti temelje se na općim problemima svojstvenih vektora. Proučavaju se jednostavnom linearnom algebrom, te je također izračunavanje ili njihova aproksimacija učinkovita. Nadalje, takvi problemi se mogu rješavati i u jezgrenom prostoru značajki koristeći dualnu reprezentaciju, tj. potrebni su nam samo podatci o skalarnom produktu između podatkovnih vektora.

2.2.1 Jezgrena metoda glavnih komponenata[1]

Za početak ćemo pretpostaviti da su podaci centrirani tj. $\sum_i x_i = 0$ [2]. Cilj nam je pronaći smisleno preslikavanje podataka i time smanjiti dimenzionalnost. Ipak, problem je postavljen tako da nemamo pristupa ikakvim oznakama $\mathbf{y_i}$. Da imamo, okrenuli bi se na PLS[3]. I bez oznaka, cilj nam je pronaći potprostor najvećih varijanci gdje broj novih dimenzija unaprijed odredimo. Varijanca je mjera rasipanja podataka, a računa se kao srednje kvadratno odstupanje od aritmetičke sredine. Može se dogoditi da je zanimljivi signal zarobljen u smjeru male varijance, i u tom slučaju metoda glavnih komponenata ili PCA nije dobra tehnika. Međutim, obično vrijedi da smjerovi male varijance pretstavljaju samo šum, te odšumljavanjem možemo samo popraviti reprezentaciju podataka.



[1] Eng. Kernel principal component analysis KPCA

[2] Centriranje je translacija osi.

[3] Eng. Partial Last Squares

Slika 2. Smjer najveće varijance podataka

Zapišimo kovarijacijsku matricu **C**

$$C_{st} = \frac{1}{l}\sum_{i}^{l} \phi(x_i)_s \phi(x_i)_t^T \text{ , } s,t = 1,\ldots,N$$

te razmotrimo izračunavanje varijance projekcije na normalizirani vektor **w** (napomenimo da su podaci centrirani)

$$\frac{1}{l}\sum_{i=1}^{l} (P_w(\phi(\mathbf{x}_i)))^2 = E[\mathbf{w}' \phi(\mathbf{x}) \phi(\mathbf{x})'\mathbf{w}] = \mathbf{w}' E[\phi(\mathbf{x}) \phi(\mathbf{x})'] \mathbf{w} = \mathbf{w'X'Xw} = \mathbf{w'Cw} ,$$

gdje se E koristi kao empirička srednja vrijednost ili $E[f(x)] = \frac{1}{l}\sum_{i=1}^{l} f(x)$.

Smjer koji maksimizira varijancu dobiva se rješavanjem sljedećeg problema

$$\max_w \ \mathbf{w'Cw} \qquad \text{uz} \qquad ||\mathbf{w}||_2 = 1.$$

Rješenje ovog problema dano je svojstvenim vektorom matrice **C** kojem je pridružena najveća svojstvena vrijednost. Svojstveni vektor koji odgovara najvećoj svojstvenoj vrijednosti je zapravo smjer u kojem su podaci najviše rastegnuti. Svojstvene vrijednosti ove matrice pretstavljaju količinu varijance u smjerovima pridruženih svojstvenih vektora. Sljedeći smjer je ortogonalan na njega, i odgovara vrijednosti najveće varijance u ortogonalnom potprostoru itd. Dakle, za smanjenje dimenzionalnosti projiciramo podatke na svojstvene - smjerove pridružene najvećim varijancama.

PCA uzima podskup *k* svojstvenih vektora kao osnovnih osi, dobivenih na skupu za učenje, te projicira podatke u prostor razapet tim osima. Nove koordinate nazivaju se glavne koordinate[4], a svojstveni vektori glavnim osima[5].

Projekcije su dane sa:

$$\mathbf{y}_i = \mathbf{U}_k^T \mathbf{x}_i \qquad \forall i$$

gdje je $\mathbf{U}_k$ *m x k* podmatrica sa prvih *k* svojstvenih vektora u stupcima, a algoritam:

---

[4] Eng. principal coordinates

[5] Eng. principal axes

Ulaz	podaci S = { $\mathbf{x}_1\ \mathbf{x}_2 \ldots \mathbf{x}_l$ } $\subset \Re^n$
	$\mathbf{C} = \frac{1}{l}\ \mathbf{X'X}$ *kovarijacijska matrica*
	[**U**,**Λ**] = eig(ℓ**C**) *dekompozicija*
	$\mathbf{x}_i^* = \mathbf{U}_k'\ \mathbf{x}_i$, *i = 1,…, ℓ preslikavanje*
Izlaz	S* = { $\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_l^*$}, *dimenzija k*

Algoritam metode glavnih komponenata

Kako je već napomenuto svojstvene vrijednosti pretstavljaju količinu varijance uhvaćene pridruženim svojstvenim vektorom. Čim je veća dimenzionalnost *k* veća je obuhvaćena varijanca. Nadalje, ortogonalna projekcija $P_{\mathbf{U}k}$ u potprostor razapet sa prvih *k* svojstvenih vektora matrice **C** je *k*-dimenzionalna ortogonalna projekcija koja minimalizira prosječnu kvadratnu udaljenosti između podatkovnih točaka i njihovih slika. Odnosno $\mathbf{U}_k$ rješava optimizacijski problem

$$\min \sum_{i=1}^{l} \| P_{\mathbf{U}}^{\perp}(\phi(\mathbf{x}_i)) \|^2_2.$$

Dokaz se može pronaći u (Shawe-Taylor et al., 2004.)

Posljedice Singular Value Decomposition.

Vidjet ćemo kako možemo naučiti nešto o kovarijacijskoj matrici **C** koristeći jezgrenu matricu **K** = **XX'**. Varijance koje su nam potrebne dane su u kovarijacijskoj matrici, ali se također mogu izračunati koristeći jezgrenu matricu. Njihova veza postat će očita ako razmotrimo njihove dekompozicije

$$\ell\mathbf{C} = \mathbf{X'X} = \mathbf{U}\ \mathbf{\Lambda}_k\ \mathbf{U'} \quad \text{i} \quad \mathbf{K} = \mathbf{XX'} = \mathbf{V}\ \mathbf{\Lambda}_k\ \mathbf{V'},$$

gdje su redci $\mathbf{u}_i$ ortonormalne matrice **U** svojstveni vektori ℓ**C**, a stupci $\mathbf{v}_i$ ortonormalne matrice **V** svojstveni vektori matrice **K**. Sad razmotrimo svojstveni vektor i pridruženu svojstvenu vrijednost **v**, λ matrice **K**. Dakle,

$$\ell\mathbf{C}(\mathbf{X}'\,\mathbf{v}) = \mathbf{X}'\mathbf{X}\mathbf{X}'\mathbf{v} = \mathbf{X}'\mathbf{K}\mathbf{v} = \lambda\mathbf{X}'\mathbf{v}$$

možemo zaključiti da je (**X'v**), λ svojstveni vektor i svojstvena vrijednost za ℓ**C**. Nadalje norma **X'v** je

$$\| \mathbf{X}'\mathbf{v} \|^2 = \mathbf{v}'\mathbf{X}\mathbf{X}'\mathbf{v} = \lambda$$

Iz gornjeg dobivamo normalizirani svojstveni vektor matrice ℓ**C**, $\mathbf{u} = \lambda^{-1/2}\,\mathbf{X}'\mathbf{v}$. Također postoji simetrija:

$$\lambda^{-1/2}\mathbf{X}\,\mathbf{u} = \lambda^{-1}\mathbf{X}\mathbf{X}'\mathbf{v} = \mathbf{v}$$

te možemo zapisati

$$\mathbf{u} = \lambda^{-1/2}\mathbf{X}'\mathbf{v} \qquad \text{i} \qquad \mathbf{v} = \lambda^{-1/2}\mathbf{X}\mathbf{u}$$

Dekompozicija matrice **X** slijedi iz sljedeće dvije jednakosti

$\mathbf{X}^*\mathbf{X}^{*\prime} = (\mathbf{X} - \mathbf{v}\mathbf{v}'\mathbf{X})\,(\mathbf{X} - \mathbf{v}\mathbf{v}'\mathbf{X})' = \mathbf{X}\mathbf{X}' - \lambda\mathbf{v}\mathbf{v}'$, i

$\mathbf{X}^{*\prime}\mathbf{X}^* = (\mathbf{X} - \mathbf{v}\mathbf{v}'\mathbf{X})'\,(\mathbf{X} - \mathbf{v}\mathbf{v}'\mathbf{X}) = \mathbf{X}'\mathbf{X} - \mathbf{X}'\mathbf{v}\mathbf{v}'\mathbf{X} = \mathbf{X}'\mathbf{X} - \lambda\mathbf{u}\mathbf{u}'$

te je $\mathbf{X}^* = \mathbf{X} - \mathbf{v}\mathbf{v}'\mathbf{X} = \mathbf{X} - \lambda^{-1/2}\mathbf{v}\mathbf{u}' = \mathbf{X} - \mathbf{X}\mathbf{u}\mathbf{u}'$.

Dakle, prvih $t = \text{rang}(\mathbf{X}\mathbf{X}') \le \min(N, \ell)$ stupaca matrice $\mathbf{U}_t$ može se odabrati kao

$$\mathbf{U}_t = \mathbf{X}'\mathbf{V}_t\mathbf{\Lambda}_t^{-1/2}.$$

Proširenjem $\mathbf{U}_t$ na **U** i matrice $\mathbf{\Lambda}_t^{-1/2}$ $N \times \ell$ matricu čiji su dodatni elementi svi nula, dobili smo singular value decomposition(SVD) matrice **X'**

$$\mathbf{X}' = \mathbf{U}\Sigma\mathbf{V}'$$

gdje je **Σ** $N \times \ell$ matrica sa svim elementima nula, osim vodećih na dijagonali sa vrijednostima $\sigma_i = \lambda_i^{-1/2}$ koje zadovoljavaju $\sigma_1 \ge \sigma_2 \ge \ldots \ge \sigma_t > 0$ za $t = \text{rang}(\mathbf{X}) \le \min(N, \ell)$, te **U** i **V** kvadratnim ortogonalnim matricama.

Prethodna definicija matrice $\mathbf{U}_t$ implicira dualni reprezentaciju svojstvenog vektora $\mathbf{u}_j$ matrice ℓ**C** prikazanog pomoću odgovarajućeg svojstvenog vektora $\mathbf{v}_j$ matrice **K** pomnoženog sa $\lambda_j^{-1/2}$ tj.

$$\mathbf{u}_j = \lambda_j^{-\frac{1}{2}} \sum_{i=1}^{l} (v_j)_i \phi(x_i) = \sum_{i=1}^{l} \alpha_i^j \phi(x_i), \qquad j = 1, \ldots, t,$$

gdje su dualne varijable α_j za *j*-ti vektor $\mathbf{u}_j$ dane sa $\alpha_j = \lambda_j^{-\frac{1}{2}} v_j$. Također možemo izračunati projekciju točke $\phi(\mathbf{x})$ na smjer $\mathbf{u}_j$ u prostoru značajki

$P_{uj}(\phi(\mathbf{x})) = \mathbf{u}_j' \phi(\mathbf{x}) = \left\langle \sum_{i=1}^{l} \alpha_i^j \phi(\mathbf{x}_i), \phi(\mathbf{x}) \right\rangle = \sum_{i=1}^{l} \alpha_i^j \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle = \sum_{i=1}^{l} \alpha_i^j k(x_i, x)$.

Jezgrena metoda glavnih komponenata

Kernel PCA je primjena PCA algoritma u jezgrenom prostoru značajki te se tako iskorištava dualna reprezentacija. Iz prethodne jednadžbe znamo da je projekcija

$$P_{Uk}(\phi(\mathbf{x})) = (\mathbf{u}_j' \phi(\mathbf{x}))_{j=1}^{k} = \left(\sum_{i=1}^{l} \alpha_i^j k(x_i, x) \right)_{j=1}^{k} \quad \text{sa} \quad \alpha_j = \lambda_j^{-\frac{1}{2}} v_j .$$

Koristeći jezgrenu matricu **K** te njenu dekompoziciju na $\mathbf{V}\ \mathbf{\Lambda}_k\ \mathbf{V}'$ jednostavno je prilagoditi PCA algoritam za prostor značajki (Shawe-Taylor et al., 2004.)

Ulaz *podaci* $S = \{ \mathbf{x}_1\ \mathbf{x}_2 \ldots \mathbf{x}_l \} \subset \Re^n$

$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ *jezgrena matrica*

// centriranje jezgrene matrice, zamjenjuje centriranje podataka

$[\mathbf{V}, \mathbf{\Lambda}] = \text{eig}(\mathbf{K})$ *dekompozicija jezgrene matrice*

$\alpha_j = \lambda_i^{-\frac{1}{2}} v_j$, $j = 1,\ldots,k$

$\mathbf{x}_i^* = \left(\sum_{i=1}^{l} \alpha_i^j k(x_i, x) \right)_{j=1}^{k}$ *preslikavanje*

Izlaz $S^* = \{ \mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_l^* \}$, *dimenzija k*

Algoritam jezgrene metode glavnih komponenata

2.2.2 Jezgrena kanonska korelacijska analiza[6]

Pretpostavimo da imamo 2 kopije korpusa, jednu napisanu na engleskom, a drugu na hrvatskom jeziku. Ovakav paralelni korpus nazivamo još i upareni set dokumenata. Zainteresirani smo za manje dimenzionalni prikaz dokumenata. Da imamo jedan jezik, mogli bi pomoću PCA dobiti dimenzije koje imaju najveće varijance. Time bi mogli riješiti problem sinonimije, jer ako se riječi često pojavljuju u dokumentima tj. vrlo su korelirane, nastoje se spojiti u istu dimenziju u novom prostoru. Za korpus sa dva prijevoda, možemo tražiti projekciju svakog prijevoda posebno tako da su projekcije maksimalno korelirane(Shawe-Taylor et al., 2004.).

Za upareni skupa podataka[7], gdje je isti objekt **x** prikazan na dva načina kao $\phi_a(\mathbf{x})$ i $\phi_b(\mathbf{x})$, $S = \{(\phi_a(\mathbf{x}_1), \phi_b(\mathbf{x}_1)), \ldots, (\phi_a(\mathbf{x}_l), \phi_b(\mathbf{x}_l))\}$ nastojimo maksimizirati empiričku korelaciju između $\mathbf{x}_a = \mathbf{w}_a'\phi_a(\mathbf{x})$ i $\mathbf{x}_b = \mathbf{w}_b'\phi_b(\mathbf{x})$ projiciranih na $\mathbf{w_a}$ i $\mathbf{w_b}$.

$$\text{Corr}(\mathbf{x}_a, \mathbf{x}_b) = \frac{\mathrm{E}[x_a, x_b]}{\sqrt{\mathrm{E}[x_a, x_a]\mathrm{E}[x_b, x_b]}} = \frac{\mathrm{cov}(x_a, x_b)}{\sqrt{\mathrm{var}(x_a, x_a)\,\mathrm{var}(x_b, x_b)}}$$

$$= \frac{\mathrm{E}[w_a'\phi(x_a)\phi(x_b)'w_b]}{\sqrt{\mathrm{E}[w_a'\phi(x_a)\phi(x_a)'w_a]\mathrm{E}[w_b'\phi(x_b)\phi(x_b)'w_b]}} = \frac{w_a'C_{ab}w_b}{\sqrt{w_a'C_{aa}w_a w_b'C_{bb}w_b}}$$

gdje smo na sljedeći način rastavili kovarijacijsku matricu

$$\mathbf{C} = \frac{1}{l}\sum_{i=1}^{l} (\phi_a(\mathbf{x}_i), \phi_b(\mathbf{x}_i))\,(\phi_a(\mathbf{x}_i), \phi_b(\mathbf{x}_i))' = \begin{pmatrix} \frac{1}{l}\sum_{i=1}^{l}\phi_a(x)\phi_a(x)' & \frac{1}{l}\sum_{i=1}^{l}\phi_b(x)\phi_a(x)' \\ \frac{1}{l}\sum_{i=1}^{l}\phi_a(x)\phi_b(x)' & \frac{1}{l}\sum_{i=1}^{l}\phi_b(x)\phi_b(x)' \end{pmatrix}$$

$$= \begin{pmatrix} C_{aa} & C_{ba} \\ C_{ab} & C_{bb} \end{pmatrix}$$

Ovaj optimizacijski problem sličan je optimizacijskom problemu PLS-a[8]. Vektori $\mathbf{w}_a$ i $\mathbf{w}_b$ određeni su opet samo svojim smjerovima jer njihovo množenje sa λ_a te λ_b daje

---

[6] Eng. Kernel Canonical correlation analysis KCCA

[7] Eng. pared or aligned dataset

$$\frac{\lambda_a \lambda_b w_a' C_{ab} w_b}{\sqrt{\lambda_a{}^2 w_a' C_{aa} w_a \lambda_b{}^2 w_b' C_{bb} w_b}} = \frac{\lambda_a \lambda_b w_a' C_{ab} w_b}{\lambda_a \lambda_b \sqrt{w_a' C_{aa} w_a w_b' C_{bb} w_b}}$$

$$= \frac{w_a' C_{ab} w_b}{\sqrt{w_a' C_{aa} w_a w_b' C_{bb} w_b}}$$

te time možemo ograničiti doljnja dva izraza u nazivniku na vrijednost 1. Postavljamo sljedeći optimizacijski problem

$$\max_{w_a, w_b} \quad \mathbf{w_a' C_{ab} w_b}$$

$$\text{uz uvjet} \quad \mathbf{w_a' C_{aa} w_a} = 1 \text{ i } \mathbf{w_b' C_{bb} w_b} = 1$$

Ako primijenimo Lagrangeov multiplikator dobivamo

$$\max \quad \mathbf{w_a' C_{ab} w_b} - \frac{\lambda_a}{2} (\mathbf{w_a' C_{aa} w_a} - 1) - \frac{\lambda_b}{2} (\mathbf{w_b' C_{bb} w_b} - 1)$$

Derivirajući po $\mathbf{w_a}$ i $\mathbf{w_b}$

$$\mathbf{C_{ab} w_b} - \lambda_a \mathbf{C_{aa} w_a} = 0 \qquad \text{i} \qquad \mathbf{C_{ba} w_a} - \lambda_b \mathbf{C_{bb} w_b} = 0$$

oduzimajući prvu jednadžbu pomnoženu sa $\mathbf{w_a'}$ te drugu pomnoženu sa $\mathbf{w_b'}$ dobivamo

$$\lambda_a \ \mathbf{w_a' C_{aa} w_a} - \lambda_b \ \mathbf{w_b' C_{bb} w_b} = 0$$

Zaključujemo, ako uzmemo dva prethodna uvjeta da je $\lambda_a = \lambda_b$ te će se u daljnjem tekstu ova vrijednost obilježavati sa λ .

Problem možemo postaviti kao općeniti problem svojstvenih vrijednosti

$$\begin{pmatrix} 0 & C_{ab} \\ C_{ba} & 0 \end{pmatrix} \begin{pmatrix} w_a \\ w_b \end{pmatrix} = \lambda \begin{pmatrix} C_{aa} & 0 \\ 0 & C_{bb} \end{pmatrix} \begin{pmatrix} w_a \\ w_b \end{pmatrix}$$

Jezgrena kanonska korelacijska analiza

Naravno, problem želimo riješiti u dualnoj formulaciji. Nastavit ćemo sa predstavljanjem $\mathbf{w_a}$ i $\mathbf{w_b}$ pomoću matrice $\mathbf{X_a}$ čiji su redci vektori $\phi_a(\mathbf{x}_i)$, $i = 1, \ldots, \ell$, te matrice $\mathbf{X_b}$ sa redcima $\phi_b(\mathbf{x}_i)$

[8] Eng. Partial last squares

$\mathbf{w_a} = \mathbf{X_a}'\alpha_a$ i $\mathbf{w_b} = \mathbf{X_b}'\alpha_b$

te optimizacijski problem postaje

max $\alpha_a' \mathbf{X_a X_a}' \mathbf{X_b X_b}' \alpha_b$

uz uvjet $\alpha_a' \mathbf{X_a X_a}' \mathbf{X_a X_a}' \alpha_a = 1$ te $\alpha_b' \mathbf{X_b X_b}' \mathbf{X_b X_b}' \alpha_b = 1$

ili zapisano pomoću jezgrenih matrica

max $\alpha_a' \mathbf{K_a K_b} \alpha_b$

uz uvjet $\alpha_a' \mathbf{K_a}^2 \alpha_a = 1$ te $\alpha_b' \mathbf{K_b}^2 \alpha_b = 1$

Pomakom u jezgreno definirani prostor značajki, dodatna fleksibilnost uvodi i opasnost od pretreniranosti. Time mislimo na pronalaženje lažnih korelacija koristeći vektore težine tako da su dvije projekcije potpuno poravnane. Na primjer, ako su podatci linearno nezavisni u oba prostora značajki možemo pronaći linearnu transformaciju koja preslikava ortogonalno svaki ulazni podatak u prostoru značajki. Sad je moguće pronaći i savršene korelacije između odabrane projekcije u jednom prostoru te proizvoljne projekcije u drugom. Maksimiziranje korelacije odgovara minimiziranju funkcije $\| \mathbf{w}_a'\phi_a(\mathbf{x}) - \mathbf{w}_b'\phi_b(\mathbf{x}) \|^2$. Korištenje jezgrene funkcije redovito dovodi do linearne nezavisnosti skupa za učenje te je potrebno uvesti kontrolu fleksibilnosti preslikavanja u prostore značajki. Regularizacijski parametri π_a i π_b glatko interpoliraju između maksimiziranja korelacije i maksimiziranja kovarijance. Koristeći Lagrangeovu metodu dolazimo do

$\mathbf{K_a K_b} \alpha_b - \lambda (1 - \pi_a) \mathbf{K_a}^2 \alpha_a - \lambda \pi_a \mathbf{K_a} \alpha_a = 0$

$\mathbf{K_b K_a} \alpha_a - \lambda (1 - \pi_b) \mathbf{K_a}^2 \alpha_b - \lambda \pi_b \mathbf{K_b} \alpha_b = 0$

Dakle formirali smo općeniti problem svojstvenih vrijednosti

$$\begin{pmatrix} 0 & K_a K_b \\ K_b K_a & 0 \end{pmatrix} \begin{pmatrix} \alpha_a \\ \alpha_b \end{pmatrix} = \lambda \begin{pmatrix} ((1-\pi_a)K_a^2 + \pi_a K_a & 0 \\ 0 & ((1-\pi_b)K_b^2 + \pi_b K_b \end{pmatrix} \begin{pmatrix} \alpha_a \\ \alpha_b \end{pmatrix}$$

Standardan pristup rješavanju ovog problema je nepotpuna Cholesky dekompozicija na srednjoj matrici sa desne strane.

Regularizacijski parametar osim računskog problema određuje i funkcijski prostor u kojem se traži rješenje. Sa većim vrijednostima regulacijskog parametra metoda je manje osjetljiva na ulazne podatke time i stabilnija (manje je vjerojatno da će pronaći lažne relacije među podacima). Sposobnost metode da uhvati koristan signal mjeri se

uspoređivanjem na stvarnim ulaznim i slučajnim podacima. Slučajni podaci dobivaju se slučajnim udruživanje parova podataka, na primjer (Hr, rand(E)) predstavlja hrvatsko-engleski korpus koji dobivamo premetanjem engleskih rečenica na stvarnom korpusu. Označimo dobivene korelacije sa $KCCA_{\pi}(Hr,E)$ sa regularizacijskim parametrom π, te **J** matricu sa svim jedinicama. Ako dolazi do pretreniranosti moguće je pronaći savršene korelacije te će vrijediti $|| \mathbf{J} - KCCA_{\pi}(Hr,E)|| \approx 0$. Za male vrijednosti regularizacijskog parametra za bilo koje povezivanje dokumenata vrijednost funkcije mjere bit će blizu 0 te zaključujemo da to indicira pretreniranost. Kako se λ povećava funkcija mjere slučajno povezanih podataka penje se daleko od 0, dok ona pravilno uparenih dokumenata ostaje korelirana. Pravilni parametar π odabiremo tako da je $|| \mathbf{J} - KCCA_{\pi}(Hr,rand(E))|| > || \mathbf{J} - KCCA_{\pi}(Hr,E)|| > 0$ (Vinokourov et al., 2002.).

2.3 Jezgrene funkcije za klasifikaciju teksta

Najjednostavniji prikaz teksta u modelu vektorskog prostora[9] je prikaz dokumenata kao „torbe riječi“[10]. Torba je skup u kojem su ponavljanja dopuštena, tako da se u obzir uzima pojava riječi i frekvencija pojavljivanja. Dokument je predstavljen riječima koje sadrži, ali se njihov redoslijed i konstrukcija rečenica ne uzimaju u obzir što dovodi do gubitka informacija. Riječ je bilo koji niz slova odvojen razmacima ili znakovima interpunkcije. Korišteni „izraz“ je sinonim za riječ. Torba, ili dokument, može se prikazati kao vektor izraza

$$\phi(d) = (\text{tf}(t_1,d), \text{tf}(t_2,d), \ldots, \text{tf}(t_N,d)) \qquad \in \Re^N,$$

u kojem svaki element tf[11](t_i,d) označava broj pojavljivanja izraza t_i u dokumentu d. Obično $\phi(d)$ ima tisuće elemenata i često je taj broj veći od broja dokumenata u skupu za učenje. Za pojedini dokument ovakva reprezentacija je jako rijetka, ima samo nekolicinu nenul elementa, te se može koristiti i drukčiji, sažetiji zapisi.

Ako sa **D** označimo dokument-izraz matricu čiji su redci dokument-vektori

$$\mathbf{D} = [\phi(d_1),\ \phi(d_2)\ ,\ \ldots\ ,\ \phi(d_l)]'$$

i analogna je prije korištenoj matrici **X**, onda **D'** predstavlja izraz-dokument matricu. Postoji direktna veza između matrica **X** i **D** gdje značajke postaju izrazi, a točke dokumenti. Također možemo definirati jezgrenu matricu

$$\mathbf{K} = \mathbf{DD'}$$

koja odgovara jezgri vektorskog prostora[12]

$$k(d_1,d_2) = \langle \phi(d_1),\ \phi(d_2) \rangle = \sum_{j=1}^{N} \text{tf}(t_j,d_1), \text{tf}(t_j,d_2)$$

Također je moguće koristiti nelinearna preslikavanja koristeći standardne jezgrene funkcije, npr. polinomialna jezgrena funkcija na normaliziranim podacima

---

[9] Eng. Vector space model VSM

[10] Eng. Bag-of-words

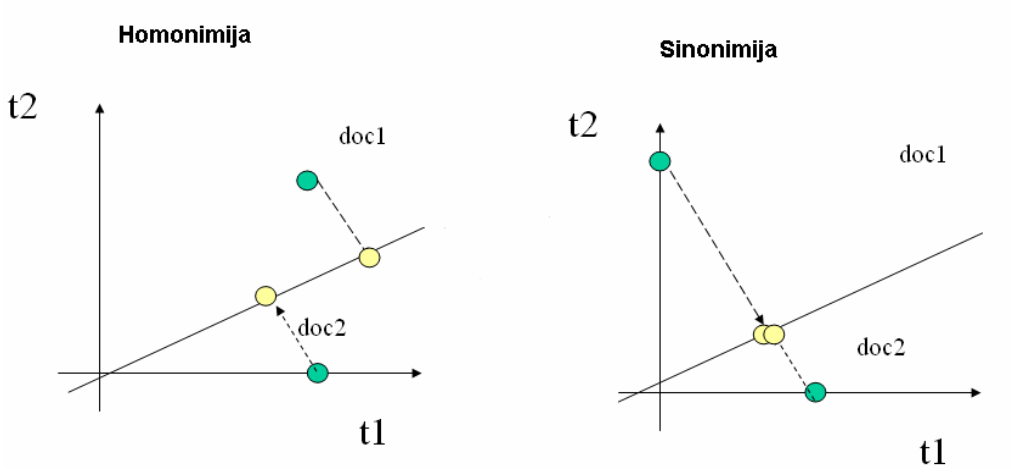
[11] Eng. Term frequency ili frekvencija pojavljivanja riječi u dokumentu

[12] Eng. Vector space kernel

$$k^*(d_1,d_2) = (k(d_1,d_2) + 1)^d = (< \phi(d_1), \phi(d_2) > +1)^d ,$$

stupnja *d* koristi do *n* višerječnih izraza[13] $0 \leq n \leq d$ za značajke. Iako se čini složenijom računski ne predstavlja veći problem od linearne. Također se može koristiti i bilo koja druga jezgrena funkcija.

Prikaz dokumenata u modelu vektorskog prostora ne uzima u obzir semantičke veze među izrazima, već izraze promatra kao izolirane pojave. Stoga su dokumenti koji govore o istoj temi, ali različitim izrazima preslikani u udaljena područja. Problemi sinonimije(različite riječi isto značenje) i homonimije(iste riječi različito značenje) mogu se riješiti dobro definiranim preslikavanjem koje udaljava dokumente koji sadrže homonime, a zbližava dokumente koji sadrže sinonime.



Slika 3. Homonimija i sinonimija

Neka rješenja semantičkih problema proizlaze iz same statistike riječi u dokumentu. Prvo je pridjeljivanje težina izrazima ili pridruženim koordinatama. Tu spada i uklanjanje stop riječi, riječi koje ne nose nikakvu informaciju o kategoriji te ih karakterizira velika frekvencija pojavljivanja. Definira se eksplicitna lista riječi koji će biti uklonjeni. Većinom su to prijedlozi, veznici i prilozi. Primjeri za hrvatski jezik su: *a, ah, ako, barem, bilo, četiri, ću, dokud, doskora, gdje, hoho, hopla, itekako, jesam, jutros, koga, manje, naprosto, nije, tebe, uzastopce, xxiii, zum,…* Drugi problem je utjecaj dužine dokumenta. Što je duži dokument više riječi sadrži te norma dokument-vektora veća. Jezgrena funkcija koja odbacuje utjecaj norme dana je:

[13] Eng. Monomials, multiword terms

$$\hat{k}(d_1, d_2) = \left\langle \frac{\phi(d_1)}{\| \phi(d_1) \|}, \frac{\phi(d_2)}{\| \phi(d_2) \|} \right\rangle = \frac{k(d_1, d_2)}{\sqrt{k(d_1, d_1) k(d_2, d_2)}}$$

Cilj nam je pronaći jezgrenu semantičku funkciju koja uzima u obzir semantičke veze među riječima. Proučimo najjednostavnije linearno preslikavanje podataka $\tilde{\phi}$ (d) = ϕ (d)**S** , gdje je **S** matrica koja može biti kvadratna, dijagonalna, ili općenito *N x k* matrica. Jezgrena funkcija dobiva zatim novu formu

$$\tilde{k} \text{ (d}_1\text{,d}_2\text{) = } \phi \text{ (d}_1\text{)}\mathbf{SS'}\phi \text{ (d}_2\text{)' = } \tilde{\phi} \text{ (d}_1\text{) } \tilde{\phi} \text{ (d}_2\text{) '.}$$

Matrica **S** zove se semantička matrica. Različiti izbori matrice vode k različitim varijantama modela vektorskog prostora[14]. Matrica **S** može se definirati kao

$$\mathbf{S = RP}$$

gdje je **R** dijagonalna matrica koja važe težine pojedinih riječi, i **P** matrica bliskosti[15] koja određuje semantičku blizinu različitih izraza.

Mjera *idf* važe težine pojedinih riječi kao funkciju inverzne frekvencije dokumenata. Ako pretpostavimo da imamo *l* dokumenata, i *df*[16]*(t)* označava broj dokumenata koji sadrže izraz *t*, onda je mjera inverzne frekvencije dokumenata za izraz *t* dana sa

$$\text{w(t) = } \ln\left(\frac{l}{df(t)}\right)$$

Dakle, smanjuje se težine riječi koje se često pojavljuju te dolaze do izražaja diskriminantno bitne riječi. Za stop riječi, ako pretpostavimo da se pojavljuju u skoro svim dokumentima vrijedi df(t) = *l* te je w(t) = 0. Iako se na ovaj način implicitno možemo riješiti stop riječi ipak je poželjnije radi učinkovitosti definirati eksplicitnu listu izraza koji će biti uklonjeni. Matrica **R** je dijagonalna sa elementima R_{tt} = *w(t).* Jezgrena funkcija ϕ (d$_1$)**RR'**ϕ (d$_2$)' koja koristi frekvenciju izraza kao i inverznu frekvenciju dokumenata često se naziva *tf-idf* reprezentacija.

---

[14] Eng. Vector space model

[15] Eng. Proximity matrix

[16] Eng. Document frequency

Međutim *tf-idf* reprezentacija i dalje ne može prepoznati dva semantički povezana izraza. Semantička bliskost riječi uvodi se pomoću matrice **P** od koje se zahtjeva da ima nedijagonalne elemente $P_{ij} > 0$ ako su izrazi *i* semantički povezan sa izrazom *j*. Ovakva jezgrena funkcija $\phi(d_1)\mathbf{PP}'\phi(d_2)'$, odgovara reprezentaciji gušćih dokument-vektora $\phi(d)\mathbf{P}$ koji ima nenul elemente za sve izraze semantički bliske izrazima koji se nalaze u dotičnom dokumentu.

Općeniti model vektorskog prostora[17]

Ova metoda nastoji uhvatiti izraz-izraz korelacije koristeći znanje o njihovom zajedničkom pojavljivanju u dokumentu. Dva izraza se smatraju semantički povezana ako se često pojavljuju u istim dokumentima. Prema ovome, dva dokumenta će biti slična i ako ne dijele zajedničke izraze nego izrazi koje sadrže često se zajedno pojavljuju u drugim dokumentima.

Dokument je predstavljen preslikavanjem:

$$\tilde{\phi}(d) = \phi(d)\mathbf{D}',$$

gdje je **D** dokument – riječ matrica, što je jednako **P** = **D'**. Ovakvo preslikavanje nije odmah jasno, ali ako raspišemo jezgrenu funkciju

$$\tilde{k}(d_1,d_2) = \phi(d_1)\mathbf{D}'\mathbf{D}\phi(d_2)'$$

možemo primijetiti da matrica **D'D** ima *(i, j)* ne nula element samo ako postoji dokument koji sadrži izraz *i* i izraz *j*,

$$(\mathbf{D}'\mathbf{D})_{ij} = \sum_d \mathrm{tf}(i,d)\,\mathrm{tf}(j,d) \ .$$

Ako je manje dokumenata od izraza ovakav postupak smanjuje dimenzionalnost preslikavanjem iz prostora veličine broja izraza u prostor veličine broja dokumenata. Iako zanimljiv, ovakav postupak je naivan u korištenju informacije zajedničkoj pojavljivanja.

[17] GVSM eng. Generalised vector space model

Latentno semantičke jezgrene funkcije[18]

Latentno semantičko indeksiranje(LSI) slijedi isti princip kao GVSM, ali se razlikuju u tehnikama koje se koriste pri dobivanju informacija o zajedničkom pojavljivanju izraza. LSI koristi singular value decomposition (SVD).

Rastava matrice **D'** je

$$\mathbf{D' = U\Sigma V}$$

gdje je **Σ** dijagonalna matrica, dok su **U** i **V** matrice čiji su stupci svojstveni vektori redom matrica **D'D** i **DD'**. LSI preslikava dokumente u prostor razapet sa prvih *k* stupaca matrice **U**

$$\mathbf{D} \rightarrow \phi(d)U_k .$$

Svojstvene vektore za skup dokumenta možemo zamisliti kao koncepte opisane linearnom kombinacijom izraza koji su odabrani na takav način da se dokumenti opisuju na najbolji način prikazani samo sa *k* takvih koncepata. Izrazi koji se često pojavljuju zajedno nastojat će se pojaviti u istom svojstvenom vektoru jer SVD ujedinjuje vrlo korelirane dimenzije da bi se dobio manji broj dimenzija sposobnih za rekonstrukciju cijelog vektora. Dakle, SVD koristi informacije o zajedničkom pojavljivanju izraza u sofisticiranijem algoritmu.

Prema definiciji LSI-a vidimo da je jednak PCA algoritmu u prostoru značajki. Nova jezgrena funkcija postaje ona PCA-a algoritma

$$\tilde{k}\,(d_1,d_2) = \phi\,(d_1)\mathbf{U_k U_k'}\,\phi\,(d_2)'$$

dok je matrica **P** u ovom slučaju odabrana kao $\mathbf{U_k}$. Ako usporedimo sa GVSM te zapišemo njegovu jezgrenu funkciju u obliku:

$$\tilde{k}_{GVSM}\,(d_1,d_2) = \phi\,(d_1)\mathbf{D'D}\,\phi\,(d_2)' = \phi\,(d_1)\mathbf{U'\Sigma U}\,\phi\,(d_2)'$$

možemo vidjeti dvije prilagodbe. Prvo, provedena je redukcija dimenzionalnosti korištenjem samo *k* svojstvenih vektora te drugo, izbacivanjem matrice **Σ** projekcije podataka postaju ortonormalne.

Činjenica identičnosti LSI i PCA algoritma vodi do zaključka da se može također implementirati u dualnoj reprezentaciji ili

[18] Eng. Latent semantic kernels

$$\phi(\mathrm{d})\mathbf{U_k} = \left(\lambda_j^{-\frac{1}{2}} \sum_{i=1}^{l} (v_j)_i k(d_i, d) \right)_{j=1}^{k},$$

gdje je *k* osnovna jezgrena funkcija koja može biti obična linearna jezgrena funkcija, ili složena jezgrena funkcija sa vaganjem težina izraza ili polinomialna jezgrena funkcija. Ovakva dualna reprezentacija naziva se latentno semantička jezgrena funkcija(LSK[19]) (Shawe-Taylor et al., 2004.).

Ako prikažemo matricu bliskosti

$$\mathbf{P} = \mathbf{U_k}\mathbf{U_k}'$$

vidimo da je ona kvadratna i određuje veze između različitih izraza. Kako se *k* povećava matrica **P** teži jediničnoj matrici i jedinice samo na dijagonali pokazuje da su sve riječi semantički različite. Dakle, vrijednost *k* kontrolira količinu semantički bitnih informacija koje se uvode u reprezentaciju.

[19] LSK eng. Latent semantic kernels

2.4 Istraživanje višejezičnih korpusa

Upareni korpusi[20] sastoje se od parova dokumenata koji su međusobni prijevodi na dva jezika. Oni su primjeri uparenih skupova za učenje korištenih pri CCA algoritmu. Prema prijevodu možemo se odnositi na dva načina: prvo, možemo ga smatrati kompleksnom oznakom za prvu verziju dokumenta, ili drugo: obje verzije možemo gledati kao dva pogleda na isti objekt.

Upareni korpus možemo koristiti pri učenju semantičkog preslikavanja koji će se koristiti samo na jednom prijevodu. Prijevodi se koristi kao složene oznake koje pronalaze semantičko preslikavanje potrebno za preslikavanje dokumenata u novi prostor bez semantički nebitni dijelova reprezentacije.

Latentno semantičke jezgrene funkcije mogu se također koristiti s ovakvim korpusima. Potrebno je spojiti dvije verzije dokumenata u jedan, višejezični tekst. Koristi se zatim PCA redukcija dimenzionalnost u k semantičkih dimenzija. Novi dokument bez prijevoda preslikava se u prostor svojstvenih vektora dobivenih spojenim dokumentima.

$$\mathbf{D} = \begin{pmatrix} D_x \\ D_y \end{pmatrix} = \mathbf{U\Sigma V}^{\mathsf{T}}$$

Latentno semantičke jezgrene funkcije imaju taj nedostatak da su ograničene spojem dvojezičnih dokumenata. Novi dokument bez prijevoda preslikava se u prostor svojstvenih vektora dobivenih spojem dokumenata tako da se i on proširi (dodaju se nule kao komponente drugog jezika). Zatim se novi dokument preslikava u prostor razapet sa k prvih svojstvenih vektora $\mathbf{U_k}$: $\mathbf{q^* = U_k^T q}$.

Alternativna metoda je korištenje CCA. Ova metoda tretira dvije verzije dokumenta kao dva pogleda na isti semantički objekt te traži dvije različite projekcije u zajednički semantički prostor. Najprije se odabere k semantičkih dimenzija, $0 \leq k \leq \ell$, sa najvećom vrijednosti korelacije i zatim se novi dokument, ili upit, preslikaju u novi prostor $\mathbf{q^* = A^T Z^T q}$,

[20] Eng. Pared or Aligned corpora

gdje je **A** ℓ *x k* matrica čiji su stupi rješenja KCCA algoritma za jedan od jezika tj. jezik koji se koristi u novom dokumentu. **Z** je skup za učenje.

Također se KCCA može koristiti i u kategorizaciji teksta. Semantički vektori u jednom od jezika **W** = **ZA** mogu se prenijeti i u druge aplikacije, npr. metodu potpornih vektora. Nova jezgrena funkcija postaje

$$\tilde{k}\,(d_1,d_2) = \phi\,(d_1)\mathbf{WW'}\,\phi\,(d_2)'$$

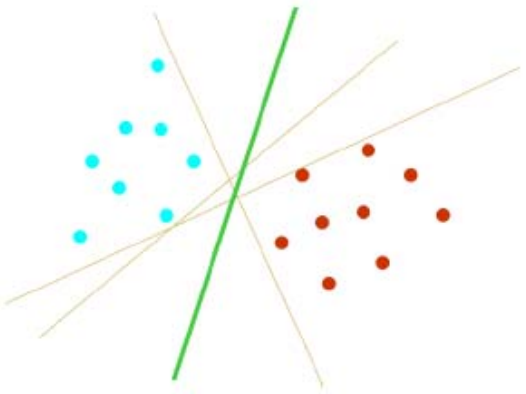
Matrica **WW'** može se jedanput izračunati i spremiti za sljedeću upotrebu pri klasifikaciji podataka metodom potpornih vektora (Vinokourov et al., 2002.).

2.5 Metoda potpornih vektora

Nakon redukcija dimenzionalnosti zadatak nam postaje klasifikacija, tj. potrebno je odrediti kojem od dva razreda pripada testni uzorak. Skup za učenje možemo zapisati u obliku $\{ \mathbf{x}_i, \mathbf{y}_i \}$, $i = 1,\ldots,n$ i $\mathbf{x}_i \in \Re^d$, $y_i \in \{-1, +1\}$, gdje su $\mathbf{x}_i$ ulazni vektori, a $\mathbf{y}_i$ varijable odziva ili oznake.

Razmotrimo jednostavan primjer u kojem su podaci linearno razdvojivi tj. možemo nacrtati ravnu liniju $f(\mathbf{x}) = \mathbf{w}\mathbf{x} - b$ tako da se svi slučajevi sa $y_i = -1$ nalaze s jedne strane i za njih vrijedi $f(\mathbf{x}) < 0$, a slučajevi sa $y_i = +1$ nalaze se s druge strane i imaju vrijednost $f(\mathbf{x}) > 0$. Prema tome testne vektore možemo klasificirati prema njihovom predznaku funkcije.

Kako postoji beskonačno mnogo takvih hiperravnina koje rješavaju naš problem klasifikacije na skupu za učenje, postavlja se pitanje kako odabrati najbolju s obzirom na to da se sve razlikuju svojom klasifikacijom samo na testnim primjerima. Na primjer, možemo povući liniju vrlo blizu članova jednog razreda, recimo $y = -1$. Na dobivenim testnim primjerima nećemo imate velike pogreške sa slučajevima koji trebaju biti klasificirani kao $y = +1$, ali vrlo je velika vjerojatnost da ćemo pogriješiti sa slučajevima $y = -1$. Pametno bi bilo postaviti liniju čim dalje od $y = -1$ i $y = -1$ slučajeva iz skupa za učenje, tj. točno u njihovoj sredini.



Slika 4. Optimalna hiperravnina

Geometrijski, vektor $\mathbf{w}$ usmjeren je ortogonalno na liniju definiranu sa $\mathbf{w}^T\mathbf{x} = b$. To se može protumačiti na sljedeći način. Za početak uzmimo $b = 0$. Sada je jasno da svi vektori, $\mathbf{x}$, kojima skalarni produkt sa $\mathbf{w}$ nestaje zadovoljavaju ovu jednadžbu tj. svi vektori ortogonalni na $\mathbf{w}$ zadovoljavaju ovu jednadžbu. Sada zamislimo da pomičemo hiperravninu iz ishodišta za neki vektor $\mathbf{a}$. Jednadžba ravnine postaje $(\mathbf{x} - \mathbf{a})^T\mathbf{w} = 0$ uz odmak $b = \mathbf{a}^T\mathbf{w}$ iliti projekciju vektora $\mathbf{a}$ na vektor $\mathbf{w}$. Bez gubitka na općenitosti možemo odabrati $\mathbf{a}$ okomit

na hiperravninu, te u tom slučaju dužina ||**a**|| = |b| / ||**w**|| predstavlja najmanju udaljenost ishodišta i hiperravnine.

Definirajmo dvije nove hiperravnine paralelne sa ravninom kojom dijelimo grupe. One pretstavljaju ravnine koje prolaze najbližim primjerima za učenje sa obje strane. Nazivat ćemo ih potpornim hiperravninama[21] jer podatkovni vektori koje sadrže podupiru ravninu.

Definirajmo i udaljenost tih hiperravnina od glavne, dijeleće, kao *d*+ i *d*-. Margina γ je definirana kao *d*+ + *d*-. Cilj nam je pronaći marginu koja je maksimalna, dok su podupiruće hiperravnine jednako udaljene od glavne. Za podupiruće hiperravnine možemo zapisati sljedeće jednadžbe:

$$\mathbf{w}^T\mathbf{x} = b + \delta$$

$$\mathbf{w}^T\mathbf{x} = b + \delta$$

Primjećujemo kako je problem pre-parametriziran te umjesto **w**, b i δ možemo koristiti parametar α tako da i dalje jednadžba za **x** bude zadovoljena. Da se riješimo ove dvosmislice potrebno je da δ = 1 tj. postavljamo skalu problema.

Zatim možemo i izračunati vrijednosti za d+ = (||b+1| - |b||) / ||**w**|| = 1 / ||**w**|| (ovo je točno samo u slučaju ako b $\notin$ (-1,0) jer tada ishodište ne pada između hiperravnina. Ako je b $\notin$ (-1,0) potrebno je koristiti d+ = (||b+1| + |b||) / ||**w**|| = 1 / ||**w**||). Dakle, margina je dva puta ta vrijednost γ = 2 / ||**w**||.

Sa gornjim definicijama možemo zapisati uvjete koje rješenje mora zadovoljiti,

$$\mathbf{w}^T\mathbf{x}_i - b \le -1 \qquad \forall\, y_i = -1$$

$$\mathbf{w}^T\mathbf{x}_i - b \ge +1 \qquad \forall\, y_i = +1$$

ili u jednadžbi

$$y_i\,(\mathbf{w}^T\mathbf{x}_i - b)\ -1 \ge 0$$

Formulirajmo primarni problem SVM-a:

$$\text{minimizirati} \qquad \frac{1}{2}\,||\mathbf{w}||^2$$

$$\text{uz uvjet} \qquad y_i\,(\mathbf{w}^T\mathbf{x}_i - b)\ -1 \ge 0 \qquad \forall\, i$$

---

[21] Eng. Support hiper-plane

ili riječima – nastojimo maksimizirati marginu, uz uvjet da svi slučajevi iz skupa za učenje padaju s vanjske strane potpornih hiperravnina. Slučajevi koji leže na potpornoj hiperravnini nazivaju se potporni vektori[22] jer podupiru hiperravninu i dakle određuju rješenje problema.

Primarni problem može se riješiti kvadratnim programom. Ipak, ovako formulirani problem ne možemo riješiti jezgrenim načinom jer ne ovisi samo o skalarnom produktu podatkovnih vektora. Transformirat ćemo ga u dualni oblik tako da ga prvo zapišemo koristeći Lagrangiana,

$$\mathcal{L}(\mathbf{w},b,\alpha) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{N}\alpha_i \; [y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1]$$

Rješenje koje minimizira primarni problem uz dani uvjet dano je sa $\min_w \max_\alpha \mathcal{L}(\mathbf{w}, \alpha)$ tj. dolazimo do problema sedla. Kad je originalna objektivna funkcija konveksna, (i samo tada), možemo zamijeniti minimizaciju i maksimizaciju. Koristeći ovo, dolazimo do novog uvjeta za vektor **w** koji mora vrijediti u točci sedla. Derivirajući po **w** i b dolazimo do

$$\mathbf{w} - \sum_i \alpha_i \, y_i\mathbf{x}_i = 0 \qquad \Rightarrow \qquad \mathbf{w}^* = \sum_i \alpha_i \, y_i\mathbf{x}_i$$

$$\sum_i \alpha_i \, y_i = 0$$

Uvrštavanjem natrag u Lagrangian problem postaje dualni

$$\text{maksimizacija } \mathcal{L}D = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{ij}\alpha_i\alpha_j \, y_iy_j\mathbf{x}_i^T\mathbf{x}_j$$

$$\text{uz uvjet} \qquad \sum_i \alpha_i \, y_i = 0$$

$$\alpha_i \geq 0 \forall i$$

Dualna formulacija je također kvadratični program, ali broj varijabli, α_i u ovom slučaju je jednak broju podatkovnih vektora, *N.* Ključna stvar je ipak da ovaj problem ovisi o $\mathbf{x}_i$ samo preko skalarnog produkta $\mathbf{x}_i^T\mathbf{x}_j$ te se uvodi supstitucija $\mathbf{x}_i^T\mathbf{x}_j \rightarrow k(\mathbf{x}_i,\mathbf{x}_j)$.

Teorija dualnost jamči da će za konveksni problem, dualni problem biti konkavan, i nadalje da jedinstveno rješenje primarnog problema odgovara jedinstvenom rješenju dualnog problema. Vrijedi $\mathcal{L}P(\mathbf{w}^*) = \mathcal{L}D(\alpha^*)$.

---

[22] Eng. Support vector

Nadalje moramo proučiti uvjete koji vrijede u točci sedla dakle moraju vrijediti i za rješenje. Zovu se KKT [23] uvjeti. Ovi uvjeti su općenito potrebni i dovoljni za problem konveksne optimizacije. Dobivaju se deriviranjem primarnog problema. Također potrebni su nam postavljeni uvjet nejednakost i ograničenje Lagrangeovih multiplikatora na nenegativnost. Te posljednje, važan "komplementarni uvjet" treba biti zadovoljen

$$\partial_w \mathcal{L}P = 0 \quad \rightarrow \quad \mathbf{w} - \sum_i \alpha_i \, y_i \mathbf{x}_i = 0$$

$$\partial_b \mathcal{L}P = 0 \quad \rightarrow \quad \sum_i \alpha_i \, y_i = 0$$

$$y_i \, (\mathbf{w}^T \mathbf{x}_i - b) \; -1 \geq 0 \qquad \forall \, i$$

$$\alpha_i \geq 0 \forall i$$

$$\alpha_i \, [\, y_i \, (\mathbf{w}^T \mathbf{x}_i - b) \; -1 \,] = 0$$

Posljednja jednadžba može biti iznenađujuća. Ukazuje na dva slučaja, ili vrijedi uvjet nejednakosti, ali nije zasićen: $y_i \, (\mathbf{w}^T \mathbf{x}_i - b) \; -1 > 0$ te u tom slučaju vrijednosti za α_i moraju biti 0. Ako je uvjet nejednakost zasićen $y_i \, (\mathbf{w}^T \mathbf{x}_i - b) \; -1 = 0$ vrijednost α_i mogu poprimiti bilo koju vrijednost $\alpha_i \geq 0$. Zasićen uvjet nejednakosti zovemo „aktivan", dok ne zasićen nazivamo „neaktivan". Svi slučajevi iz skupa za učenje sa $\alpha_i > 0$ pretstavljaju aktivne uvjete na potpornoj hiperravnini te se zovu potporni vektori. Obično ih ima malo, i zovu ih još „rijetka" rješenja (većina α nestaje).

Zainteresirani smo i za funkciju f(.) koju možemo koristiti za klasifikaciju:

$$f(\mathbf{x}) = \mathbf{w}^{*T} \mathbf{x} - b^* = \sum_i \alpha_i \; y_i \, \mathbf{x}_i^T \, \mathbf{x} - b^*$$

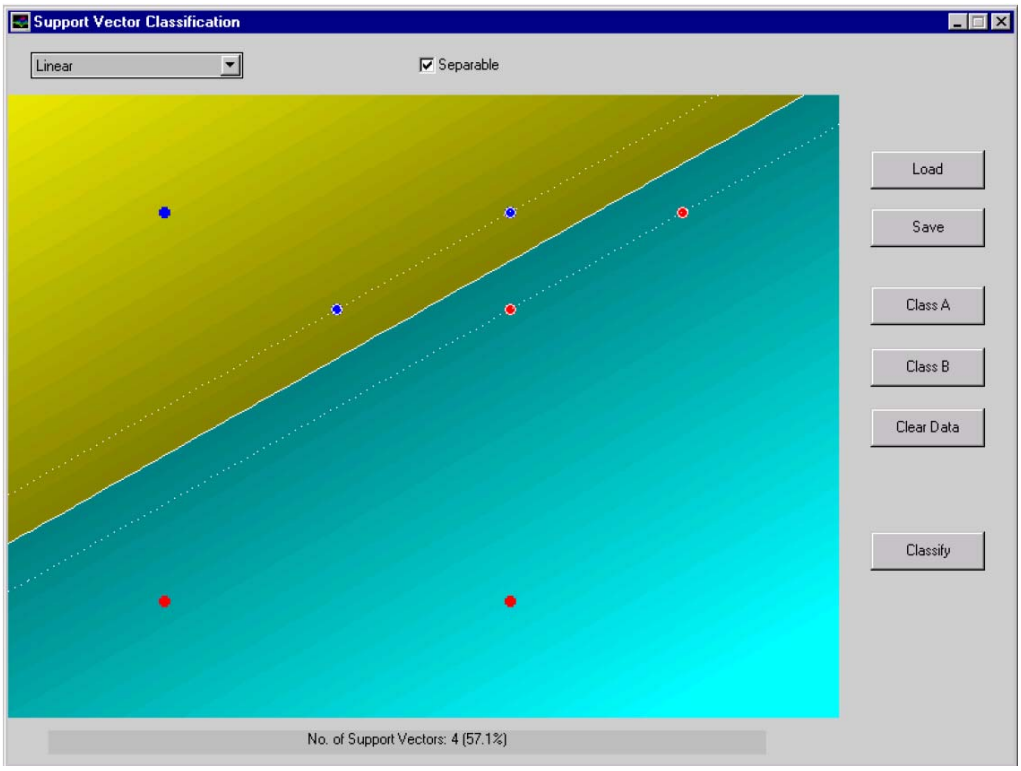
Primjenom KKT uvjeta dolazimo do rješenja za b* koristeći „komplementarni uvjet"

$$b^* = (\sum_j \alpha_j \, y_j \, \mathbf{x}_j^T \, \mathbf{x} - y_i) \quad i \text{ je potporni vektor}$$

[23] Kratica stoji umijesto Karush-Kuhn-Tucker uvjeta

gdje smo koristili $y_i^2 = 1$. Koristeći jedan potporni vektor može se odrediti b, ali zbog numeričke stabilnosti bolje je koristiti njihov prosjek.

Najvažniji zaključaj je opet da se funkciju f(.) može izraziti samo preko skalarnog produkta $\mathbf{x}_i^T\mathbf{x}_i$ koji se može zamijeniti sa jezgrenom matricom $k(\mathbf{x}_i,\mathbf{x}_j)$ za pomak u više dimenzionalni nelinearni prostor. Nadalje, kako je α rijetko, ne moramo koristiti puno elemenata jezgrene matrice za predviđanje klasifikacije novog ulaza **x**.



Slika 5. Primjer optimalne hiperravnine sa SV-ima na marginama [24]

Linearno neodvojivi slučajevi

Svi skupovi nisu linearno odvojivi te prema tome trebamo promijeniti gornje formulacije. Očito je da problem leži u uvjetu koji ne može uvijek biti ispunjen. Promijenimo taj uvjet koristeći „olabavljenu varijablu“ ξ_i

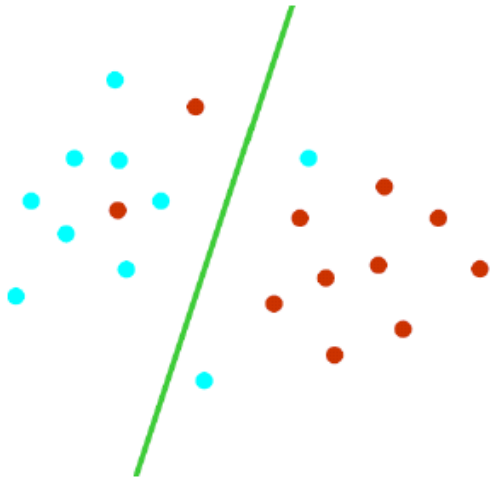
$$\mathbf{w}^T\mathbf{x}_i - b \le -1 + \xi_i \quad \forall\, y_i = -1$$

[24] Slika je dobivena pomoću sučelja MATLAB SVM Toolbox, Steve Gunn

$$\mathbf{w}^T\mathbf{x}_i - b \geq +1 - \xi_i \quad \forall\, y_i = +1$$

$$\xi_i \geq 0 \forall i$$

Varijable ξ_i omogućuju kršenje uvjeta, ali potrebno je i njih kazniti – kako se ξ_i odabire sve većim, gornji uvjet postaje beskoristan. Funkcija kazne, u obliku $C(\sum_i \xi_i)^k$ vodi konveksnom optimizacijskom problemu za pozitivne cjelobrojne *k*. Za *k = 1,2* još je uvijek kvadratni program (QP). Odabrat ćemo *k =1*. C kontrolira razmjenu između margine i kazne.



Slika 6. Općenita optimalna dijeleća hiperravnina

Da bude s krive strane potporne hiperravnine, podatkovnom slučaju potrebno je $\xi_i > 1$. Dakle, sumu $\sum_i \xi_i$ možemo interpretirati kao mjeru koliko su prekršaji teški, te je gornja granica za broj prekršaja.

Novi primarni problem postaje

$$\text{minimizirati} \quad \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i$$

$$\text{uz uvjet} \quad y_i\,(\mathbf{w}^T\mathbf{x}_i - b) - 1 + \xi_i \geq 0 \quad \forall\, i$$

$$\xi_i \geq 0 \forall i$$

te vodi do Lagrangiana

$$\mathcal{L}(\mathbf{w}, b, \xi_i, \alpha, \mu) = \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i - \sum_{i=1}^{N} \alpha_i \,[y_i\,(\mathbf{w}^T\mathbf{x}_i - b) - 1 + \xi_i] - \sum_{i=1}^{N} \mu_i \xi_i$$

Iz čega izvodimo KKT uvjete

1. $\partial_w \mathcal{L}P = 0 \rightarrow$ $\mathbf{w} - \sum_i \alpha_i y_i \mathbf{x}_i = 0$

2. $\partial_b \mathcal{L}P = 0 \rightarrow$ $\sum_i \alpha_i y_i = 0$

3. $\partial_\xi \mathcal{L}P = 0 \rightarrow$ $C - \alpha_i - \mu_i = 0$

4. uvjet 1 $y_i (\mathbf{w}^T \mathbf{x}_i - b) - 1 + \xi_i \geq 0 \quad \forall i$

5. uvjet 2 $\xi_i \geq 0 \forall i$

6. uvjet multiplikatora 1 $\alpha_i \geq 0 \forall i$

7. uvjet multiplikatora 2 $\mu_i \geq 0 \forall i$

8. „komplementarni uvjet“ 1 $\alpha_i [y_i (\mathbf{w}^T \mathbf{x}_i - b) - 1 + \xi_i] = 0$

9. „komplementarni uvjet“ 1 $\mu_i \xi_i = 0$

Iz ovih uvjeta dedukcijom dolazimo so sljedećega. Ako pretpostavimo da je $\xi_i > 0$, tada je $\mu_i = 0$, dakle $\alpha_i = C$(1) i $\xi_i = 1 - y_i (\mathbf{w}^T \mathbf{x}_i - b)$. Također kad je $\xi_i = 0$ slijedi $\mu_i > 0$ i $\alpha_i < C$. Nadalje za $\xi_i = 0$ vrijedi i $y_i (\mathbf{w}^T \mathbf{x}_i - b) - 1 = 0$, te $\alpha_i > 0$. U drugom slučaju, ako je $y_i (\mathbf{w}^T \mathbf{x}_i - b) - 1 > 0$ tada je $\alpha_i = 0$. Kao i prije, za podatke koji nisu potporni vektori i s prave strane ravnine vrijedi $\alpha_i = \xi_i = 0$ (svi uvjeti su neaktivni). Na potpornoj ravnini, još uvijek vrijedi $\xi_i = 0$, ali je sad $\alpha_i > 0$. Napokon, za podatke sa krive strane potporne hiperravnine α_i teži $\alpha_i = C$, a ξ_i uravnotežuje kršenje uvjeta tako da vrijedi $y_i (\mathbf{w}^T \mathbf{x}_i - b) - 1 + \xi_i = 0$.

Geometrijski možemo izračunati razmak između potporne ravnine i slučaja koji krši uvjet kao $\xi_i / ||\mathbf{w}||$. Kako je ravnina definirana sa $y_i (\mathbf{w}^T \mathbf{x}_i - b) - 1 + \xi_i = 0$ paralelna sa potpornom hiperravninom te je udaljena od ishodišta za $|1 + y_i b - \xi_i| / ||\mathbf{w}||$, a potporna ravnina je udaljena $|1 + y_i b| / ||\mathbf{w}||$, dolazimo do gornje udaljenosti.

Naposlijetku, trebamo dualizirati problem zbog učinkovitog rješavanja pomoću jezgrenih metoda. Opet koristimo KKT uvjete da se riješim **w**, b i ξ,

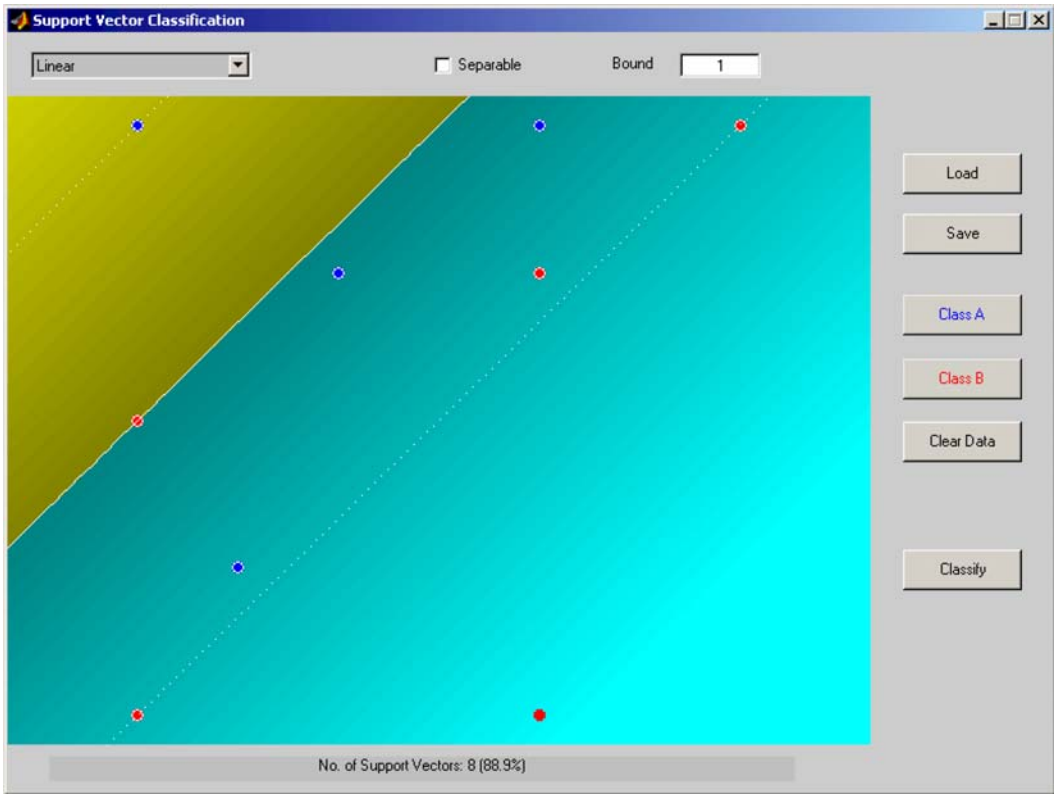
$$\text{maksimizacija } \mathcal{L}_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

uz uvjet $\sum_i \alpha_i \; y_i = 0$

$$0 \le \alpha_i \le C \forall i$$

Iznenađujuće je da je to skoro isti kvadratni problem(QP) kao i prije, ali sa dodatnim uvjetom na multiplikatoru α_i koji je sad zatvoren u kutiju. Uvjet je izveden iz α_i = C - μ_i te $\mu_i \ge 0$. Napomenimo još i da ovisi samo o skalarnom produktu vektora $\mathbf{x}_i^T \mathbf{x}_i$.

Parametar C uvodi dodatnu kontrolu nad klasifikatorom te ga je potrebno odrediti koristeći znanje o šumu u podacima. Uz C = 1 potporni vektori više ne trebaju ležati na margini te se orijentacija hiperravnine i širina margine mijenjaju.

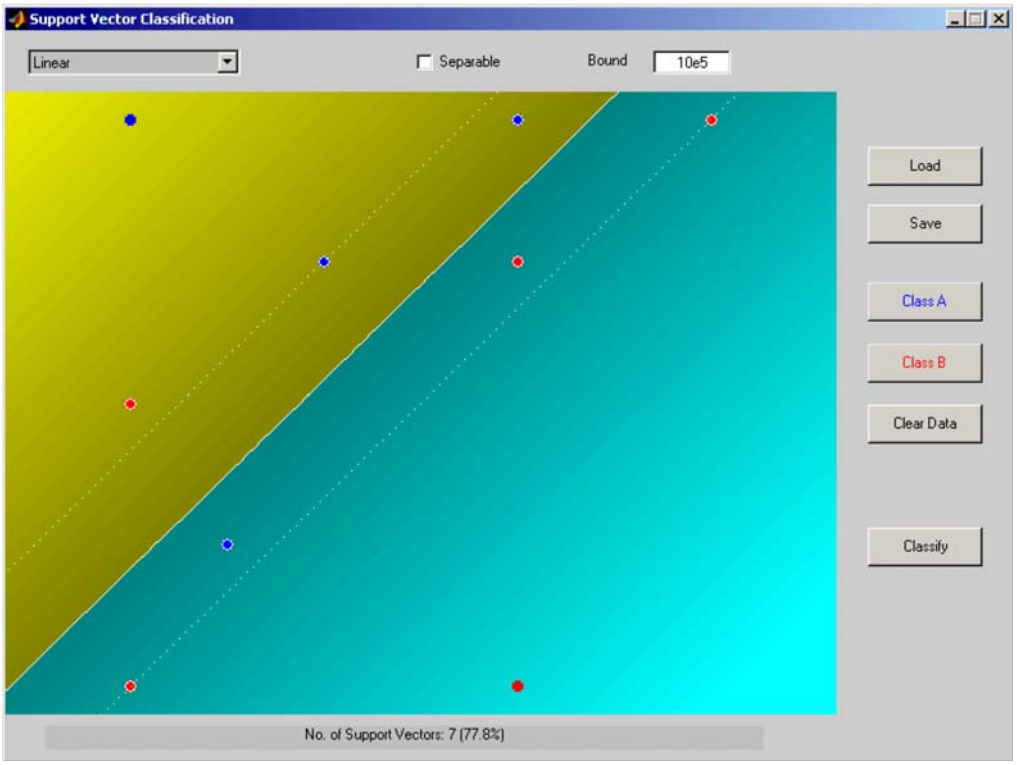


Slika 7. Primjer optimalne hiperravnina sa SV-ima na marginama (C = 1)

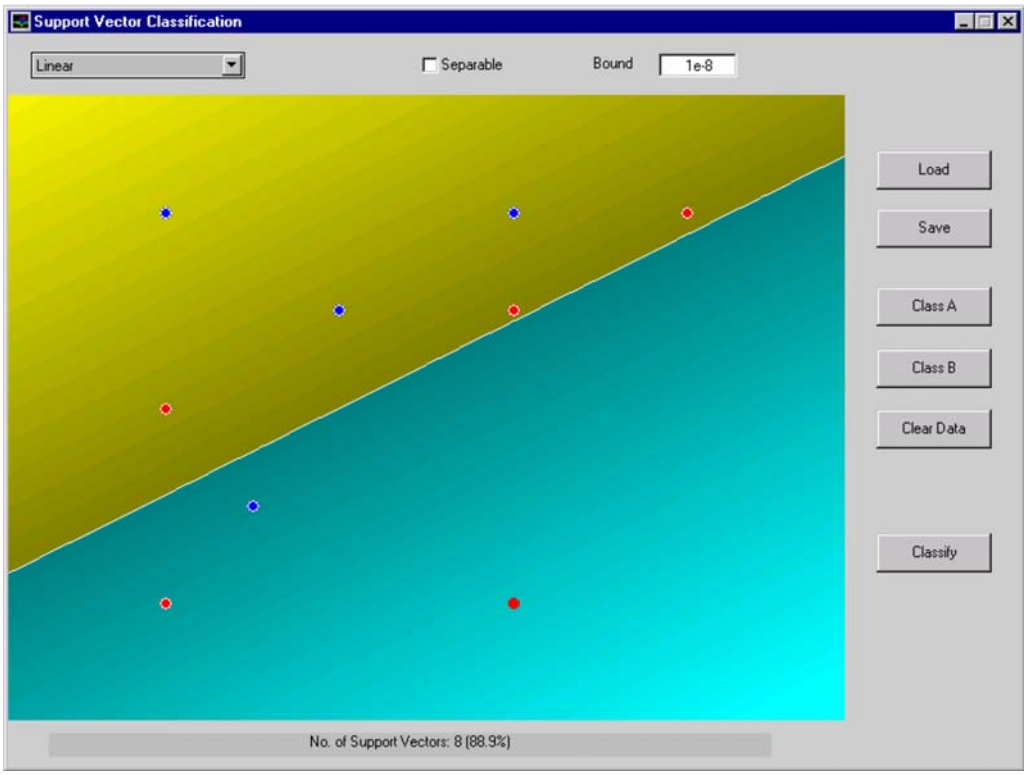
U ograničenje kad $\lim C \to \infty$ rješenje teži optimalnoj hiperravnini dobivenoj na linearno nerazdvojivim podacima.

U ograničenje kad $\lim C \to 0$ rješenje teži maksimiziranju margine, te će nakon nekog vremena svi Lagrangovi multiplikatori poprimiti vrijednost C. Klasifikator daje naglasak na minimizaciju pogrešno klasificiranih slučajeva, ali na taj način da povećava marginu, stvara marginu velike širine. Kao posljedica smanjenja C-a veličina margine se povećava.

Za određivanje parametra C koristi se postupak krosvalidacije.



Slika 8. Primjer optimalne hiperravnine ($C = 10^5$)



Slika 9. Primjer optimalne hiperravnine ($C = 10^{-8}$)

2.5.1 Upotreba metode potpornih vektora

Klasifikator metodom potpornih vektora koristimo u sljedećim koracima:

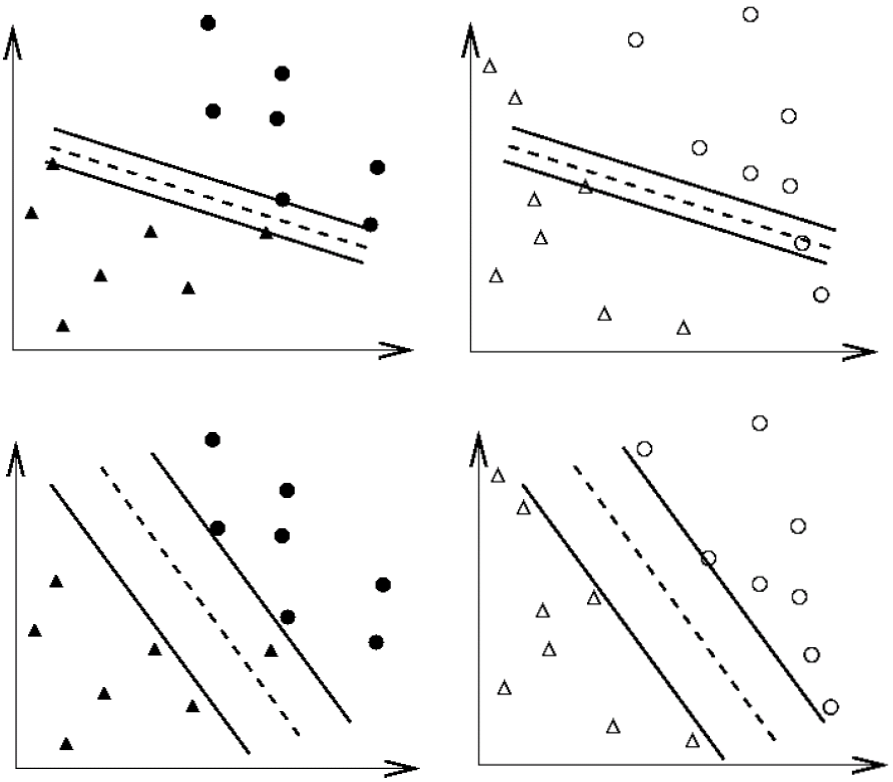
- Zapišemo podatke u formatu pogodnom za SVM program
- Provedemo jednostavno skaliranje podataka(ako je potrebno)
- Razmotrimo jezgrene funkcije, npr. RBF
- Koristeći krosvalidaciju dolazimo do najboljih parametara C i γ
- Koristeći najbolje parametre C i γ učimo na cijelom skupu za učenje[25]
- Testiranje

Skaliranje podataka je vrlo važno, kao i skaliranje pri korištenju neuronskih mreža, tako i kod SVM-a. Najveća je prednost izbjegavanje dominacije atributa sa velikim numeričkim rasponom prema onima sa malim rasponom. Druga prednost je i izbjegavanje numeričkih teškoća pri izračunavanju. Kako jezgrene vrijednosti ovise o skalarnom produktu preslikanih vektora velike vrijednosti atributa mogle bi uzrokovati numeričke probleme. Preporuča se linearno skaliranje [-1, +1] ili [0, 1]. Također potrebno je skalirati i testni skup na isti način.

Pri odabiru jezgrene funkcije, RBF je prvi izbor. RBF nelinearno preslikava uzorke u višedimenzionalni prostor, tako da, za razliku od linearnog jezgrenog modela, može riješiti i slučajeve kad su uzorci linearno neodvojivi. Nadalje, linearna jezgrena funkcija je samo specijalan slučaj RBF-a te se može dokazati da linearna jezgrena funkcija sa parametrom kazne C* ima iste performanse kao RBF sa (C, γ). I sigmoidalna jezgrena funkcija se ponaša kao RBF za određene parametre. Sljedeći razlog u odabiru stavlja RBF ispred polinomialne jezgrene funkcije jer ona koristi više hiperparametara koji utječu na složenost rješavanja. RBF se također lakše izračunava. Vrijednosti jezgrene funkcije su $0 \leq K_{ij} \leq 1$ za razliku od polinomialne funkcije gdje vrijednosti idu do beskonačnosti. Još moramo napomenuti da sigmoidalna jezgrena funkcija nije ispravna za neke parametre (Chang et al., 2005.).

[25] Najbolji parametri mogu biti neprilagođeni za cijeli skup za učenje, ali u praksi oni parametri dobiveni krosvalidacijom pogodni su za cijeli skup za učenje.

Pri korištenju RBF-a potrebno je podesiti dva parametra: C i γ. Potrebno je odabrati takve parametre koji čim točnije predviđaju oznake testnih podataka, ali to nužno ne znači da je potrebno dobiti veliku točnost na skupu za učenje, tj. točno predviđanje oznaka na skupu na kojem su one već poznate. Obično se skup za učenje dijeli na dva dijela u kojem jedan postaje nepoznat pri učenju klasifikatora. Točnost na ovakvom skupu preciznije govori o klasificiranju nepoznatih podataka. Ovakva procedura zove se krosvalidacija. Općenito se krosvalidacija izvršava na sljedeći način: podijeli se skup za učenje na *v* jednakih dijelova; sekvencijalno jedan podskup se testira koristeći klasifikator dobiven na preostalih *v - 1* podskupu. Dakle, svaka instanca skupa za učenje predviđa se samo jedanput tako da je krosvalidacijska točnost postotak točno klasificiranih podataka. Također krosvalidacija može spriječiti pretreniranost. Na sljedećoj slici puni krugovi i trokuti pretstavljaju skup za učenje, dok prazni skup za testiranje. Prvi klasifikator postiže loše rezultate na skupu za testiranja jer je pretreniran na skupu za učenje. Također ni točnost pri krosvalidaciji nije visoka. Drugi klasifikator bez pretreniranosti, daje bolje krosvalidacijske rezultate kao i testnu točnost (Chang et al., 2005.).



Slika 10. Pretrenirani klasifikator i bolji klasifikator

Tipični parovi C i γ isprobavaju se te se odabire onaj s najvećom krosvalidacijskom točnošću. Pokazalo se da je eksponencijalno rastuća sekvenca C i γ praktična metoda pri njihovom određivanju (npr. C = 2^{-5}, 2^{-3},…, 2^{15}, γ = 2^{-15}, 2^{-13},…, 2^{3})

2.5.2 Biblioteka funkcija LIBSVM te OSU SVM

Prilikom implementacije metode potpornih vektora, korištištena je javno dostupna biblioteka funkcija LIBSVM[26] verzija 2.6 objavljena u travnju 2004., pisana u programskom jeziku C++ te Javi. Detalj o njoj mogu se pronaći na [http://www.csie.ntu.edu.tw/~cjlin/libsvm/]. LIBSVM biblioteka implementira više tipova klasifikatora. Ova implementacija koristi tip nazvan C-SVC koji odgovara klasifikatoru sa slabom marginom. Korištena je RBF jezgrena funkcija.

Također pri klasifikaciji podataka reduciranih pomoću PCA-a i CCA-a korišteno je MATLAB sučelje na LIBSVM – OSU SVM, autori su Junshui Ma i Stanley Ahalt sa Ohio State University. Prema autorima, dovoljno je brz da može konkurirati svim SVM implementacijama u C-u, te nasljeđuje LIBSVM – upravljanje memorijom što ga čini upotrebljivim i na ogromnim skupovima za učenje.

Pri dobivanju krosvalidacijskih parametara C i γ korišten je program grid.py napisan u programskom jeziku Python. Program grid.py nalazi se u na stranicama LIBSVM – a.

---

[26] Autori ove biblioteke su Chih-Chung Chang i Chih-Jen Lin, Department of Computer Science and Information Engineering, National Taiwan University.

2.5.3 Klasifikacija u više klasa

Prethodno opisana metoda potpornih vektora klasificira podatke u dvije klase sa oznakama $y_i \in \{-1, +1\}$. Za višeklasno klasificiranje odabrana je metoda „jedan-prema-jedan“[27](Hsu et al.,2002.) u kojoj se stvara *k(k-1)/2* klasifikator koji odvaja po dvije različite klase (Chang et al., 2005.).

U klasifikaciji se koristi strategija glasovanja, pri ćemu svaki binarni klasifikator daje svoj glas. Svaki ulazni vektor klasificira se prema najvećem broju glasova. Za slučajeve u kojima dvije klase imaju jednak broj glasova, iako se ne smatra dobra tehnikom, odabire se klasa sa manjim indeksom.

[27] Eng. One-against-one

3 REZULTATI

3.1 Ulazni podaci

Kao ulazni podaci na raspolaganju su dvije baze, Vjesnik – baza članaka iz hrvatskih dnevnih novina Vjesnik u razdoblju od 2000. – 2003., te Croatia Weekly (brojevi 5-118) u razdoblju od 1998.-2000.

3.1.1 Croatia Weekly

Bazu članaka iz Hrvatsko-engleskoga paralelnoga korpusa za potrebe projekta prof. dr. Bojane Dalbelo-Bašić, Croatia Weekly, izradio je prof. Marko Tadić sa Zavoda za lingvistiku Filozofskoga fakulteta Sveučilista u Zagrebu. Izvor su novine Croatia Weekly, brojevi 5-118, koji su izlazili u razdoblju od 1998.- 2000.

Baza sadrži 4 kategorije:

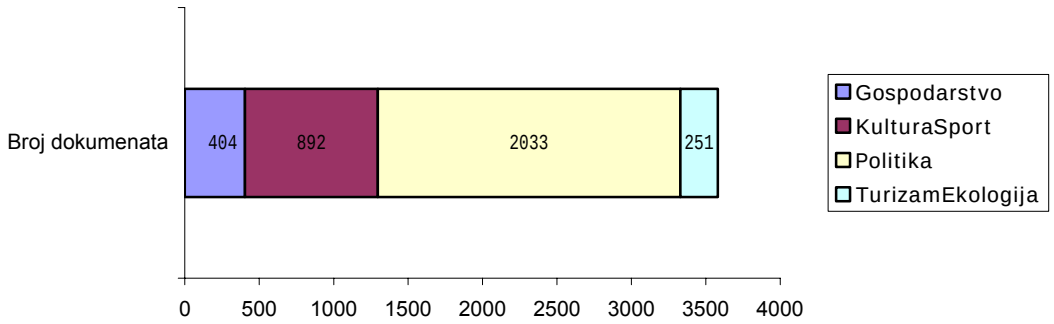
po - politika

go - gospodarstvo

te - turizam i ekologija

ks - kultura i sport

U korpusu se nalazi 3.580 članaka na hrvatskom jeziku i isto toliko članaka na engleskom jeziku. Članci na hrvatskom jeziku nisu normalizirani – iz njih su samo izbačene stop riječi. Jedan od problema ove baze je nejednako raspoređen broj dokumenata po kategorijama. Skup za učenje i skup za testiranje svaki sadrže 50% dokumenta svake kategorije. Za usporedbu, skup za učenje sadrži 1017 dokumenata politike, ali samo 128 dokumenata turizma i ekonomije.



Slika 11. Broj dokumenata po kategorijama za hrvatsko-engleski korpus.

3.1.2 Baza novinskih članaka Vjesnik

Baza Vjesnikovih članaka također je izradio prof. Marko Tadić sa Zavoda za lingvistiku Filozofskoga fakulteta Sveučilista u Zagrebu. Baza sadrži sve novinske članke (Vjesnik između 2000. i 2003. godine), preuzete iz Hrvatskog nacionalnog korpusa [http://www.hnk.ffzg.hr/].

Sažeta baza se sastoji od 10.000 dokumenata podijeljenih u kategorije:

ck - crna kronika

go - gospodarstvo

ku - kultura

sp - sport

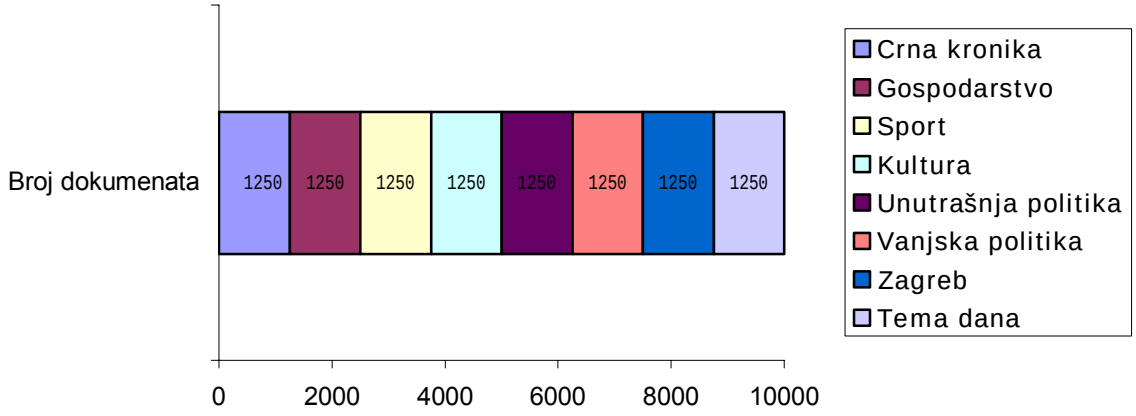
td - tema dana

un - unutarnja politika

vp - vanjska politika

zg - zagreb.

U svakoj kategoriji nalazi se 1.250 dokumenata.



Slika 12. Broj dokumenata po kategorijama za Vjesnikovu bazu članaka

3.1.3 Baza novinskih članaka Vjesnik AMNv1.0

Riječ je o bazi novinskih članaka Vjesnik na kojoj je provedena morfološka obrada postupkom Automatske morfološke normalizacije koju je razvio dipl. ing. Jan Šnajder, asistent na Zavodu za elektroniku, mikroelektroniku, računalne i inteligentne sustave Fakulteta elektrotehnike i računarstva.

Automatska morfološka normalizacija (AMN) je postupak kojim se pojavnice u tekstu svode na svoje morfološke norme. Osim što na taj način smanjujemo dimenzionalnost ulaznih podataka, povećavamo kvalitetu teksta koji želimo kategorizirati jer uklanjamo rasipanje značenja istog pojma na više leksički različitih oblika.

Postoje 3 vrste morfološke normalizacije:

1. *Flektivna*

   Korištena je oznaka "-i" pri označavanju pripadnih baza (eng. Inflective). Eliminira efekt flektivne norme na način da sve oblike neke riječi svodi na jedan oblik, lemu ako je nju moguće automatski odrediti, inače odgovara kombinaciji više lema.

2. *Derivacijska*

   Korištena je oznaka "-id" pri označavanju pripadnih baza. Nakon svođenja različitih oblika riječi na njihovu lemu, nalazi zajedničkog predstavnika više lema, prema derivacijskim pravilima morfologije. Derivacijska pravila morfologije opisuju tvorbu riječi (iz jedne leme u drugu) eventualno mijenjajući vrstu riječi. Smisao derivacijskih normi jest ostvarivanje još veće redukcije dimenzionalnosti ulaznih podataka od one koje se postiže primjenom inflekcijskih pravila. Pored toga postiže se i veća koncentracija značenja nekog pojma. Npr. riječi "kompjuter" i "kompjuterski" imaju vrlo blisko značenje (vezane su uz isti pojam), zato je sigurno poželjno svesti ih na zajednički oblik prije samog postupka klasifikacije.

3. *Terminirajuća*

   Korištena je oznaka "-idt" pri označavanju pripadnih baza. U slučajevima kad ne djeluju derivacijska pravila onda se reže riječ nakon 7-og slova.

Npr. neka je zadana riječ "kompjuterskog". To je pridjev, i njegova lema je nominative jednine muškog roda: "kompjuterski". To bi ujedno bila i inflekcijska norma, ako je AMN postupkom ispravno pronađena. Osnova riječi "kompjuterski" je "kompjuter", pa je derivacijska norma "kompjuter".

Ovom se metodom broj različitih pojavnica u prosjeku smanjio na oko 45% početnog broja pojavnica.

Više informacije o ovoj metodi dostupno je na Internet adresi [http://www.zemris.fer.hr/~jan/amn].

3.2 Rezultati usporedbe PCA i CCA

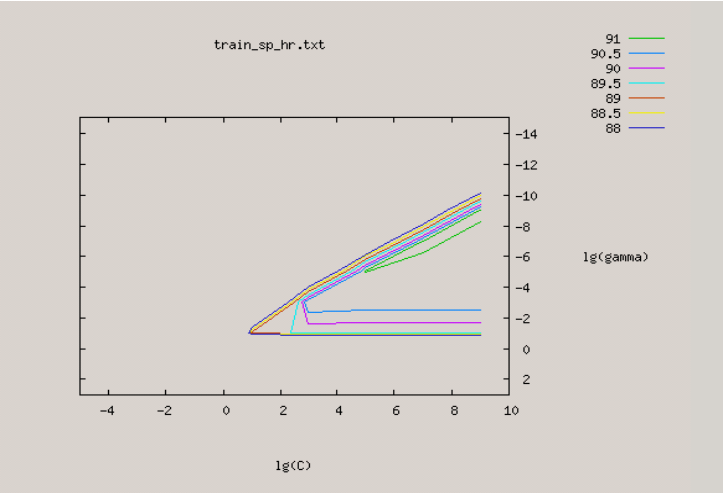
PCA i CCA algoritam implementirani su u MATLAB-u. Korištena je baza Croatia Weekly, hrvatsko-engleski paralelni korpus.

Bitno je spomenuti MATLAB-ovu funkciju `eigs` koja se povezuje na Fortranovu bibiloteku ARPACK, i koristi procedure `DSAUPD`, `DSEUPD`, `DNAUPD`, `DNEUPD`, `ZNAUPD`, i `ZNEUPD`.

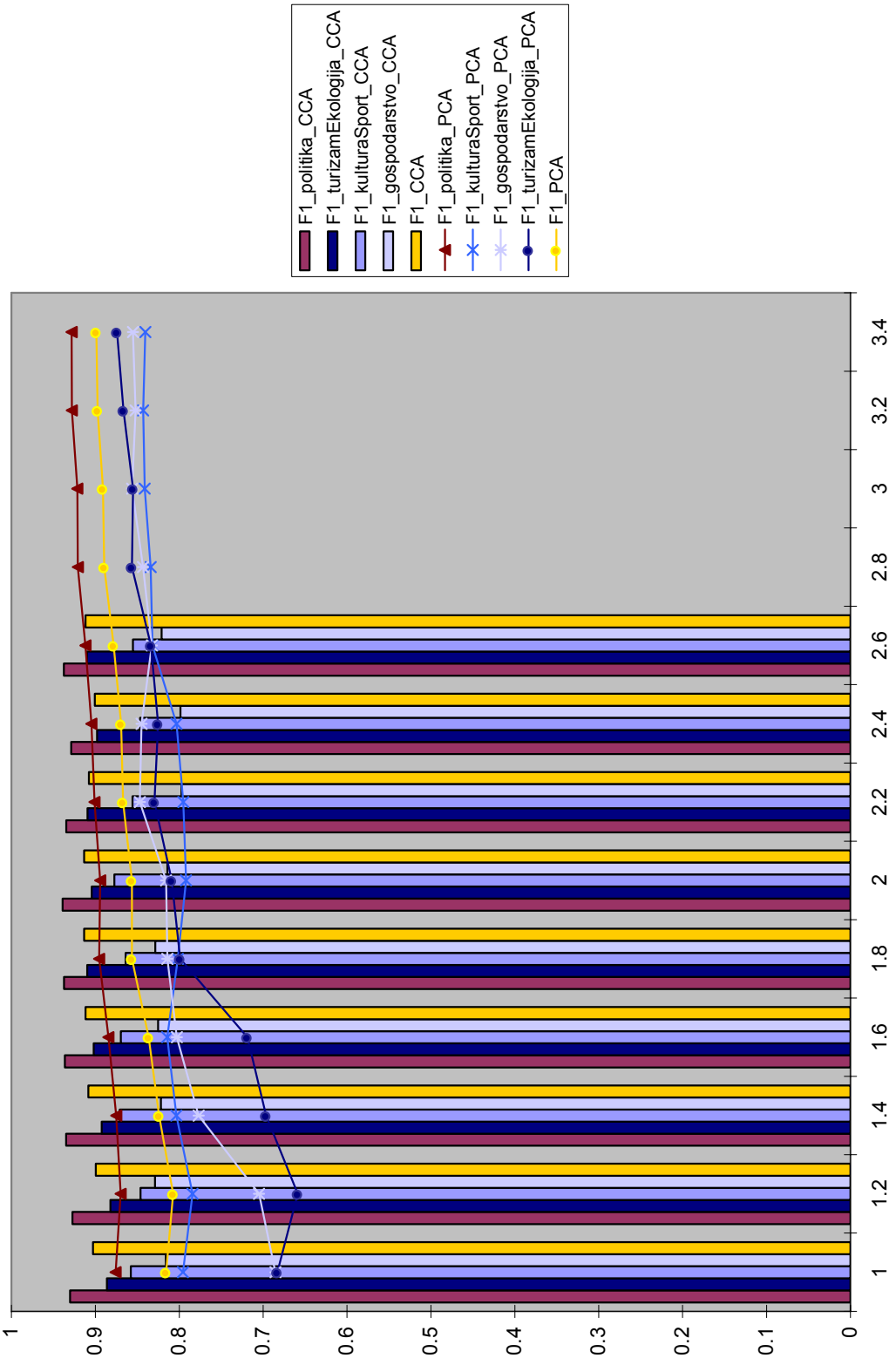
Eigs funkcija koristi Arnoldi metodu za izračunavanje svojstvenih vektora i rješavanje općeg problema svojstvenih vrijednosti rijetkih matrica – velikih matrica sa malim brojem nenul elemenata. Njeno rješenje je aproksimacija međutim upotreba točne dekompozicije za izračunavanje svojstvenih vrijednosti iscrpljuje veliku memoriju računala.

Dok PCA algoritam nastoji pronaći smjerove maksimalne varijance među dokumentima na hrvatskom jeziku, CCA traži smjerove maksimalne kovarijance hrvatskih dokumenata sa engleskim te se skup za učenje i testiranje na hrvatskom jeziku preslikava na novo dobivene osi.

Parametri C i γ za metodu potpornih vektora dobiveni su pomoću programa grid.py napisanog u programskom jeziku Python, koji je sastavni dio LIBSVM-a. Program koristi tehniku krosvalidacije te skup za učenje dijeli na dva dijela: dok na jednom uči, drugi služi za testiranje i obrnuto. Na dobivenom konturnom plotu krosvalidacijske točnosti možemo očitati parametre koji nam odgovaraju. Najveća vrijednost krosvalidacijske točnosti 91.065% dobivena je za parametre *C = 128,* γ *= 0.0078125* te su ti parametri korišteni u svim eksperimentima sa ovom bazom.



Slika 13. Konturni plot krosvalidacijske točnosti na skupu za učenje



Slika 14. Usporedba PCA i CCA redukcije dimenzionalnosti

(mjera F1 po kategorijama i F1 ^{micro})

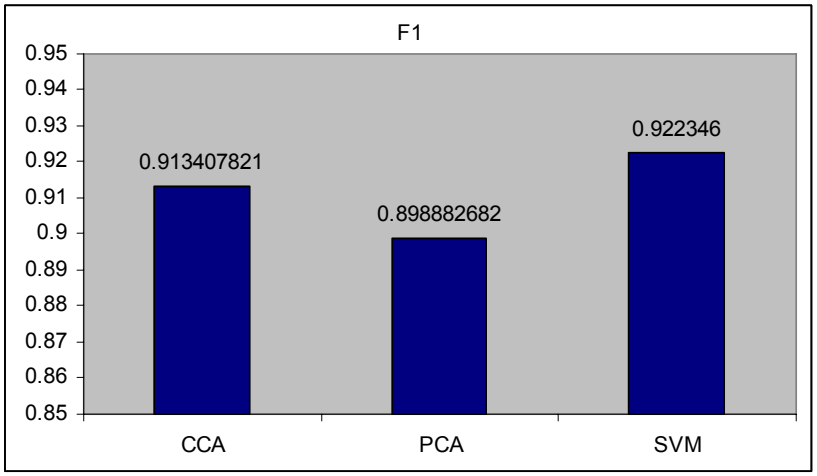
dimenzionlnost k = 10^x

Dimenzionalnost *k* je povećavana od *k = 10* do *k = 500 z*a CCA, te od *k = 10* do *k = 1700* za PCA. Napomenimo da je redukcija dimenzionalnosti ograničena odozgo i njen maksimum je *k = 1790*. To je očito ako se prisjetimo da se oba problema rješavaju pomoću jezgrene matrice dimenzionalnosti *1790 x 1790* ili *broj dokumenata x broj dokumenata*.

Regulacijski parametar CCA algoritma postavljen je na 1.5 uzastopnim ispitivanjem korelacijske matrice.

Cilj nam je procijeniti redukciju dimenzionalnosti općenito te također usporediti PCA i CCA metode. Kako je već navedeno PCA traži osi maksimalne varijance podataka dok CCA traži kovarijance između dokumenata na hrvatskom i engleskom jeziku. Prema rezultatima možemo zaključiti koliko uvođenje kompleksnih oznaka, tj. novog jezika pomaže u klasifikaciji. Prema (Fortuna, 2004.) novi jezik se može nadodati i umjetno, koristeći prijevodne programe kao npr. *Google Language Tools*[28].

Također možemo uspoređivati rezultate dobivene redukcijom dimenzionalnosti i bez redukcije dimenzionalnosti, primjenom metode potpornih vektora na dokumente. Prema rezultatima redukcija dimenzionalnosti daje neznatno gore rezultate nego bez redukcije.

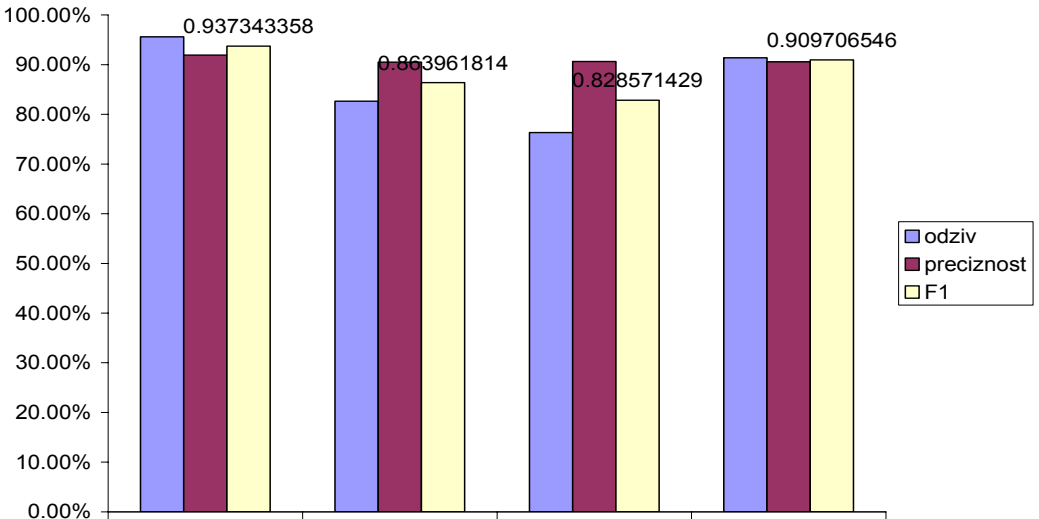


Slika 15. Usporedba mjere F1 ^{micro} za redukciju dimenzionalnosti CCA (k = 63), PCA (k = 1584) te bez redukcije

[28] http://www.google.com/language_tools

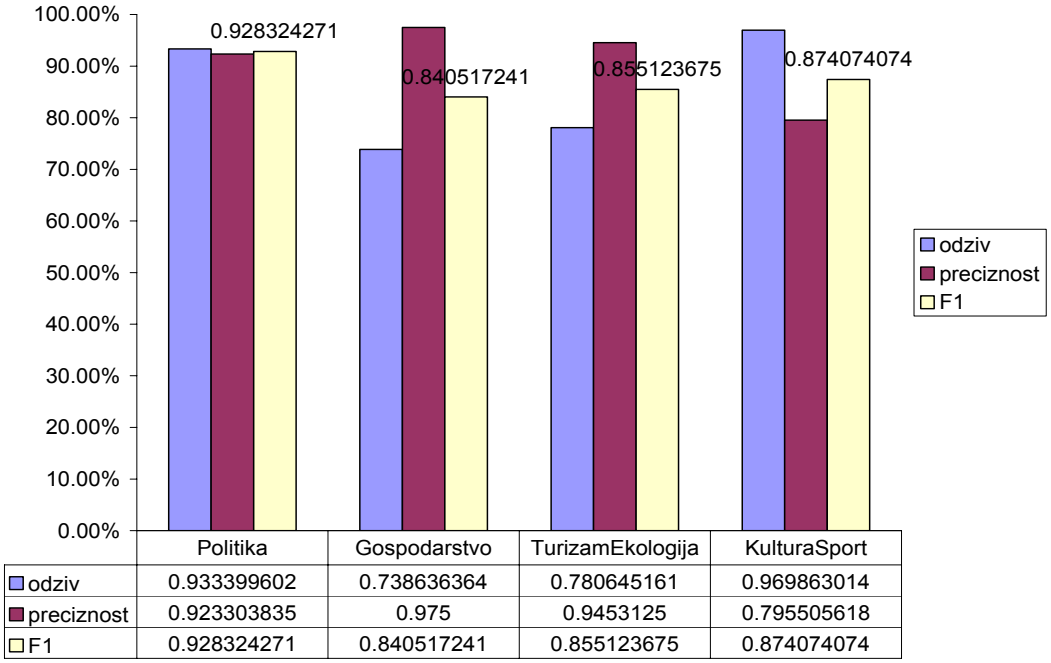
Povećanjem dimenzionalnosti prostora značajki rezultati mjere F1 za PCA i CCA postaju sve sličniji. Sa manjim vrijednostima *k*, PCA ima relativno lošije rezultate u usporedbi s CCA metodom.

Na sljedećim slikama vidimo da u svim kategorijama mjera F1 CCA metode nadmašuje PCA. Najvećih varijacija među njima dolazi u kategoriji gospodarstvo. Kako je odziv za PCA mali, dok je preciznost velika možemo zaključiti da u toj kategoriji PCA propušta velik broj dokumenata koji se krivo klasificiraju. CCA metoda je obuhvatnija, ali također popušta na preciznosti – u tu kategoriju nadodaju se i dokumenti drugih kategorija.



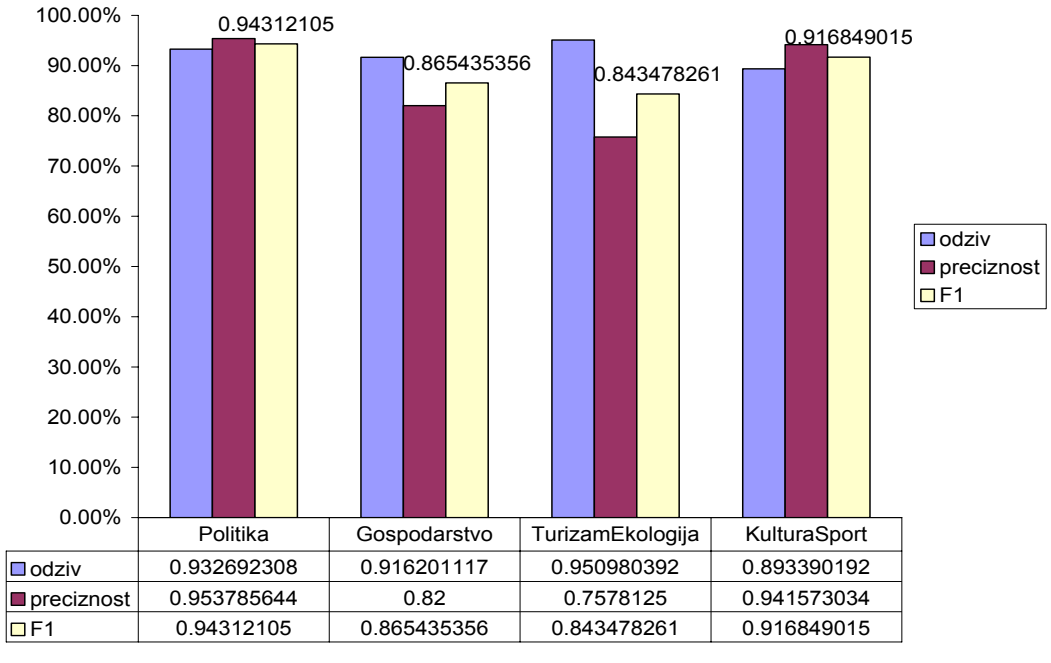
	Politika	Gospodarstvo	TurizamEkologija	KulturaSport
odziv	0.95603272	0.826484018	0.763157895	0.9138322
preciznost	0.919370698	0.905	0.90625	0.905617978
F1	0.937343358	0.863961814	0.828571429	0.909706546

Slika 16. Odziv, preciznost i F1 za CCA (k = 63)



	Politika	Gospodarstvo	TurizamEkologija	KulturaSport
odziv	0.933399602	0.738636364	0.780645161	0.969863014
preciznost	0.923303835	0.975	0.9453125	0.795505618
F1	0.928324271	0.840517241	0.855123675	0.874074074

Slika 17. Odziv, preciznost i F1 za PCA (k = 1584)



	Politika	Gospodarstvo	TurizamEkologija	KulturaSport
odziv	0.932692308	0.916201117	0.950980392	0.893390192
preciznost	0.953785644	0.82	0.7578125	0.941573034
F1	0.94312105	0.865435356	0.843478261	0.916849015

Slika 18. Odziv, preciznost i F1 za SVM

Vjerojatnosti

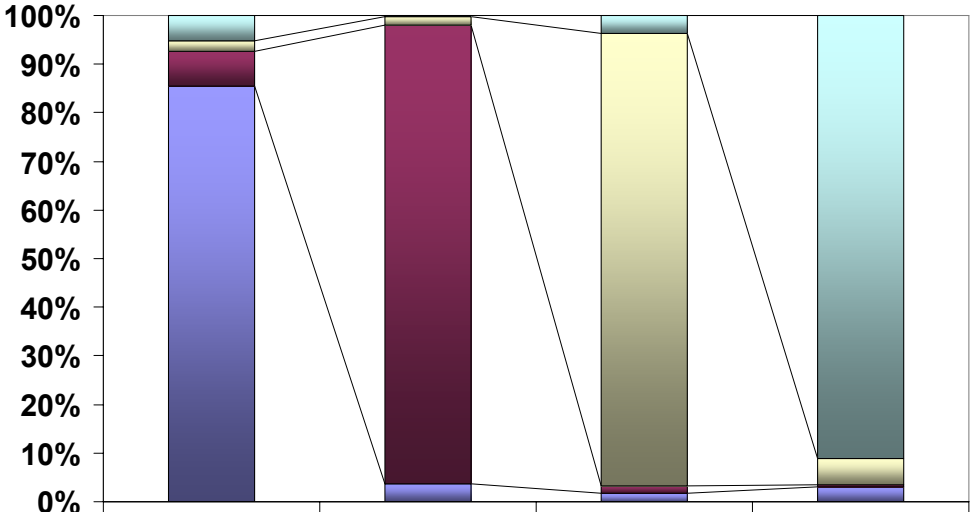
Metoda potpornih vektora kao izlaz predviđa oznake razreda, ali ne daje nikakve vjerojatnosne informacije. Korištene biblioteke funkcija metode potpornih vektora, LIBSVM, proširene su ovakvim proračunima koji su spremljeni u Confusion Matrix. Svaki element matrice daje vjerojatnost

$$\text{ConfMatrix}[i, j] = P(X \text{ u } j \mid X \text{ u } i)$$

ili vjerojatnost klasificiranja dokumenta kategorije *i* u kategoriju *j*.

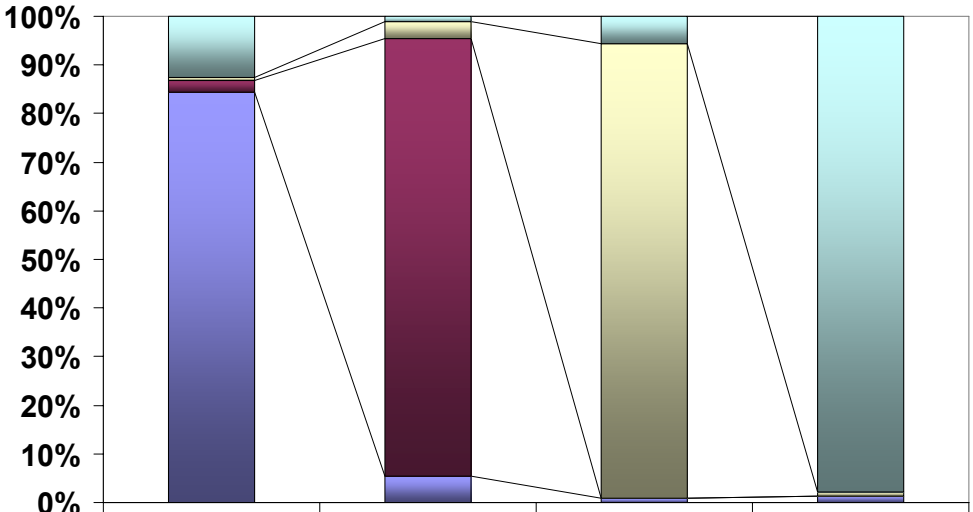
Više o vjerojatnostima i njihovom izračunavanju može se pronaći u (Chang et al. , 2005.)

Zaključke koje dobijemo vjerojatnostima možemo uspoređivati sa odzivom i preciznošću. Tako za PCA kategorije gospodarstvo i turizam i ekologija imaju skoro jednako nizak odziv i visoku preciznost. U Confusion matrici to znači redak popunjen većinom postocima blizu 0 i jednim postotkom malo manjim od 1 koji odgovara dotičnoj kategoriji – ili malo će se drugih kategorija klasificirati u dotičnu – velika preciznost. Također stupac označava odziv, a raznoliko popunjen za ove dvije kategorije daje mali odziv. Veliki odziv za PCA ima kultura i sport – većina dokumenta te kategorije predviđanjem će ostati u toj kategoriji. Mala preciznost te kategorije objašnjavamo velikim brojem dokumenata politike za koje se predviđa da pripadaju kulturi i sportu.



	Politika	Gospodarstvo	TurizamEkologija	KulturaSport
KulturaSport	0.05618	0.0022472	0.035955	0.90562
TurizamEkologija	0.023438	0.015625	0.90625	0.054688
Gospodarstvo	0.075	0.905	0.015	0.005
Politika	0.91937	0.034415	0.016716	0.029499

Slika 19. Confusion Matrix za k = 63 CCA



	Politika	Gospodarstvo	TurizamEkologija	KulturaSport
KulturaSport	0.13708	0.011236	0.05618	0.79551
TurizamEkologija	0.0078125	0.039063	0.94531	0.0078125
Gospodarstvo	0.025	0.975	0	0
Politika	0.9233	0.058014	0.0088496	0.0098328

Slika 20. Confusion Matrix za k = 1584 PCA

4 KONCEPTNI VEKTORI

U ovom poglavlju proučiti ćemo razdiobu veliko dimenzionalnih i rijetkih tekstualnih skupova podataka kao npr. Vjesnikove baze u različite konceptualne kategorije.

4.1 Konceptni vektori i objektivna funkcija

Dokument vektore $\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n}$ koji popunjavaju *riječ x dokument* matricu, ako pretpostavimo da su normalizirani, možemo promatrati kao točke jedinične kugle u $\Re^m$. Nadalje, sve su komponente dokument vektora pozitivne i nalaze se u $\Re^m_{\geq 0}$. Za ovakve vektore skalarni produkt je prirodna mjera sličnosti. Za dva vektora **x** i **y** u $\Re^m_{\geq 0}$ vrijedi

$$\mathbf{x}^T\mathbf{y} = ||\mathbf{x}||\,||\mathbf{y}||\cos\theta(\mathbf{x},\mathbf{y}) = \cos\theta(\mathbf{x},\mathbf{y})$$

gdje je $\theta(\mathbf{x},\mathbf{y})$ kut između dva vektora i $0 \leq \theta(\mathbf{x},\mathbf{y}) \leq \pi/2$.

Produkt $\mathbf{x}^T\mathbf{y}$ naziva se «kosinusna sličnost». Svojstvo dokument vektora da su rijetki omogućava lako i učinkovito računanje konceptne sličnosti.

Za dokument vektore $\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n}$ pretpostavimo razdiobu dokumenata u *k* različitih grupa $\pi_1, \pi_2, \ldots, \pi_k$ tako da

$$\bigcup_{j=1}^{k} \pi_j = \{\mathbf{x_1},\mathbf{x_2},\ldots,\mathbf{x_n}\} \quad \text{i vrijedi} \quad \pi_j \cup \pi_l = \varnothing \text{ ako } j \neq l.$$

Za svaki $1 \leq j \leq k$ računa se srednji vektor ili centroid grupe dokument vektora sadržanih u grupi π_j formulom

$$\mathbf{m}_j = \frac{1}{n_j}\sum_{x\in\pi j} \mathbf{x}$$

gdje je n_j broj dokument vektora u grupi π_j. Kako centroidi ne moraju imati jediničnu normu, njihov smjer se zadržava u konceptnim vektorima

$$\mathbf{c}_j = \frac{\mathrm{m}_j}{||\,\mathrm{m}_j\,||}$$

Za svaki jedinični vektor **z** u $\Re^m$ iz Cauchy – Schwarz nejednakost vrijedi

$$\sum_{x\in\pi j} \mathbf{x}^T\mathbf{z} \leq \sum_{x\in\pi j} \mathbf{x}^T\mathbf{c}_j$$

Prema tome konceptne vektore možemo promatrati kao one vektore koji su najbliži u kosinusnoj sličnosti prema svim dokumentima u grupi $\pi_{j.}$

Produkt $\sum_{x \in \pi j} \mathbf{x}^T \mathbf{c}_j$ možemo koristiti kao mjeru koherencije ili kvalitete grupa. Ako su svi dokumenti u grupi identični, tada će prosječna koherencija grupe biti najveća tj. imat će vrijednost 1. S druge strane, ako vektori u grupi jako variraju, prosječna koherencija će biti blizu 0. Svako particioniranje u grupe $\{ \pi_j \}^k_{j=1}$ mjerimo pomoću *objektivne funkcije*[29] (Dhillon et al., 2001.)

$$Q(\{ \pi_j \}^k_{j=1}) = \sum_{j=1}^{k} \sum_{x \in \pi j} \mathbf{x}^T \mathbf{c}_j$$

4.2 Spherical k-means algoritam

Sljedeći korak je pronaći particioniranje dokument vektora $\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_n}$ u *k* različitih grupa $\pi_1^*, \pi_2^*, ..., \pi_k^*$ tako da se maksimizira objektivna funkcija tj. tražimo rješenje sljedećeg optimizacijskog problema

$$\{ \pi_j \}^k_{j=1} = \arg\max_{\{ \pi_j \}^k_{j=1}} Q(\{ \pi_j \}^k_{j=1})$$

Kako je problem NP težak za pronalaženje rješenja koristi se heuristički iterativni algoritam - spherical k-means algoritam:

1. Počinje se sa proizvoljnom razdiobom dokument vektora $\{ \pi_j^{(0)} \}^k_{j=1}$. Indeks iteracije t jednak je 0, a $\{ \mathbf{c}_j^{(0)} \}^k_{j=1}$ su konceptni vektori dobiveni trenutnim particioniranjem.
2. Za svaki dokument vektor $\mathbf{x}_i$, i $1 \leq i \leq n$ potrebno je pronaći konceptni vektor njemu najbliži i izračunati novo particioniranje $\{ \pi_j^{(t+1)} \}^k_{j=1}$

$$\pi_j^{(t+1)} = \{ \mathbf{x} \in \{\mathbf{x}_i\}^n_{i=1} : \mathbf{x}^T \mathbf{c}_j^{(t)} > \mathbf{x}^T \mathbf{c}_l^{(t)}, 1 \leq l \leq n, l \neq j \}, \quad 1 \leq j \leq k$$

Riječima, $\pi_j^{(t+1)}$ je skup svih dokument vektora koji su najbliži konceptnom vektoru $\mathbf{c}_j^{(t)}$. Ovako definirane grupe nazivaju se *Voronoi* ili *Dirichlet*-ove grupe.

[29] Eng. objective function

3. Izračunavaju se novi konceptni vektori prema novom grupiranju $\{ \mathbf{c}_j^{(t+1)} \}_{j=1}^{k}$
4. Ako je ispunjen uvjet zaustavljanja tada je $\pi_j^{+} = \pi_j^{(t+1)}$ i $\mathbf{c}_j^{+} = \mathbf{c}_j^{(t+1)}$ za $1 \leq j \leq k$. Inače t se poveća za 1 i vraćamo se na korak 2.

Jedan od primjera kriterija zaustavljanja je $| Q(\{ \pi_j^{(t)} \}_{j=1}^{k}) - Q(\{ \pi_j^{(t+1)} \}_{j=1}^{k}) | < \varepsilon$ tj. algoritam se zaustavlja ako je promjena objektivne funkcije nakon jedne iteracije manja od praga.

Nadalje vrijednost objektivne funkcije ne smanjuje se iteracijama

$$Q(\{ \pi_j^{(t)} \}_{j=1}^{k}) \leq Q(\{ \pi_j^{(t+1)} \}_{j=1}^{k}) \text{ , za } \forall\, t \geq 0$$

Dokaz:

$$Q(\{ \pi_j^{(t)} \}_{j=1}^{k}) = \sum_{j=1}^{k} \sum_{x \in \pi j(t)} \mathbf{x}^T \mathbf{c}_j^{(t)} = \sum_{j=1}^{k} \sum_{l=1}^{k} \sum_{x \in \pi j(t) \cap \pi l(t+1)} \mathbf{x}^T \mathbf{c}_j^{(t)}$$

$$<= \sum_{j=1}^{k} \sum_{l=1}^{k} \sum_{x \in \pi j(t) \cap \pi l(t+1)} \mathbf{x}^T \mathbf{c}_l^{(t)}$$

$$= \sum_{l=1}^{k} \sum_{j=1}^{k} \sum_{x \in \pi j(t) \cap \pi l(t+1)} \mathbf{x}^T \mathbf{c}_l^{(t)}$$

$$= \sum_{l=1}^{k} \sum_{x \in \pi l(t+1)} \mathbf{x}^T \mathbf{c}_l^{(t)} \leq \sum_{l=1}^{k} \sum_{x \in \pi l(t+1)} \mathbf{x}^T \mathbf{c}_l^{(t+1)} = Q(\{ \pi_j^{(t+1)} \}_{j=1}^{k})$$

Također postoji $lim_{\, t \to \infty} Q(\{ \pi_j^{(t)} \}_{j=1}^{k})$ i vrijedi

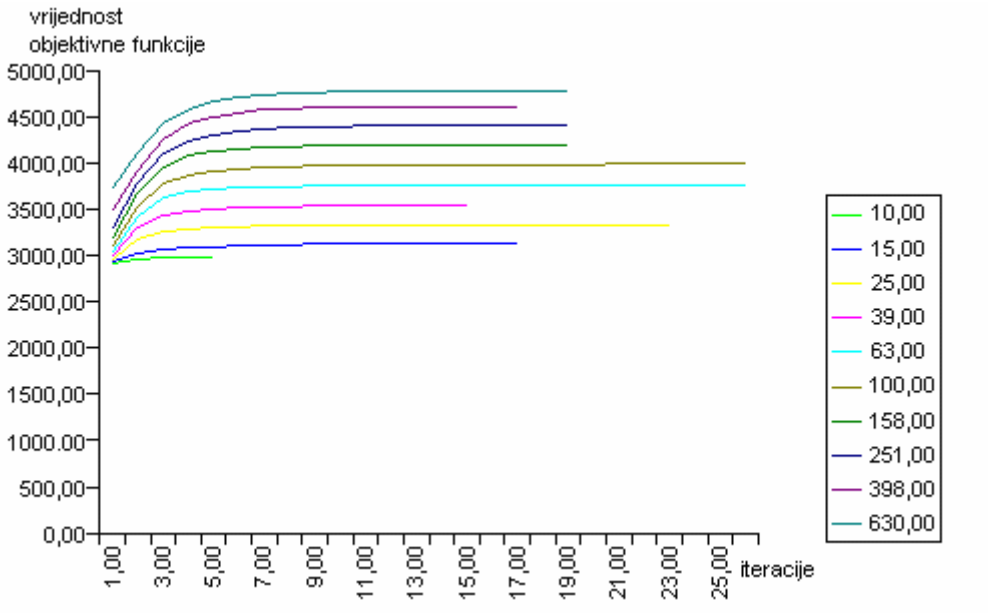
$$Q(\{ \pi_j^{(t)} \}_{j=1}^{k}) = \sum_{j=1}^{k} n_j^{(t)} ||\mathbf{m}_j^{(t)}|| \leq \sum_{j=1}^{k} n_j^{(t)} = n$$

Objektivna funkcija je rastuća i ograničena odozgo konstantom tj. iteracijama konvergira i granica postoji. Ove dokaze može se pronaći u (Dhillon et al., 2005.)

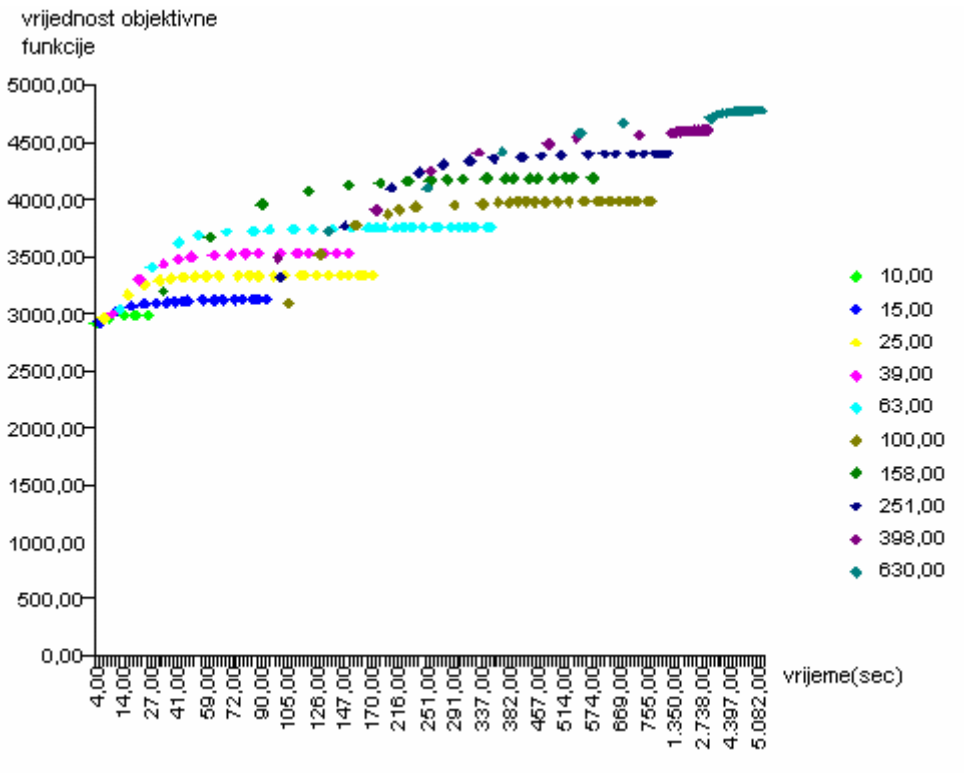
Spherical k-means algoritam kao i ostali gradijentni algoritmi može zaglaviti u lokalnom maksimumu. Ipak u eksperimentima daje dobre rezultate.

4.3 Eksperimentalni rezultati

4.3.1 Objektivna funkcija je rastuća



Slika 21. Promjena objektivne funkcije po iteracijama za različiti k (broj grupa)



Slika 22. Promjena objektivne funkcije u vremenu(sekunde) za različiti k (broj grupa)

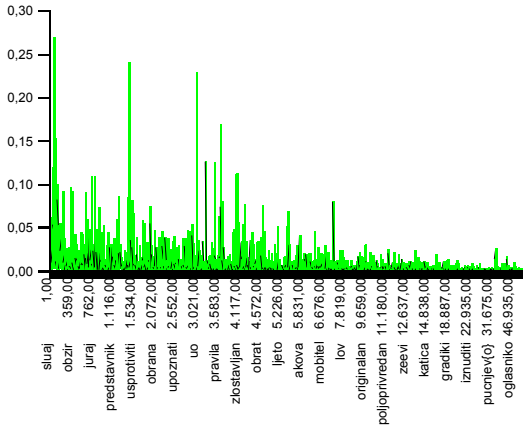
4.3.2 Konceptni vektori su lokalni, rijetki i teže ortonormalnosti

U prethodnom tekstu $\{ \pi_j^{+} \}_{j=1}^{k}$ označava zadnje particioniranje dokumenata u k grupa. Za svaku grupu $1 \le j \le k$ definiramo skup riječi W_j, riječ $1 \le w \le m$ nalazi se u W_j ako je težina te riječi u konceptnom vektoru $\mathbf{c}_j$ veća nego u ostalim konceptnim vektorima ili

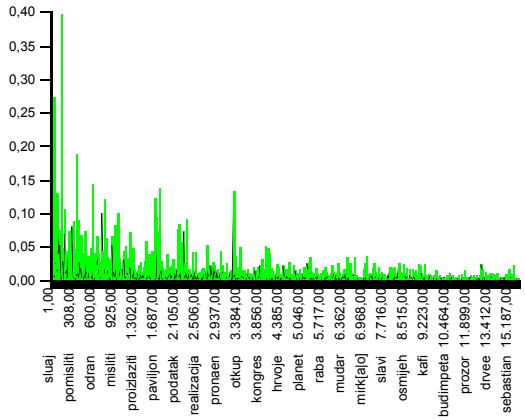
$$W_j = \{ w : 1 \le w \le m , \mathbf{c}_{wj} > \mathbf{c}_{wl}, 1 \le l \le k, l \ne k \}.$$

Skup riječi može se upotrijebiti za opisati grupe dokument vektora. U eksperimentu koji slijedi Vjesnikova baza razdijeljena je u 8 grupa, onoliko koliko ima kategorija različitih vrsta članaka. Svaki skup riječi lokalizira se oko jedne vrste članaka, sporta, kulture, unutrašnje politike, crne kronike, gospodarstva, teme dana, vanjske politike, Zagreba. U tablici 1. prikazani su konceptni vektori za svih 8 kategorija i prvih 15 riječi kako se pojavljuju u konceptnom vektoru. Različitim bojama obilježene su skupine riječi (crvenom je obilježena riječ pripadajuće skupine riječi). Primjećujemo da je većina težina konceptnog vektora koncentrirana ili lokalizirana u pridruženom skupu riječi. Sada možemo reći da su konceptni vektori lokalizirani.

Riječ	Vrijednost	Sljedeća vrij
godina	0,27	0,36
sud	0,24	0,24
policija	0,23	0,23
kaznen	0,17	0,17
rei	0,15	0,19
zagreb	0,15	0,26
djelo	0,13	0,13
upanijski	0,12	0,12
svjedok	0,12	0,12
sat	0,12	0,12
optuen	0,11	0,11
ubojstvo	0,11	0,11
dan	0,11	0,13
automobil	0,11	0,11
kuna	0,11	0,19



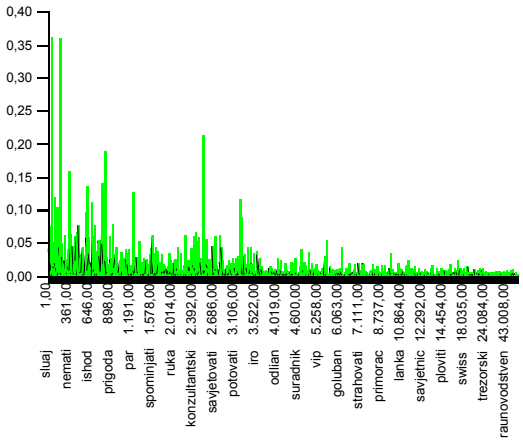
Crna kronika



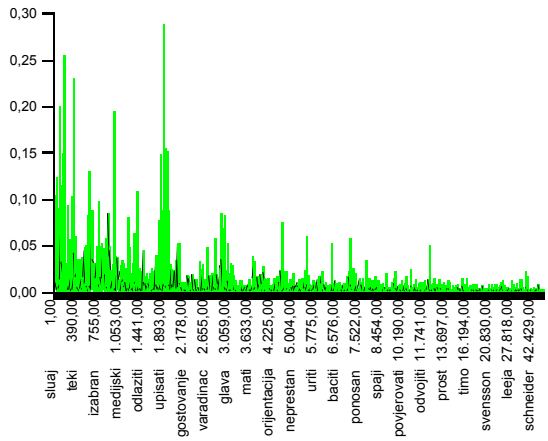
Kultura

Riječ	Vrijednost	Sljedeća vrij
hrvatski	0,40	0,48
godina	0,27	0,36
moi	0,19	0,28
zagreb	0,16	0,26
nov	0,14	0,16
film	0,14	0,14
knjiga	0,13	0,13
imati	0,13	0,20
izloba	0,12	0,12
velik	0,12	0,12
djelo	0,11	0,13
kultura	0,11	0,11
rad	0,10	0,14
ovjek	0,10	0,10
festival	0,09	0,09

Riječ	Vrijednost	Sljedeća vrij
godina	0,36	0,36
hrvatski	0,36	0,48
banka	0,21	0,21
kuna	0,19	0,19
moi	0,16	0,28
trite	0,14	0,14
nov	0,14	0,16
cijena	0,13	0,13
imati	0,12	0,20
dionica	0,12	0,12
tvrtka	0,11	0,11
zagreb	0,11	0,26
trebati	0,10	0,16
poslovan	0,10	0,10
rast	0,09	0,09



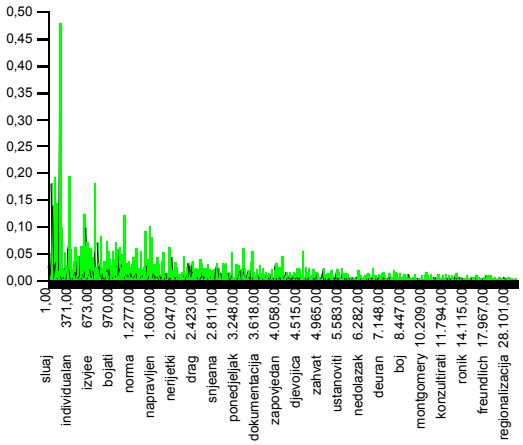
Gospodarstvo



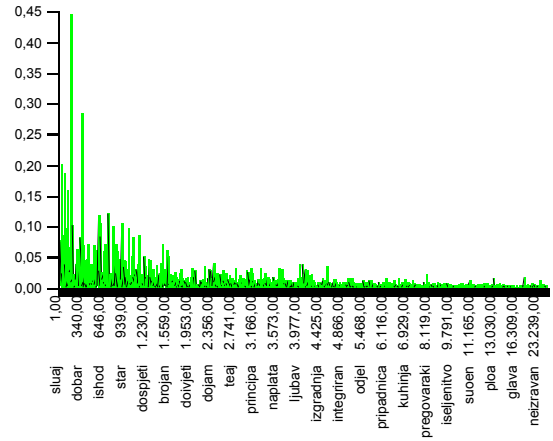
Sport

Riječ	Vrijednost	Sljedeća vrij
igra	0,15	0,15
igra	0,20	0,20
utakmica	0,29	0,29
hrvatski	0,26	0,48
moi	0,23	0,28
imati	0,20	0,20
zagreb	0,17	0,26
igrati	0,16	0,16
klub	0,15	0,15
momad	0,15	0,15
pobjeda	0,13	0,13
godina	0,12	0,36
trener	0,12	0,12
trebati	0,11	0,16
minuta	0,11	0,11
dan	0,10	0,13

Riječ	Vrijednost	Sljedeća vrij
hrvatski	0,48	0,48
moi	0,19	0,28
vlada	0,19	0,19
predsjednik	0,18	0,19
godina	0,18	0,36
trebati	0,14	0,16
rei	0,14	0,19
zagreb	0,14	0,26
imati	0,13	0,20
kazati	0,12	0,12
nov	0,12	0,16
zakon	0,12	0,12
ministar	0,10	0,10
prav	0,10	0,10
stranka	0,10	0,11



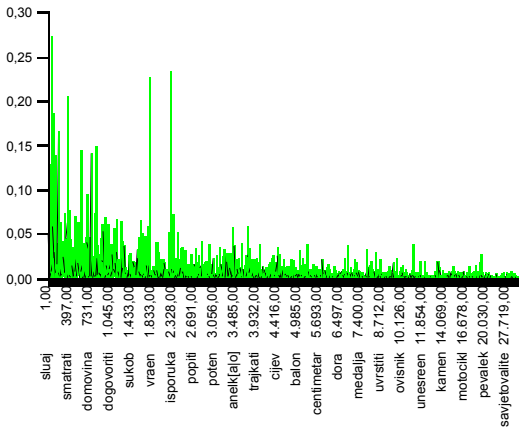
Tema dana



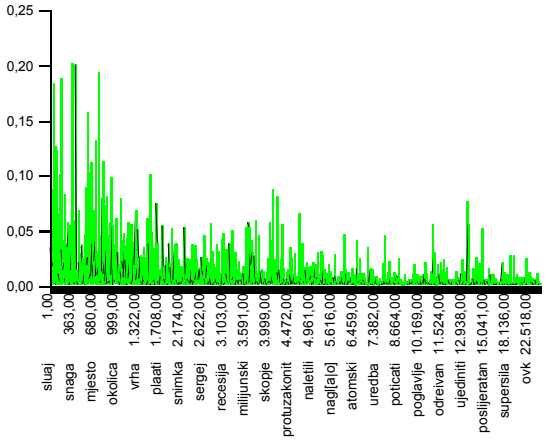
Unutrašnja politika

Riječ	Vrijednost	Sljedeća vrij
hrvatski	0,44	0,48
moi	0,28	0,28
godina	0,20	0,36
imati	0,19	0,20
trebati	0,16	0,16
vlada	0,15	0,19
predsjednik	0,12	0,19
nov	0,12	0,16
politiki	0,11	0,11
moz	0,10	0,10
stranka	0,10	0,11
prav	0,10	0,10
drava	0,10	0,11
morati	0,10	0,10
ovjek	0,10	0,10

Riječ	Vrijednost	Sljedeća vrij
godina	0,27	0,36
zagreb	0,26	0,26
grad	0,23	0,23
gradski	0,23	0,23
moi	0,21	0,28
rei	0,19	0,19
hrvatski	0,17	0,48
zagrebaki	0,16	0,16
kuna	0,15	0,19
nov	0,14	0,16
rad	0,14	0,14
trebati	0,14	0,16
imati	0,13	0,20
dan	0,13	0,13
sat	0,10	0,12



Zagreb

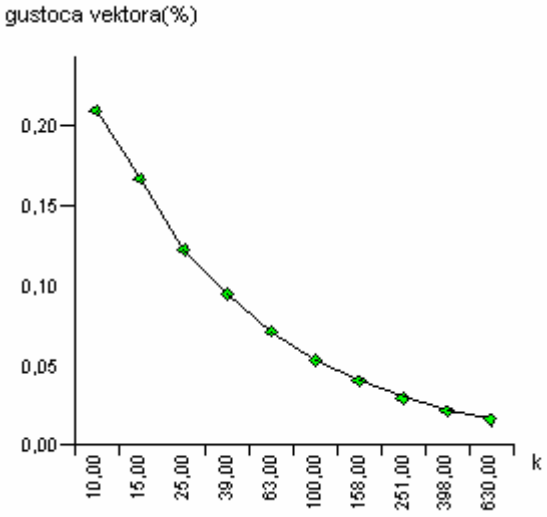


Vanjska politika

Riječ	Vrijednost	Sljedeća vrij
moi	0,20	0,28
ameriki	0,20	0,20
predsjednik	0,19	0,19
hrvatski	0,19	0,48
godina	0,18	0,36
nov	0,16	0,16
zemlja	0,13	0,13
rei	0,13	0,19
imati	0,12	0,20
vlada	0,12	0,19
drava	0,11	0,11
stranka	0,11	0,11
politiki	0,10	0,11
trebati	0,10	0,16
ministar	0,10	0,10

Tablica 1. *Osam konceptnih vektora koji odgovaraju grupiranju Vjesnikove baze u 8 grupa.*

U tablici 1. također možemo primijetiti da su konceptni vektori rijetki, tj. imaju mali broj nenegativnih elemenata. Od 100 000 elemenata, popunjeno je 20 000 odnosno 20%. Kako broj grupa raste tako je manji broj dokument vektora koji se nalaze unutar pojedine grupe i možemo očekivati sve veću rijetkost konceptnih vektora. Dokument vektori su oko 99% rijetki.

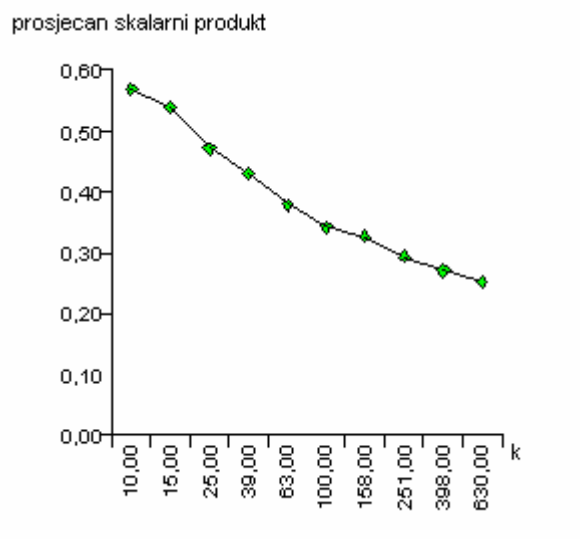


Slika 23. *Povećanjem broja konceptnih vektora oni postaju sve rjeđi*

Za svaki k ≥ 2 možemo izračunati prosječan skalarni produkt konceptnih vektora $\{\mathbf{c}_j^{+}\}_{j=1}^{k}$ kao

$$\frac{2}{k(k-1)} \sum_{j=1}^{k} \sum_{l=j+1}^{k} (\mathbf{c}_j^{+})^{T} \mathbf{c}_l^{+}$$

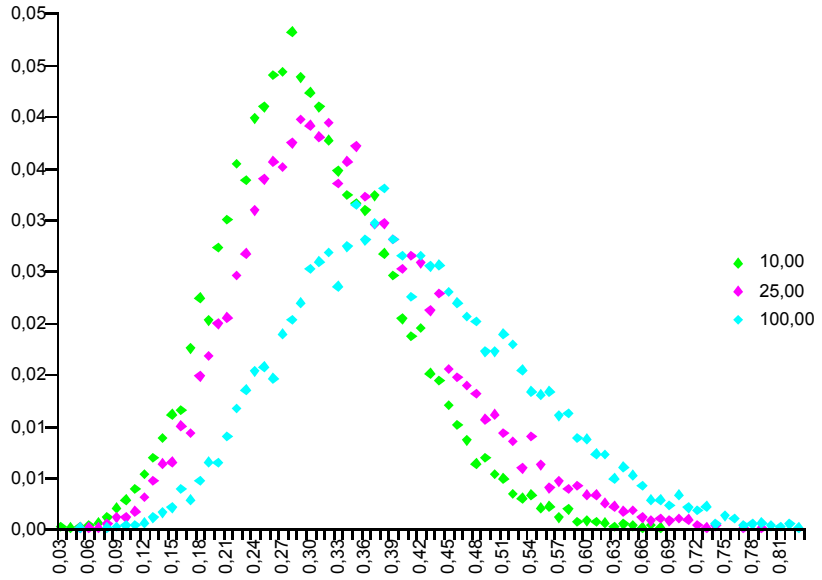
Prosječan skalarni produkt poprima vrijednosti između [0,1] gdje 0 odgovara ortonormalnim vektorima. Kako se broj *k* grupa povećava tako prosječan skalarni produkt teži prema 0 iliti konceptni vektori teže ortonormalnosti.



Slika 24. *Konceptni vektori teže ortonormalnosti*

4.3.3 Struktura unutar i između grupa

Kako smo prije vidjeli svaki dokument vektor je najbliži upravo konceptnom vektoru grupe u kojoj se on nalazi. Strukturu unutar grupe možemo promatrati pomoću funkcije vjerojatnost kosinusnih sličnosti unutar grupe ili skalarnog produkta $\mathbf{x}^T\mathbf{c}_j$ dokument-vektora i pripadajućeg konceptnog vektora.

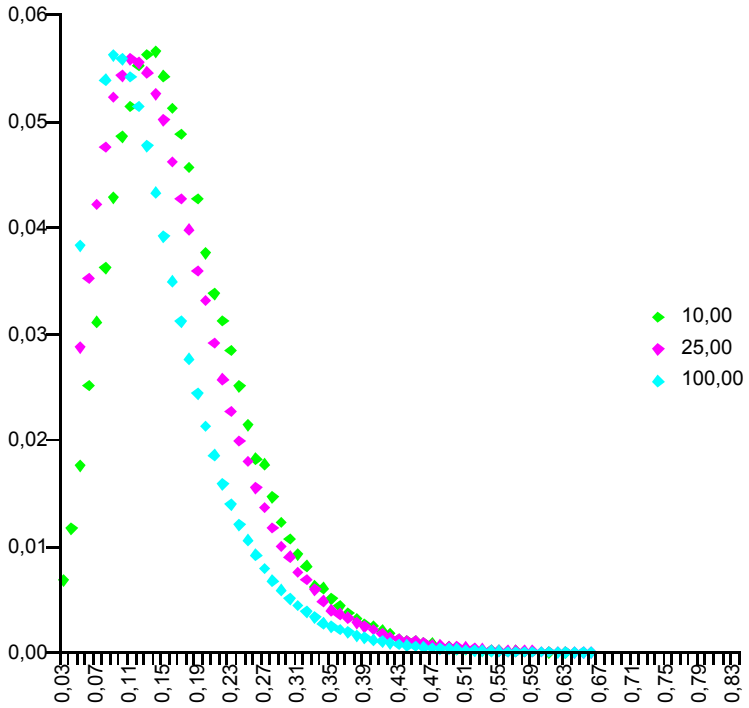


Slika 25. *Funkcija vjerojatnosti* $\mathbf{x}^T\mathbf{c}_j$ unutar grupe

Za *k* je 10 vidimo da se većina vrijednosti nalazi u intervalu [0.1, 0.5] iz čega bi mogli zaključiti da ne postoje vektori bliski svojim konceptnim vektorima. Dakle konceptne vektore ne trebamo zamišljati okružene točkama dokument - vektora, zapravo postoji veliki prazni prostor između dokument - vektora i pridruženih konceptnih vektora. Povećanjem broja konceptnih vektora *k* sve je veća kompaktnost unutar grupe, kao što je bilo za očekivati, dok se očekivanje funkcije vjerojatnosti lagano odmiče prema većim vrijednostima. Dakle, kako se *k* povećava i prazan prostor oko konceptnih vektora se smanjuje.

S druge strane, iako dokument vektori nisu bliski s pripadajućim konceptnim vektorom, još su udaljeniji od ostalih konceptnih vektora. To svojstvo nam zapravo omogućava smisleno grupiranje dokumenata. Strukturu između grupa, ili skalarni produkt $\mathbf{x}^T\mathbf{c}_j$ dokument vektora i ostalih konceptnog vektora možete vidjeti na slici 5. Struktura

između grupa se progresivno se udaljava od unutra grupne strukture i teži prema 0. To navodi na očitu sličnost sa fraktalima (Mandelbrot, 1988).



Slika 26. *Funkcija vjerojatnosti* $\mathbf{x}^T\mathbf{c}_j$ *između grupa*

5 UPORABA RAZLIČITIH METRIKA

Samo srce datamining i information-retrivial zadataka je funkcija udaljenosti kojom mjerimo sličnost među podacima. Možemo reći da većina algoritama kritično ovisi o upotrebi dobre metrike na ulaznom prostoru. Na primjer, K-means, algoritam najbližih susjeda i jezgreni algoritmi kao SVM, trebaju dobru metriku koja reflektira bitne veze među podatcima. Ovaj problem najviše se očituje u nenadziranim algoritmima kao grupiranju i povezan je sa još jednim problemom na tom području: Ako su tri algoritma korištena za grupiranje kolekcije dokumenata, jedna grupa prema autorstvu, druga prema temi, a treća prema stilu pisanja, tko može reći koje je «pravo» rješenje? Još gore, ako imamo samo algoritam koji grupira po temi, a mi želimo po stilu pisanja, relativno je malo mehanizama kojim možemo pravilno grupirati podatke. Pretpostavimo da korisnik ukaže na određene točke u ulaznom prostoru ($\Re^n$) kao «slične». Možemo li naučiti metriku udaljenosti prema ovim vezama, metriku koja dodjeljuje male udaljenosti sličnim parovima? U prethodnom primjeru to bi značilo da iz grupe dokumenata pisanih istim stilom, možemo odrediti bitne značajke u određivanju stila.

Minkowsky metrika

D(x,y) = $\left(\sum_{i=1}^{m} | x_i - y_i |^r \right)^{\frac{1}{r}}$

Euklidska metrika

D(x,y) = $\sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$

Manhattan/city block metrika

D(x,y) = $\sum_{i=1}^{m} | x_i - y_i |$

Mahalnobis metrika

D(x,y) = $[\det V]^{\frac{1}{m}} (x-y)^T V^{-1} (x-y)$

Kvadratna metrika

D(x,y) = $(x-y)^T Q (x-y)$

5.1 Euklidska mjera udaljenosti

Podatci u našem slučaju su vektori u $\Re^m$ prostoru sa elementima koji pretstavljaju broj pojavljivanja *i*-te značajke. Ako su značajke riječi, kao u ovom slučaju, to zovemo «bag of words», ali općenito značajke ne moraju predstavljati riječi te ćemo u daljnjem tekstu

upotrebljavati općenitiji izraz – «značajke». Za mjerenje udaljenosti među vektorima najčešće se upotrebljava Euklidska udaljenost iliti dot produkt[30]

$$d(\mathbf{u}, \mathbf{v})^2 = (\mathbf{u} - \mathbf{v})^T(\mathbf{u} - \mathbf{v})$$

$$= \sum_{j=1}^{n} (\mathbf{u}_i - \mathbf{v}_i)^2$$

koja se najčešće koristi s heurističkim mjerenjem težina značajki[31] kao npr. izračunavanje *tfidf* u preprocesiranju. Postoje dva bitna problema sa ovom udaljenošću:

1. Zanemarene su korelacije među riječima

2. Mjerenje značajki je neizbježno proizvoljno.

Prvi problem je posebno važan kod jezika, jer lingvističke značajke (riječi) imaju snažne međusobne korelacije. Ovakve korelacije ne mogu se razmatrati običnim dot produktom. Ovakav problem moguće je riješiti specifičnim jezgrenim funkcijama.

Drugi problem je jednostavniji. Dok *tfidf* često daje dobre rezultate u praksi postoji više opcija, osobito u *tf* kao logaritmi i kvadratni korijeni, ali ne postoji princip po kojem se one odabiru. Nadalje, ne postoji teoretska osnova koja daje optimalnost u pogledu funkcija udaljenosti.

Gornji problem korelacija među značajkama i težina značajki može se gledati kao problem definiranja prikladne matrice u prostoru značajki, temeljene na distribuciji podataka.

5.2 Mahalanobisova udaljenost

Pretpostavimo da imamo skup točka $X = \{ \mathbf{x} \}_{i=1}^{n} \subseteq \Re^m$ i postoje informacije o sličnosti S: $(\mathbf{x}_i,\mathbf{x}_j) \in S$ ako su $\mathbf{x}_i$ i $\mathbf{x}_j$ slični,

Pitanje je kako naučiti funkciju udaljenosti $d(\mathbf{x},\mathbf{y})$ tako da se poštuje sličnost, tj. da su slične točke blizu?

---

[30] Kada su vektori normalizirani, $|\mathbf{u}| = |\mathbf{v}| = 1$, Euklidska udaljenost je $(\mathbf{u} - \mathbf{v})^T(\mathbf{u} - \mathbf{v}) = |\mathbf{u}|^2 + |\mathbf{v}|^2 - 2\mathbf{u} \cdot \mathbf{v} \propto -\mathbf{u} \cdot \mathbf{v} = -\cos(\mathbf{u}, \mathbf{v})$; (Manning and Schutze, 1999)

[31] feature weighting

Proučimo funkciju udaljenosti u obliku:

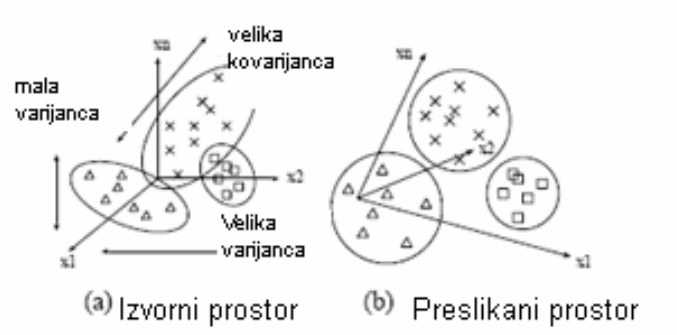
$$d(\mathbf{x}, \mathbf{y}) = d_A(\mathbf{x}, \mathbf{y}) = \| \mathbf{x} - \mathbf{y} \|_A = \sqrt{(x-y)^T A(x-y)}$$

Da bi gornja funkcija bila metrika – i zadovoljavala nenegativnost i nejednakost trokuta – potrebno je da A bude pozitivna, semi-definitna. Postavljajući **A** = **I** daje Euklidsku udaljenost; ako ograničenim A na dijagonalnu matricu, učimo metriku u kojoj različite osi imaju različite «težine»; općenito A parametrizira obitelj Mahalanobisovih udaljenosti na $\Re^n$ [32].

Ako zapišemo gornju jednadžbu u drukčijem obliku (matrica A simetrična)

$$d_A(\mathbf{x}, \mathbf{y})^2 = (\mathbf{A}^{1/2}(\mathbf{u} - \mathbf{v}))^T(\mathbf{A}^{1/2}(\mathbf{u} - \mathbf{v}))$$

dolazimo do zaključka da je učenje takve metrike jednako preslikavanju podataka, koje svaki vektor **x** preslika u $\mathbf{A}^{-1/2}\mathbf{x}$ i zatim se primjenjuje Euklidska metrika u $\mathbf{A}^{-1/2}$-preslikanom prostoru. Možemo zamisliti da su podaci u grupama distribuirani kako je prikazano na slici 1(a), u elipsoidnom obliku koji ima velike (ko)varijance za neke, a male (ko)varijance za druge dimenzije. Nadalje, grupa nije ni poravnana sa osima koordinatnog sustava. Kad pronađemo matricu **A** koja minimalizira šumove u grupi, koja smanjuje velike varijance, a povećava male stvarajući od elipsoidnog oblika kružni u novom $\mathbf{A}^{-1/2}$-preslikanom prostoru (slika 1(b)), možemo očekivati da smo dobili novu, pouzdaniju metriku.



Slika 27. Geometrija prostora značajki

[32] Ako se koristi funkcija preslikavanja ϕ, a time funkcija udaljenosti postaje $\sqrt{(\phi(x)-\phi(y))^T A(\phi(x)-\phi(y))}$, može se naučiti i nelinearna metrika

5.3 Učenje funkcije udaljenosti

Učenje funkcije udaljenosti možemo grubo podijeliti na učenje metrike i jezgreno učenje.

5.3.1 Učenje metrike

Učenje metrike pokušava pronaći optimalnu linearnu transformaciju grupe podatkovnih vektora koja bolje karakterizira sličnost među vektorima. Transformacija je sama linearna, ali podatkovni vektori mogu se prije preslikati koristeći nelinearnu funkciju $\phi(x)$. Ova transformacija jednaka je dodjeljivanju težina značajkama vektora pa se učenje metrike naziva i «feature weighting». Za skup podatkovnih vektora $X = \{ \mathbf{x} \}_{i=1}^{n}$ u $\Re^m$, učenje metrike nastoji naučiti funkciju udaljenosti $d_A(\mathbf{x}_i,\mathbf{x}_j)$ između vektora $\mathbf{x}_i$ i $\mathbf{x}_j$. Matematički funkciju udaljenosti možemo izraziti kao

$$d_A(\mathbf{x}, \mathbf{y}) = \sqrt{(\phi(x) - \phi(y))^T A(\phi(x) - \phi(y))}$$

gdje je **A** pozitivna (semi-)definitna matrica koja zadovoljava svojstva metrike – nenegativnost i nejednakost trokuta. Općenito, **A** je parametar Mahalanobisovih udaljenosti u $\Re^m$. Izbor funkcije preslikavanja ϕ i matrice skaliranja **A** je upravo to što razlikuje algoritme za učenje metrike udaljenosti.

Mnogi algoritmi temelje se na korištenju kontekstualnih informacija ili «side informations». Kontekstualne informacije mogu biti informacije dobivene od korisnika o sličnim karakteristikama podskupa podataka. Na temelju takvih informacija, (Bar-hillel et al., 2003.) koriste Relevant Component Analysis (RCA) u izračunavanju Mahalanobisove metrike. Autori koriste relaciju ekvivalentnosti za kontekstualnu informaciju. Računaju

$$\mathbf{C} = \frac{1}{P}\sum_{j=1}^{|G|}\sum_{i=1}^{|Sj|} (\mathbf{x_{ji}} - \mathbf{m_j})(\mathbf{x_{ji}} - \mathbf{m_j})^T$$

gdje je $\mathbf{m}_j$ centroid *j*-te grupe, dok $|\mathbf{G}|$ i $|\mathbf{S}_j|$ pretstavljaju broj grupa i broj uzoraka u *j*-toj grupi. Matrica $\mathbf{W} = \mathbf{C}^{-1/2}$ koristi se za transformaciju, a inverz matrice **C** kao Mahalanobisova matrica.

(Aggarwal, 2003.) razmatra funkcije udaljenosti osjetljive na pojedine karakteristike podataka. Korišteni modeli su parametrizirani Minkowski model:

$$D(\mathbf{x_i}, \mathbf{x_j}, \lambda) = \left(\sum_{r=1}^{m} \lambda_r |x_{ir} - x_{jr}|^P \right)^{1/P}$$

i parametrizirani kosinusni model

$$\cos(\mathbf{x_i}, \mathbf{x_j}, \lambda) = \sum_{r=1}^{m} \frac{\lambda_r x_{ir} x_{jr}}{\sqrt{\sum_{r=1}^{m} \lambda_r^2 x_{ir}^2 \sum_{r=1}^{m} \lambda_r^2 x_{jr}^2}}$$

Oba modela nastoje minimizirati pogrešku s obzirom na λ_r. Parametrizirani Minkowski model može se promatrati kao dodjeljivanje težina značajkama na ulaznom prostoru. Slično, i parametrizirani kosinusni model možemo promatrati kao skalarni produkt na ulaznom prostoru.

(Xing et al., 2003) problem u (Bar-hillel et al., 2003.) promatraju kao problem konveksne optimizacije i razrađuju tehnike za dobivanje dijagonalne i pune Mahalanobisove matrice. Za razliku od RCA ovdje se koriste iterativne metode osjetljive na postavke parametara i također računski zahtjevne.

Jednostavan način definiranja kriterija za željenu metriku je već spomenuti zahtjev da parovi točaka $(\mathbf{x_i},\mathbf{x_j}) \in S$ budu blizu tj. želimo minimizirati $\sum_{(xi,xj)\in S} \| \mathbf{x_i} - \mathbf{x_j} \|^2_A$. Trivijalno rješenje problema je za $\mathbf{A} = 0$, dodajući uvjet $\sum_{(xi,xj)\in D} \| \mathbf{x_i} - \mathbf{x_j} \|^2_A \geq 1$ osiguravamo da S ne preslika sve točke u jednu. Ovdje D označava skup parova točaka za koje se ili zna da «nisu slične» ako je ta informacija eksplicitno dostupna ili da se uzimaju svi parovi točaka koji se ne nalaze u S. Dolazimo do optimizacijskog problema

$$\min_A \sum_{(xi,xj)\in S} \| \mathbf{x_i} - \mathbf{x_j} \|^2_A$$

$$\text{st.} \quad \sum_{(xi,xj)\in D} \| \mathbf{x_i} - \mathbf{x_j} \|^2_A \geq 1$$
$$A \geq 0$$

Izbor konstante 1 na lijevoj strani nejednadžbe je proizvoljan, ali i nebitan; ako konstantu zamijenimo bilo kojom pozitivnom konstantom to rezultira samo mijenjanjem matrice **A** u $c^2\mathbf{A}$.

S jedne strane (Xing et al., 2003.) gleda na ovaj optimizacijski problem kao konveksni. U slučaju da želimo izračunati dijagonalnu matricu $\mathbf{A} = \text{diag}(A_{11}, A_{22}, ..., A_{mm},)$ dolazimo do učinkovitog algoritma koji koristi Newton-Raphsonovu metodu. Definirajmo

$$g(A) = g(A_{11},\ldots, A_{nn}) = \sum_{(xi,xj)\in S} \| \mathbf{x_i} - \mathbf{x_j} \|^2_A - \log\left(\sum_{(xi,xj)\in D} \| \mathbf{x_i} - \mathbf{x_j} \|^2_A\right)$$

i slijedi da minimaliziranje funkcije g(uz $A \succeq 0$ [33]) je ekvivalentno rješavanju gornjeg problema. Dakle, možemo koristiti Newton-Raphsonovu metodu za optimiziranje funkcije g.

Ako tražimo punu matrica **A**, Newton-Rapsonova metoda postaje presložena (složenost je $O(n^6)$). Polazimo od algoritma gradijentnog spusta i iterativnih projekcija

Iterate
  Iterate
    $A := \arg\min_{A'}\{||A' - A||_F : A' \in C_1\}$
    $A := \arg\min_{A'}\{||A' - A||_F : A' \in C_2\}$
  until A converges
  $A := A + \alpha(\nabla_A g(A))_{\perp \nabla_A f}$
until convergence

gdje je $||\ .\ ||_F$ već ranije spominjana Frobeniusova norma matrice $||\mathbf{A}||_F = \sqrt{\sum_{i=1}^{p}\sum_{j=1}^{q}|aij|}$.

Postavljamo sljedeći problem

$$\max_A \quad g(A) = \sum_{(xi,xj)\in D} ||\ \mathbf{x_i} - \mathbf{x_j}\ ||^2_A$$

$$\text{st.} \quad f(A) = \sum_{(xi,xj)\in S} ||\ \mathbf{x_i} - \mathbf{x_j}\ ||^2_A \leq 1$$
$$A \geq 0$$

Koristi se gradijentni silazak za rješavanje funkcije g iza kojeg slijede iterativne projekcije koje osiguravaju da vrijede oba dva postavljena uvjeta tj. nakon gradijentnog koraka $\mathbf{A} = \mathbf{A} + \alpha\nabla_A g(\mathbf{A})$, projicira se matrica **A** u $C_1 = \{\mathbf{A}: \sum_{(xi,xj)\in S} ||\ \mathbf{x_i} - \mathbf{x_j}\ ||^2_A \leq 1\}$ i C_2 = $\{\mathbf{A}: \mathbf{A} \geq 0\}$ kako je prikazano u algoritmu. Razlog ovakvog pristupa je računski nezahtjevno projiciranje matrice **A** u C_1 i C_2. Prvi uvjet $\arg\min_{A'} \{||\ \mathbf{A'} - \mathbf{A}\ ||_F : \mathbf{A'} \in C_1\}$ lako je riješiti skupom rijetkih linearnih jednadžbi i složenosti je $O(n^2)$. Sljedeći korak projekcije najprije rastavlja matricu **A** na $\mathbf{A} = \mathbf{X}^T\Lambda\mathbf{X}$, gdje je $\Lambda = \text{diag}(\lambda_1, \lambda_{2,\ldots,} \lambda_n)$ dijagonalna matrica svojstvenih vrijednosti matrica **A** koju zamjenjujemo sa $\Lambda' = \text{diag}(\max\{0,\lambda_1\}, \max\{0,\lambda_2\}_{,\ldots,} \max\{0,\lambda_n\})$, a matricu **A** sa $\mathbf{A} = \mathbf{X}^T\Lambda'\mathbf{X}$.

(Mochihasi,2004.) kako sam i kaže, dijeli isti cilj i nasljeđuje djelo (Xing et al.,2003.). Za razliku od parova sličnih točaka i minimaliziranja udaljenosti među njima, (Mochihashi,2004.) traži matricu **M** koja minimalizira udaljenost između pojedinih

[33] Da ovaj uvjet bude ispunjen, potrebno je modificirati Newtonov pomak sa $H^{-1}\nabla g$ sa $\alpha\ H^{-1}\nabla g$, gdje je α parametar koraka koji daje najviši spust uz uvjet da je $A_{ii} \geq 0$

podatkovnih vektora i pridruženom centroida grupe kojoj pripada. Matematički napisan ovaj kvadratni problem uz $\mathbf{s}_i$ koji označava *i*-ti podatkovni vektor i $\mathbf{c}_i$ centroid *i*-te grupe od njih N, je

$$M = \arg\min_M \sum_{i=1}^{N} \sum_{sj \in Xi} d_M (\mathbf{s_j}, \mathbf{c_i})^2$$

$$= \arg\min_M \sum_{i=1}^{N} \sum_{sj \in Xi} (\mathbf{s_j} - \mathbf{c_i})^T \mathbf{M} (\mathbf{s_j} - \mathbf{c_i})$$

uz uvjet da je $|\mathbf{M}| = 1$. Uvjet, kako je prije navedeno, isključuje mogućnost da je $\mathbf{M} = 0$, a 1 je proizvoljna konstanta. Promjenom te konstante sa c dobivamo novo rješenje $c^2\mathbf{M}$.

Matrica koja rješava problem minimizacije definiran prije je

$$\mathbf{M} = |\mathbf{A}|^{1/n} \mathbf{A}^{-1}$$

gdje je $\mathbf{A} = [a_{kl}]$

$$a_{kl} = \sum_{i=1}^{N} \sum_{sj \in Xi} (\mathbf{s_{jl}} - \mathbf{c_{il}}) (\mathbf{s_{jk}} - \mathbf{c_{ik}})$$

Kad je **A** singularna, možemo umjesto $\mathbf{A}^{-1}$ koristiti njen Moore-Penrose pseudoinverz $\mathbf{A}^{+}$[34]. Općenito se **A** sastoji od jezičnih značajki, vrlo je rijetka i često singularna. Dakle, $\mathbf{A}^{+}$ je često potrebna u računanju.

U problemima grupiranja gdje su značajke lingvističke ovakav način dobivanja funkcije udaljenosti je računski nemoguć. U matrica M imala bi *broj riječi* x *broj riječi* elemenata. Ipak, u kombinaciji sa metodama redukcije dimenzionalnosti, npr. SVD-om, ovakva metrika bila bi upotrebljiva.

Općenito, učenje metrike nastoji naučiti dobru funkciju udaljenosti pomoću težina značajki na ulaznom prostoru. Iako se značajke može preslikati pomoću nelinearne funkcije , takva transformacija je eksplicitna i zbog računske složenosti ovakav pristup je nepraktičan. Jezgrena metoda učenja funkcije udaljenosti uspješno zaobilazi problem računske složenosti.

[34] Moore-Penrose pseudoinverz $\mathbf{A}^{+}$ jedinstvena je matrica koja ima svojstvo da vrijedi $x = \mathbf{A}^{+}y$ kao rješenje najmanjih kvadrata za $\mathbf{A}x = y$ i ako **A** je singularna.

5.3.2 Jezgreno učenje

Jezgrene metode implicitno preslikavaju skup podatkovnih vektora $X = \{ \mathbf{x} \}_{i=1}^{n}$ na ulaznom prostoru *I* u drugi, više dimenzionalni (moguće i beskonačni) projicirani prostor P pomoću funkcije preslikavanja ϕ. Jezgrena funkcija *k* definirana je kao skalarni produkt između dva vektora u projiciranom prostoru P, kao

$$k(\mathbf{x_i}, \mathbf{x_j}) = < \phi(\mathbf{x_i}), \phi(\mathbf{x_j}) >.$$

Jezgrene metode koriste skalarni produkt (**K**) kao mjeru sličnosti. Potrebno je odabrati i pravilni **K** koji mora biti pozitivan, (semi-) definitivan i simetričan. Koriste se polinomne jezgrene funkcije, Gaussov RBF, Laplace RBF. U P, udaljenost se može izračunati pomoću jezgrenog trika

$$d(\mathbf{x_i}, \mathbf{x_j}) = \sqrt{K(x_i, x_i) + K(x_j, x_j) - 2K(x_i, x_j)}$$

gdje ispravnost funkcije udaljenosti slijedi iz ispravnosti jezgrene matrice **K**. Zbog velike uloge jezgrene funkcije, loš odabir vodi lošim performansama ove, jezgrene metode. Umjesto da se koristi predefinirani **K** mnogi se istražuje i učenje **K** na skupu podataka za učenje.

(Cristianini et al., 2002) predložili su «kernel alignment» za mjerenje sličnosti između dvije jezgrene funkcije. Geometrijski, «alignment» je definiran kao kosinusni kut između dvije jezgrene matrice, nakon transformiranja te dvije matrice u vektore. Vezano uz to, predložili su i idealiziranu jezgrenu matricu (**K***). Pretpostavimo da je $y(\mathbf{x}_i) \in \{-1,1\}$ je oznaka klase za $\mathbf{x}_i$. **K*** je definiran kao

$$\mathbf{K}^*(\mathbf{x_i}, \mathbf{x_j}) = 1, \text{ ako } y(\mathbf{x_i}) = y(\mathbf{x_j})$$

$$0, \text{ ako } y(\mathbf{x_i}) \neq y(\mathbf{x_j})$$

a pretstavlja ciljnu jezgrenu matricu kojoj početna treba težiti. Učenje jezgrene matrice proučavali su (Kwok et al., 2003) i (Zhang, 2003.) te (Wu et al., 2004). Daljnje ulaženje u ovu problematiku izvan je teme ovoga rada.

5.4 Jezgreni k-means algoritam

Kao i kod spherical k-mean algoritma potrebno je definirati objektivnu funkciju koju ćemo u ovom slučaju minimalizirati. Za ulazni skup dokument-vektora $\mathbf{x_1}$, $\mathbf{x_{2,...,}}$ $\mathbf{x_n}$, centroida grupa $\mathbf{c_1}$, $\mathbf{c_{2,..,}}$ $\mathbf{c_k}$, i funkciju *f* koja svakom dokument-vektoru pridružuje jednu grupu nastojimo minimalizirati

$$\sum_{j=1}^{k} || \phi(\mathbf{x_i}) - \mathbf{c}_{f(xi)} ||^2.$$

Početni centroidi su slučajno odabrani, da bi se gornja funkcija minimalizirala mijenjajući funkciju *f* kao i centroide grupa. Postupak je iterativan. Najprije se točke pomiču u grupu čijem su centroidu najbliže. Jasno je da će to minimalizirati gornju funkciju. Zatim se u sljedećem koraku mijenja centroid grupe sa centrom mase svih točaka koje se nalaze unutar te grupe. Također se može dokazati da i ovaj korak minimalizira gornju funkciju. Nakon konačnog broja iteracija algoritam će konvergirati.

Definirajmo matricu **A** dimenzija ℓ *x N* kao

A_{ik} = 1, ako je $\mathbf{x_i}$ nalazi u grupi *k*

0, inače

U svakom retku matrice A mora postojati točno jedan element, dok suma svakog retka daje broj dokument-vektora u jednoj grupi. Koordinate centroida možemo izračunati

X' A D

Gdje je **X** matrica dokument-vektora u redcima, a **D** je *N X N* dijagonalna matrica sa elementima koji su inverz stupčaste sume matrice **A** (broj elemenata dodijeljenih pojedinoj grupi). Udaljenost vektora $\phi(\mathbf{x})$ od centroida dana je sa

$$|| \phi(\mathbf{x}) - \mathbf{c}_k ||^2 = ||\phi(\mathbf{x})||^2 - 2 < \phi(\mathbf{x}) , \mathbf{c}_k > + || \mathbf{c}_k ||^2$$

$$= k(\mathbf{x}, \mathbf{x}) - 2 (\mathbf{k'AD})_k + (\mathbf{DA'XX'AD})_{kk}$$

gdje je **k** vektor skalarnih produkata između vektora $\phi(\mathbf{x})$ i vektora za učenje.

Dakle grupa kojoj se vektor $\phi(\mathbf{x})$ dodaje definirana je sa

$$\text{argmin}_{1 \le k \le N} || \phi(\mathbf{x}) - \mathbf{c}_k ||^2 = \text{argmin}_{1 \le k \le N} (\mathbf{DA'KAD})_{kk} - 2 (\mathbf{k'AD})_k$$

gdje je **K** jezgrena matrica.

U svakoj se iteraciji na ovaj način računa grupa kojoj je dodijeljen pojedini vektor. Kao i u većini jezgrenih algoritama, tako i u ovome, algoritam dobiva informacije o ulaznim podacima i o prostoru u koji se preslikavaju preko jezgrene matrice **K**. Možemo reći da jezgrena matrica ima centralnu ulogu u jezgrenim algoritmima i dakle njen odabir jako je važan (Shawe-Taylor et al., 2004.).

5.5 Odabir jezgrene funkcije

Jezgrena matrica sadrži sve bitne informacije o položaju ulaznih podataka u prostoru značajki. Razmotrit ćemo dva načina pogrešnog odabira jezgrene matrice. Ako je jezgrena funkcija preopćenita i ne daje važnost određenim sličnostima među ulaznim podacima. Jezgrena funkcija daje iste težine svakom ulaznom paru, time nedijagonalni elementi jezgrene matrice postaju mali, a dijagonalni teže 1. Jezgrena matrica svedena je na koncept identiteta. Dakle, došli smo do pretreniranosti jer možemo točno klasificirati skup za učenje, dok jezgrena matrica nema svojstvo generalizacije na novim podacima. S druge strane, ako je matrica potpuno uniformna, tada je svaki ulaz sličan svakom drugom ulazu. Ovaj problem preslikava svaki ulaz u isti vektor u prostoru značajki i vodi podtreniranosti podataka. Geometrijski, prvi problem odgovara preslikavanju ulaza u ortogonalne vektore u prostoru značajki, dok su u drugom problemu sve točke preslikane u istu sliku.

Idealan odabir jezgrene matrice temeljio bi se na našem a priori znanju problema i pojednostavnio bi učenje na odabir funkcije uzorka u prostoru značajki. Nažalost, nije uvijek moguće pravilno odabrati jednu jezgrenu matricu a priori, već moramo odabrati obitelj jezgrenih matrica koje reflektiraju naše a priori očekivanje i ostavljamo si na izbor odabir jezgrene matrice koju ćemo koristiti. Dakle, sustav koji uči mora sad riješiti dva problema: odabrati jezgrenu matricu iz obitelji jezgrenih matrica i poslije ili konkurentno odabrati funkciju uzorka u prostoru značajki.

Mnogi pristupi se mogu koristiti u rješavanju dvojnog problema učenja. Najjednostavniji koriste malo podataka iz skupa za učenje, i često ne uzimaju o obzir oznake kod učenja s učiteljem. Razrađenije metode koje koriste oznake trebaju mjeru «dobrote» u rješavanju jezgrenog dijela problema. Dakle, uvodi se pojam sličnosti među jezgrenim matricama i odabire se jezgena matrica koja koja je najbliža idealnoj jezgrenoj matrici

$$\mathbf{K}^*(\mathbf{x}_i, \mathbf{x}_j) = 1, \text{ ako } y(\mathbf{x}_i) = y(\mathbf{x}_j)$$

$$0, \text{ ako } y(\mathbf{x}_i) \neq y(\mathbf{x}_j).$$

Mjera sličnosti među jezgrama, ili u slučaju idealne jezgrene matrice između jezgrene matrice i cilja, mora zadovoljavati osnovna svojstva: mora biti simetrična, maksimizirati se kad su argumenti jednaki i minimalizirati se kad se primjeni na dvije nezavisne jezgrene matrice. Nadalje, u slučaju idealne jezgrene matrice ograničit ćemo se samo na skup za učenje jer se idealna jezgrena matrica može izračunati samo na tom

skupu. Dakle, potrebno je dokazati da se procjena može dobiti koristeći samo manji skup podataka.

(Christianini et al.,2001.) uvode pojam jezgrenog alignmenta, mjere sličnosti između dvije jezgrene funkcije ili jezgrene funkcije i ciljne funkcije.

Alignment jezgrene funkcije k_1 i jezgrene funkcije k_2 na skupu S = {$\mathbf{x_1}$, $\mathbf{x_2}$,..., $\mathbf{x_n}$} definiran je kao

$$A(S,k_1,k_2) = \frac{< K_1, K_2 >_F}{\sqrt{< K_1, K_1 >_F < K_2, K_2 >_F}}$$

gdje su K_i jezgrene matrice dobivene primjenom funkcije k_i na skupu S. Alignment možemo promatrati i kao kosinus kuta između dva bi-dimenzionalna vektora K_1 i K_2. Ako je K_2 idealni kernel definiran sa K_2 = yy', gdje je y vektor oznaka {-1,1} na skupu S, tada

$$A(S,K,yy') = \frac{< K, yy' >_F}{\sqrt{< K, K >_F < yy', yy' >_F}} = \frac{< K, yy' >_F}{m\sqrt{< K, K >_F}}$$ jer vrijedi

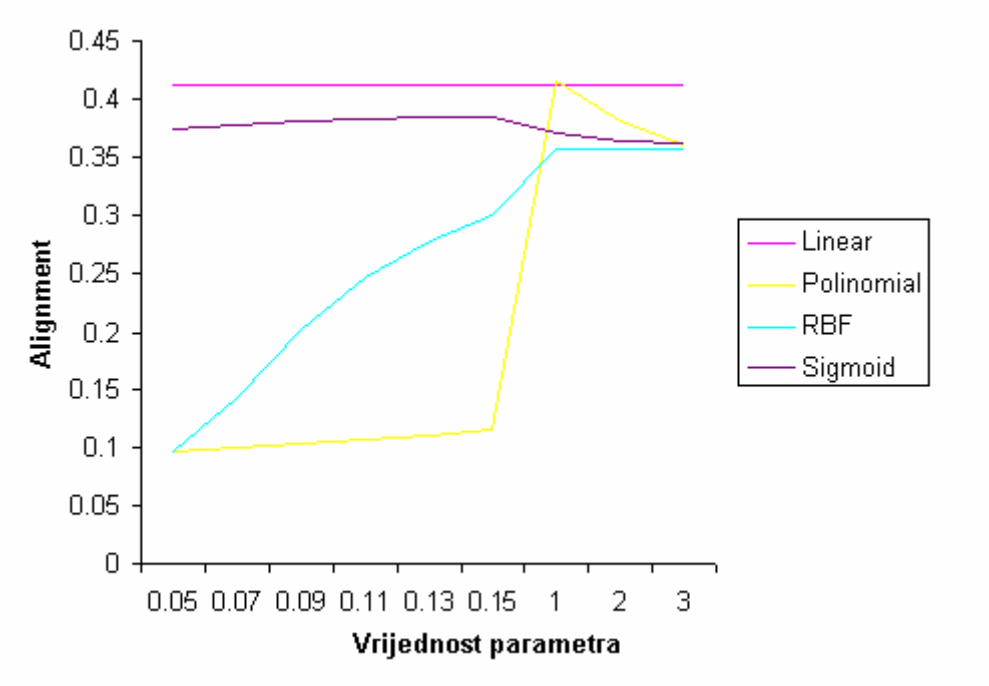
$$< yy', yy' >_F = m^2$$

Dakle, mjeru sličnosti između dvije jezgre transformirali smo u mjeru dobrote pojedine jezgrene funkcije.

(Christianini et al. ,2001.) pokazali su da ako je očekivanje alignmenta visoko, onda postoji funkcija koja dobro generalizira. Dakle, možemo optimirati alignment na skupu za učenje i očekivati da će na testnom skupu jezgrena funkcija također davati visoki alignment i imati dobre performanse.

Najprije ćemo odabrati obitelj(linearna, RBF,…) jezgrenih funkcija. Zatim je potrebno odabrati parametre tih funkcija koji će davati najbolje rezultate na Vjesnikovoj bazi.

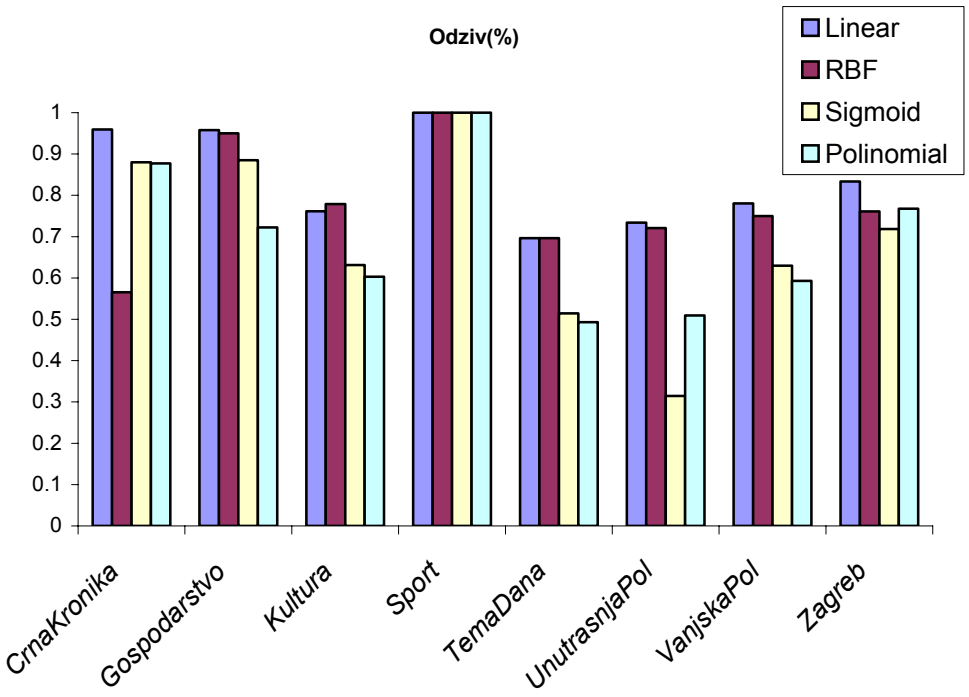
Linearna	$F(\mathbf{x_i},\mathbf{x_j}) = (\mathbf{x_i}\ \mathbf{x_j})$
RBF	$F(\mathbf{x_i},\mathbf{x_j}) = \exp(-r\|\|\mathbf{x_i} - \mathbf{x_j}\|\|^2)$
Sigmoidalna	$F(\mathbf{x_i},\mathbf{x_j}) = \tanh(a^* \mathbf{x_i}\mathbf{x_j})$
Polinomialna	$F(\mathbf{x_i},\mathbf{x_j}) = (\mathbf{x_i}\ \mathbf{x_j} + 1)^d$



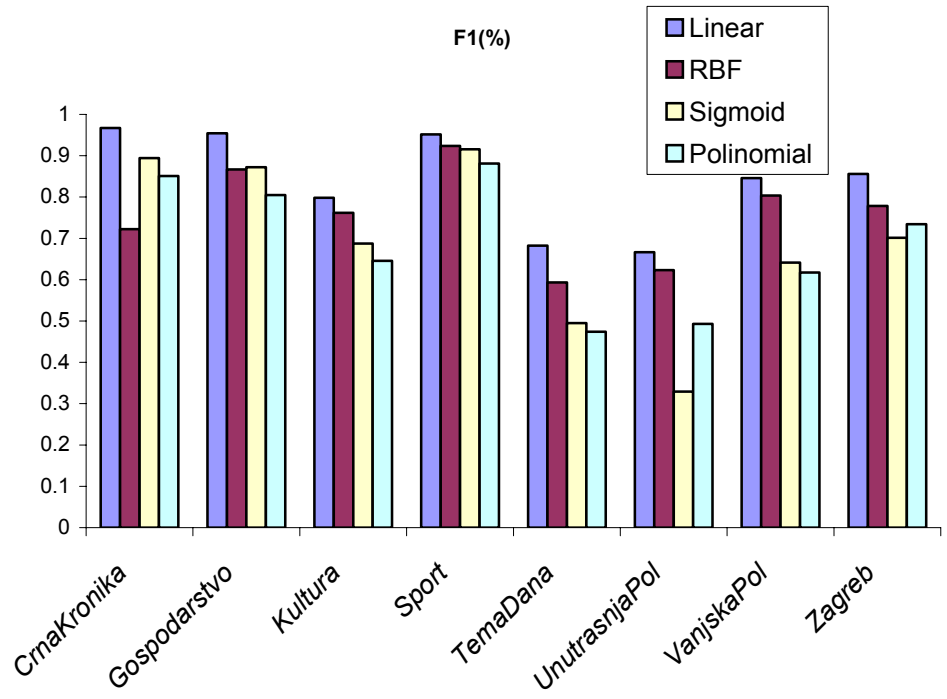
Slika 28. Alignment za različite parametre σ i različite jezgrene funkcije

Prema rezultatima vidimo da linearna funkcija (koja zapravo odgovara kosinusu kuta između vektora) ima najveći alignment sa našim idealnim kernelom dobivenim na skupu za učenje. Ta funkcija nije ovisna o parametrima. Linearnoj funkciji najviše se približava polinomialna za parametar 1, ili $(\mathbf{x}_i\ \mathbf{x}_j + 1)^1$ za koju možemo reći da je skoro identična linearnoj. Ako primijenimo jezgreni k-means na skupu za učenje, možemo vidjeti da točnost grupiranja na tom skupu sa različitim jezgrenim matricama potvrđuje pretpostavke postavljene sa alignmentom.

Isto tako uspješnost jezgrene funkcije kao metrike možemo ocijeniti i na taj način da gledamo $\phi(\mathbf{x}) - \mathbf{c}_k$, ili udaljenost vektora u prostoru značajki od njegovog konceptnog vektora na skupu za učenje. Odziv označava koliki je broj vektora koji pripada dotičnoj kategoriji pravilno klasificiran, ili u našem slučaju najbliži centroidu upravo svoje grupe. Prema rezultatima za sport(kategorija 4) možemo vidjeti da su svi dokumenti u toj kategoriji u svim metrikama i promjenama jezgrene matrice zapravo najbliži uvijek svom centroidu. Druge kategorije mijenjaju se izborom jezgrene funkcije, ali većinom vidimo da linearna jezgrena funkcija daje najbolje rezultate. Ako pogledamo mjeru F1, tj. i preciznosti i odzivu damo istu važnost, vidimo da linearna funkcija nadmašuje sve druge.



Slika 29. Postotak vektora koji su najbliži svom centroidu upotrebom različitih metrika



Slika 30. Mjera F1, ocjena odziva i preciznosti vektora i njihove udaljenosti od pripadnih centroida

6 KONCEPTNO INDEKSIRANJE

Konceptno indeksiranje također koristi grupe dokument vektora za preslikavanje u niže dimenzionalni prostor. Postoje nenadzirana i nadzirana tehnika redukcije dimenzionalnosti.

Konceptno indeksiranje na sljedeći nenadzirani način računa redukciju dimenzionalnosti. Kao i kod konceptne dekompozicije, ako je *k* željeni broj dimenzija, koristi se *k* konceptnih vektora kao osi novog *k*-dimenzionalnog prostora. Uz *n* dokument vektora $\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_n}$ u $\Re^m$ možemo definirati *riječ x dokument* matricu kao

$$\mathbf{X} = [\mathbf{x_1}\ \mathbf{x_2}\ ...\ \mathbf{x_n}] ,$$

i konceptnu matricu *m* x *k* tako da su stupci matrice konceptni vektori

$$\mathbf{C} = [\mathbf{c_1}\ \mathbf{c_2}\ ...\ \mathbf{c_k}] .$$

Svaki konceptni vektor predstavlja osi novog prostora, i *k*-dimenzionalna reprezentacija dokument vektora dobiva se projiciranjem u ovaj prostor. Ta projekcija može se matrično zapisati, ako je $\mathbf{d}_i$ *i*-ti dokument vektor tada **dC** predstavlja *k*-dimenzionalnu reprezentaciju tog vektora, a $\mathbf{X}^T\mathbf{C}$ predstavlja preslikani cijeli skup dokumenata. Slično, upit **q** se preslikava se **qC**. Sličnost između dva dokumenta u reduciranom prostoru računa se kosinusnom sličnosti između reduciranih vektora.

U slučaju nadzirane redukcije dimenzionalnosti, konceptno indeksiranje koristi već postojeće klase dokumenta i unutra njih traži grupe sličnih dokumenta. U najjednostavnijem slučaju svaka grupa pripada jednoj klasi iz skupa dokumenata. U ovom slučaju rang niže dimenzionalnog prostora bit će jednak broju klasa. Općenito tražimo niže dimenzionalan prostor ranga *k* koji je veći od broja klasa *l*. Najprije se grupira u *l* grupa tako da se stvori grupa za svaku klasu dokumenata, a onda se one dalje particioniraju dok ne dobijemo *k* grupa. Bitno je primijetiti da će svaka grupa dokumenata sadržavati dokumente samo jedne klase. Nakon što su grupe određene konceptno indeksiranje ne razlikuje se od opisanog u prethodnom odjeljku – kao u slučaju nenadzirane redukcije dimenzionalnosti.

Za razumijevanje konceptnog indeksiranja mora se kao prvo shvatiti bit konceptnih vektora i drugo značenje reducirane dimenzionalne reprezentacije svakog dokumenata. Kako smo u prethodnom poglavlju opisali konceptni vektor sumarizira kontekst dokument vektora u njegovoj grupi. Također svaka grupa nastoji obuhvatiti čim sličnije vektore, tj. dokument vektori koji se nalaze u različitim grupama manje su slični nego oni u istoj grupi. Za *k* konceptnih vektora i dokument **d** *i*-ta koordinata reduciranog dimenzionalnog prostora je kosinusna sličnost između dokumenta **d** i *i*-tog konceptnog vektora. Dimenzije

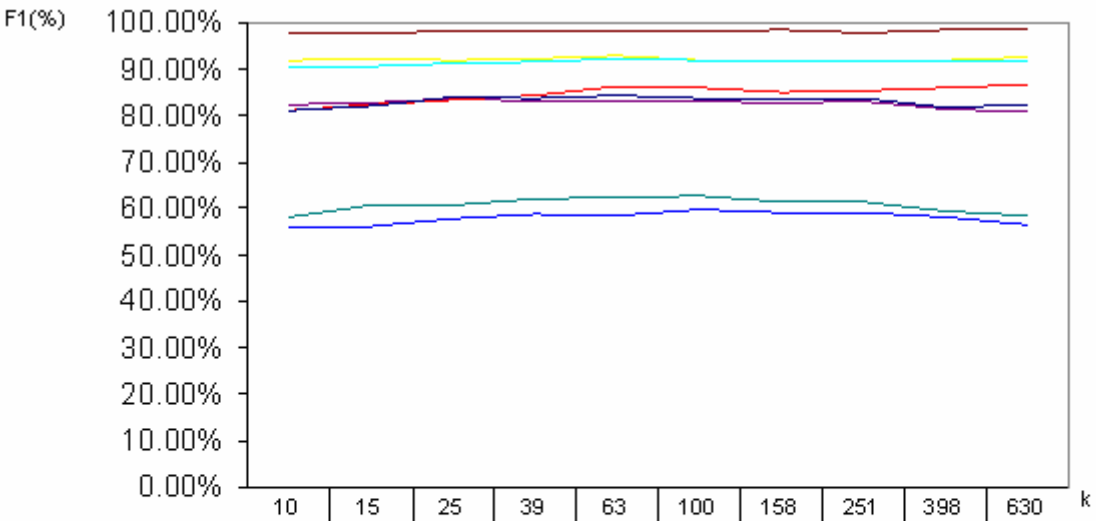
reduciranog prostora odgovaraju stupnju do kojeg dokument vektor odgovara konceptu obuhvaćenom konceptnim vektorom. Ova interpretacija preslikavanja dokument-vektora u niže dimenzionalni prostor razlog je za nazivanje ovakve redukcije dimenzionalnosti *konceptno indeksiranje* (Karypis et al., 2000).

6.1 Eksperimentalni rezultati

6.1.1 Nadzirana i nenadzirana redukcija dimenzionalnosti

Koristeći konceptno indeksiranje kao redukciju dimenzionalnosti, ulaz u SVM smanjen je sa matrice veličine *m* x *n* (100 000 x 10 000) na matricu *k* x *n* (k x 10 000). Zajedno sa smanjenjem ulaznih podataka možemo očekivati lošije rezultate koji će se postupno popravljati povećanjem *k*. Parametri SVM-a postavljeni su na $C = 1000$, $\gamma = 0.01$ postupkom krosvalidacije.

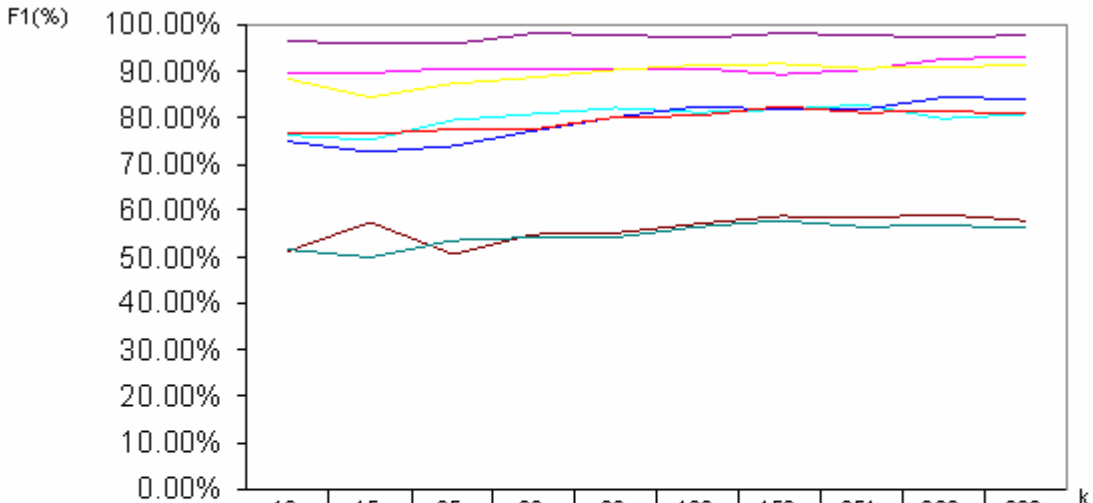
Prvo su testirani nenadzirana i nadzirana redukcija dimenzionalnosti.



	10	15	25	39	63	100	158	251	398	630
CrnaKronika	0.9178	0.9198	0.9184	0.9228	0.9264	0.917	0.9195	0.9159	0.9179	0.9241
Kultura	0.9039	0.9045	0.9114	0.9152	0.9202	0.9184	0.9182	0.9181	0.9142	0.9166
Gospodarstvo	0.8245	0.8247	0.8344	0.8292	0.8298	0.8306	0.8273	0.8295	0.8132	0.808
Sport	0.9783	0.9782	0.9789	0.9817	0.9805	0.9796	0.9828	0.978	0.9824	0.9821
UnutrasnjaPolitika	0.5809	0.6073	0.6067	0.6208	0.624	0.6249	0.6127	0.6115	0.5936	0.5839
TemaDana	0.5579	0.5622	0.5782	0.5878	0.583	0.598	0.5903	0.5909	0.5795	0.5646
Zagreb	0.8107	0.8238	0.8342	0.8416	0.8632	0.8582	0.8492	0.8522	0.8577	0.8646
VanjskaPolitika	0.811	0.8181	0.8402	0.8355	0.8416	0.8369	0.8346	0.8368	0.8163	0.8235

Slika 31. *Mjera F1 po kategorijama sa k konceptnih vektora*

Nadzirana redukcija dimenzionalnosti



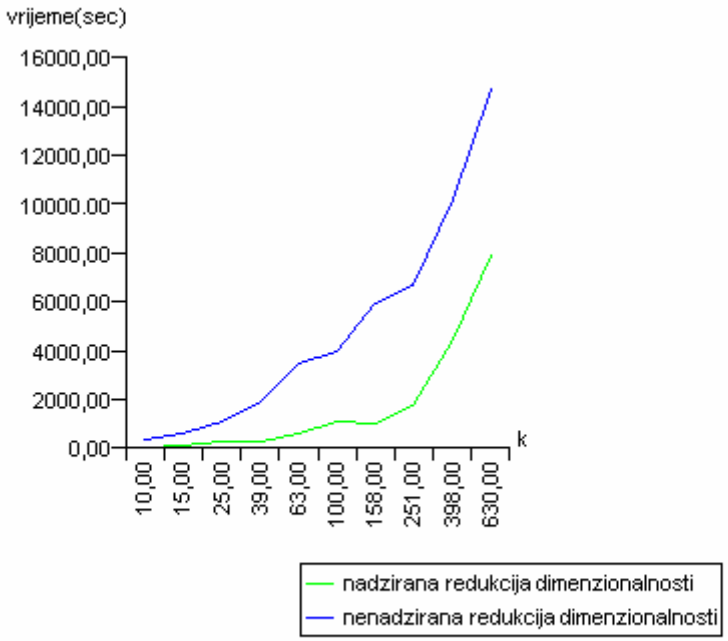
	10	15	25	39	63	100	158	251	398	630
CrnaKronika	0.8956	0.8963	0.9064	0.9028	0.901	0.9058	0.8908	0.9014	0.9233	0.9296
Kultura	0.8805	0.8422	0.8737	0.8847	0.9014	0.9128	0.9136	0.9056	0.9076	0.9123
Gospodarstvo	0.7592	0.7512	0.7932	0.8069	0.8186	0.8083	0.8169	0.8272	0.7956	0.8056
Sport	0.9637	0.9579	0.9577	0.9794	0.9781	0.9718	0.9805	0.9771	0.969	0.9786
UnutrasnjaPolitika	0.5104	0.5747	0.5065	0.5463	0.5515	0.5704	0.5881	0.5848	0.5901	0.5782
TemaDana	0.5145	0.4999	0.5358	0.54	0.5425	0.5637	0.5777	0.5631	0.5682	0.5591
Zagreb	0.7472	0.7235	0.7389	0.7692	0.7997	0.8237	0.8169	0.8157	0.8434	0.8399
VanjskaPolitika	0.7666	0.7646	0.7747	0.7753	0.8003	0.8035	0.8232	0.8112	0.8128	0.8081

Slika 32. *Mjera F1 po kategorijama sa k konceptnih vektora*

Nenadzirana redukcija dimenzionalnosti

Uspoređivanjem nadzirane i nenadzirane redukcije dimenzionalnosti, možemo vidjeti da zapravo dobivamo klasifikaciju iste točnosti. Mjera F1 za nenadziranu redukciju dimenzionalnosti je manja ili jednaka odgovarajućoj mjeri F1 nadzirane redukcije dimenzionalnosti po kategorijama. Jedina bitna razlika je vremenska. Kako kod nenadzirane redukcije trebamo uspoređivati svaki dokument vektor sa svim konceptnim vektorima kroz iteracije, a kod nadzirane samo sa konceptnim vektorima njegove kategorije, možemo zaključiti i dokazati da će nenadzirana redukcija dimenzionalnosti biti znatno sporija.

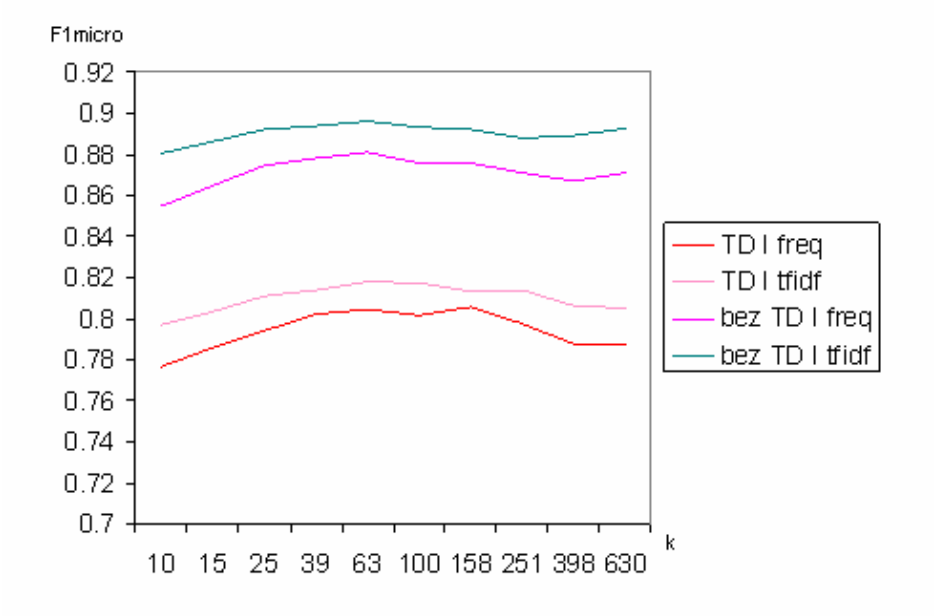
U sljedećim testiranjima korištena je samo nadzirana redukcija dimenzionalnosti te su se mijenjale matrice *riječ x dokument* dobivene različitom morfološkom normalizacijom. Tri vrste matrica „-i“, „-id“, „-idt“ označavaju 3 vrste morfološke normalizacije kojom su dobivene: flektivna, derivacijska te terminirajuća.



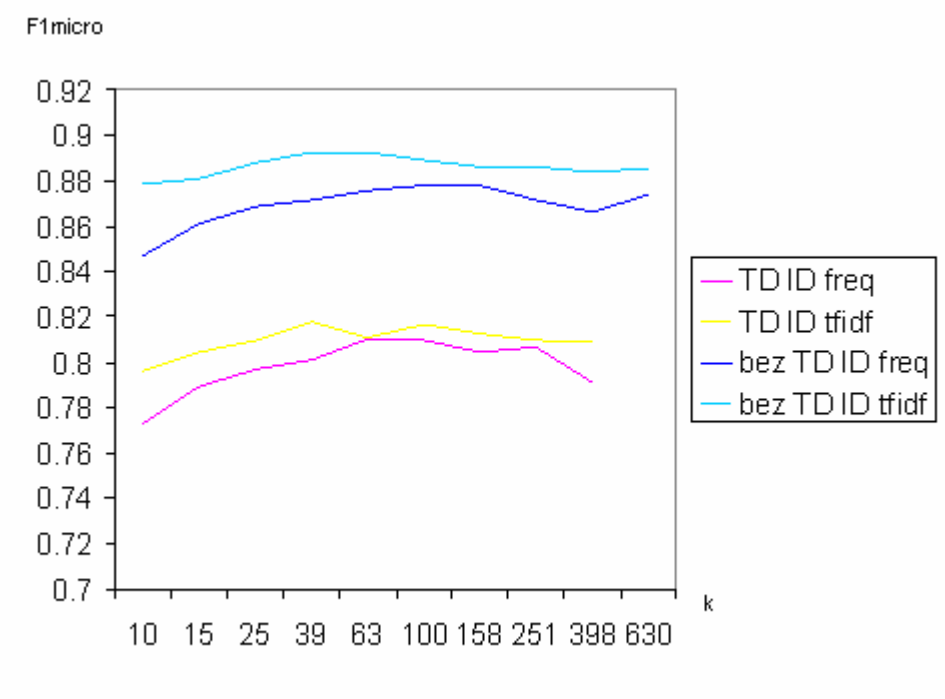
Slika 33. Vremenska ovisnost(sekunde) algoritma redukcijom dimenzionalnosti u k grupa

Usporedba nadzirane I nenadzirane redukcije dimenzionalnosti

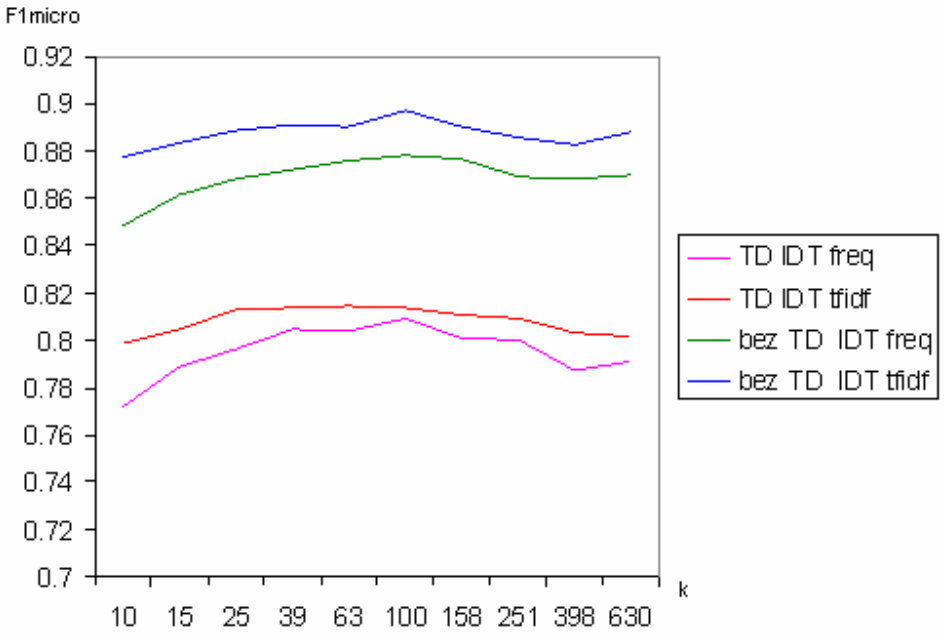
6.1.2 Utjecaj morfološke normalizacije na hrvatski jezik



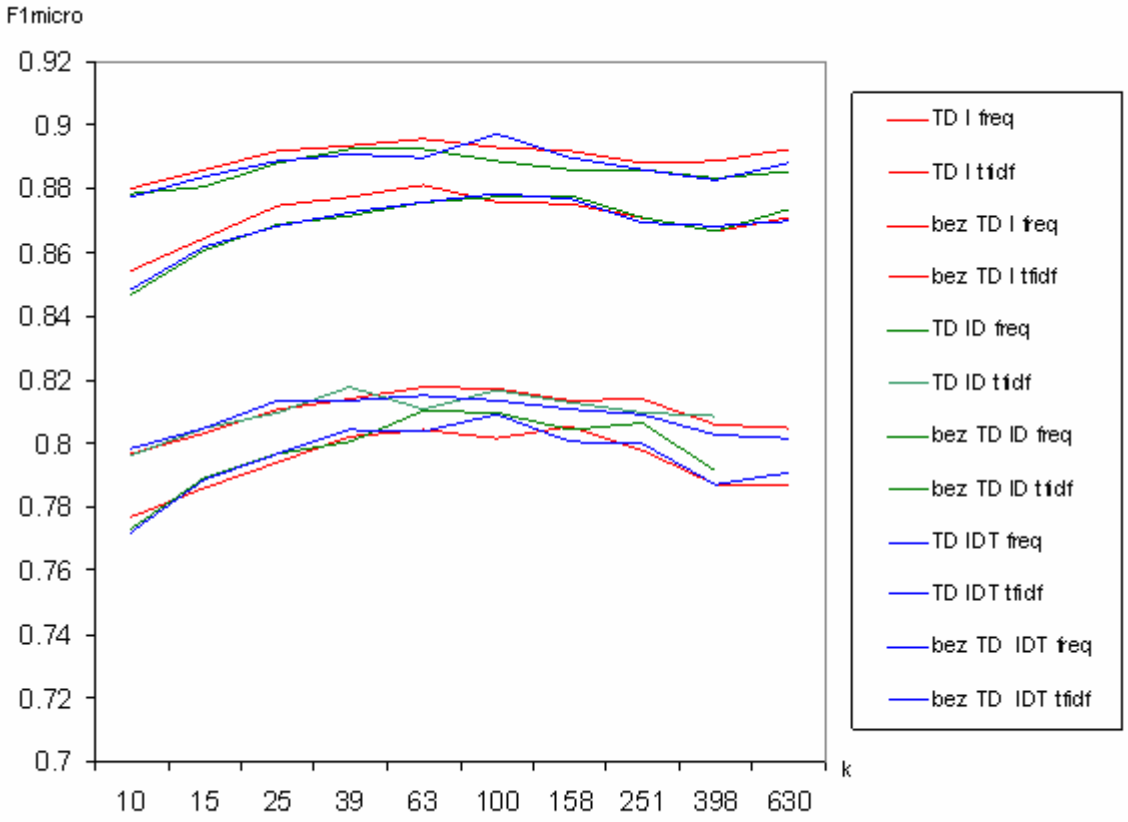
Slika 34. Mjera $F1^{micro}$ za baze sa i bez teme dana(TD, bez TD), korištenjem frekvencija ili mjere tfidf za flektivnu morfološku normalizaciju



Slika 35. Mjera $F1^{micro}$ za baze sa i bez teme dana(TD, bez TD), korištenjem frekvencija ili mjere tfidf za derivacijsku morfološku normalizaciju



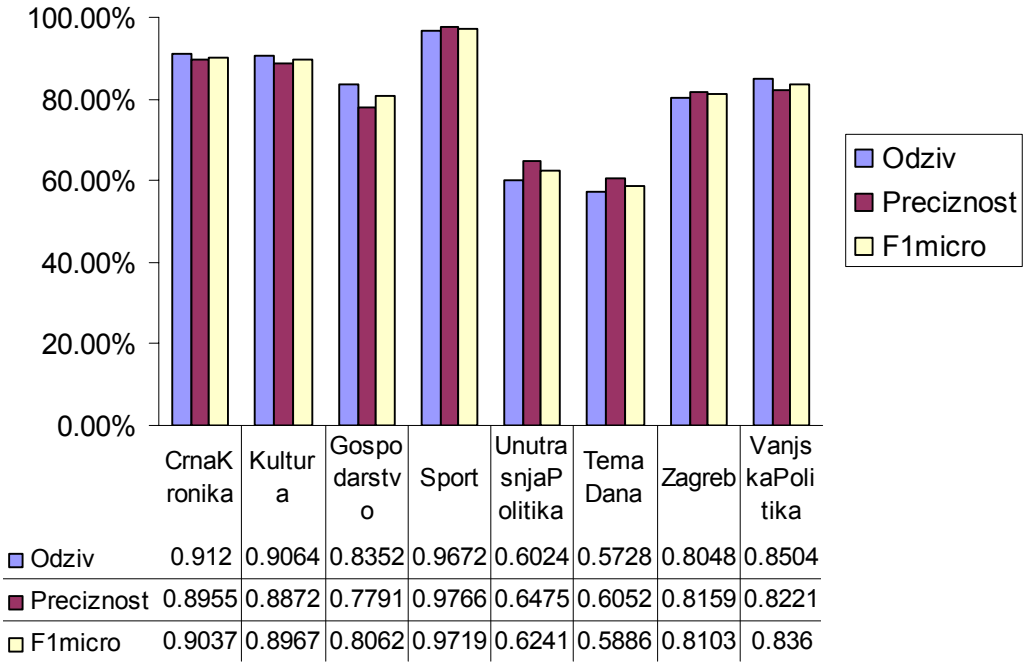
Slika 36. Mjera $F1^{micro}$ za baze sa i bez teme dana(TD, bez TD), korištenjem frekvencija ili mjere tfidf za terminirajuću morfološku normalizaciju



Slika 37. Mjera $F1^{micro}$ za baze sa i bez teme dana(TD, bez TD), sve 3 morfološke normalizacije(I,ID,IDT) te korištenjem frekvencija ili mjere tfidf po broju korištenih konceptnih vektora

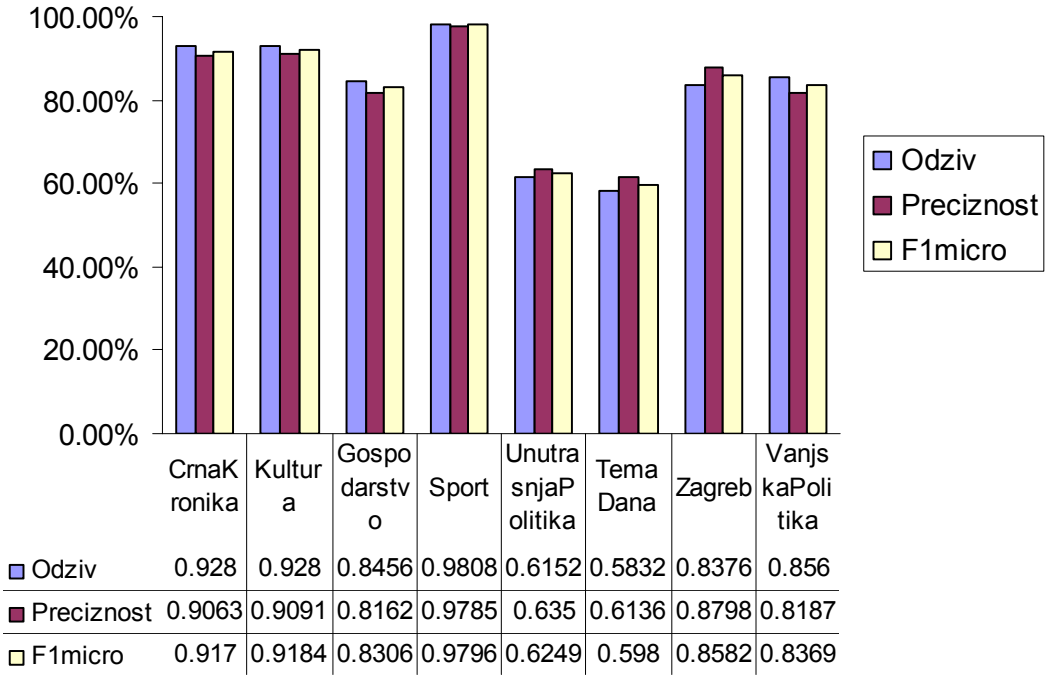
Sve morfološke obrade daju približno iste rezultate, iako ih možemo sortirati po neznatnoj uspješnosti: flektivna, terminirajuća, derivacijska. Velike su promjene u rezultatima sa i bez kategorije tema dana, te u oba slučaja tfidf pokazuje se kao bolji odabir.

6.1.3 Mjere uspješnosti za različite morfološke normalizacije, sa i bez kategorije tema dana (broj konceptnih vektora: 100)



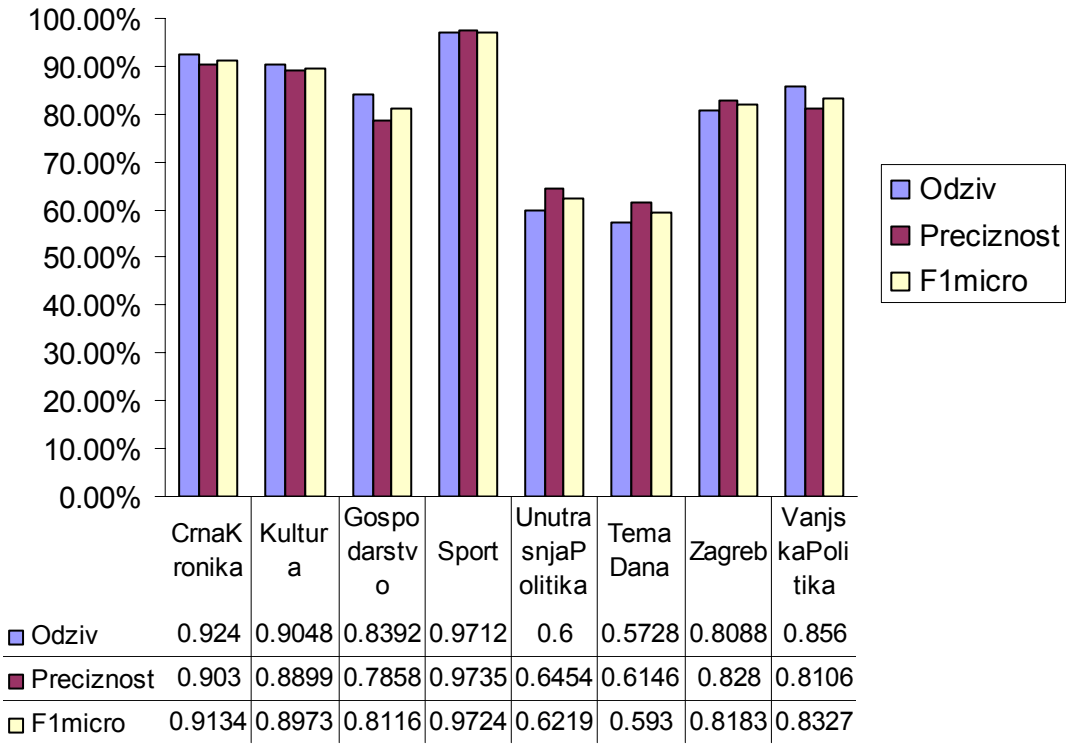
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	TemaDana	Zagreb	VanjskaPolitika
Odziv	0.912	0.9064	0.8352	0.9672	0.6024	0.5728	0.8048	0.8504
Preciznost	0.8955	0.8872	0.7791	0.9766	0.6475	0.6052	0.8159	0.8221
F1micro	0.9037	0.8967	0.8062	0.9719	0.6241	0.5886	0.8103	0.836

Slika 38. Mjera $F1^{micro}$ za flektivnu morfološku normalizaciju, frekvencija



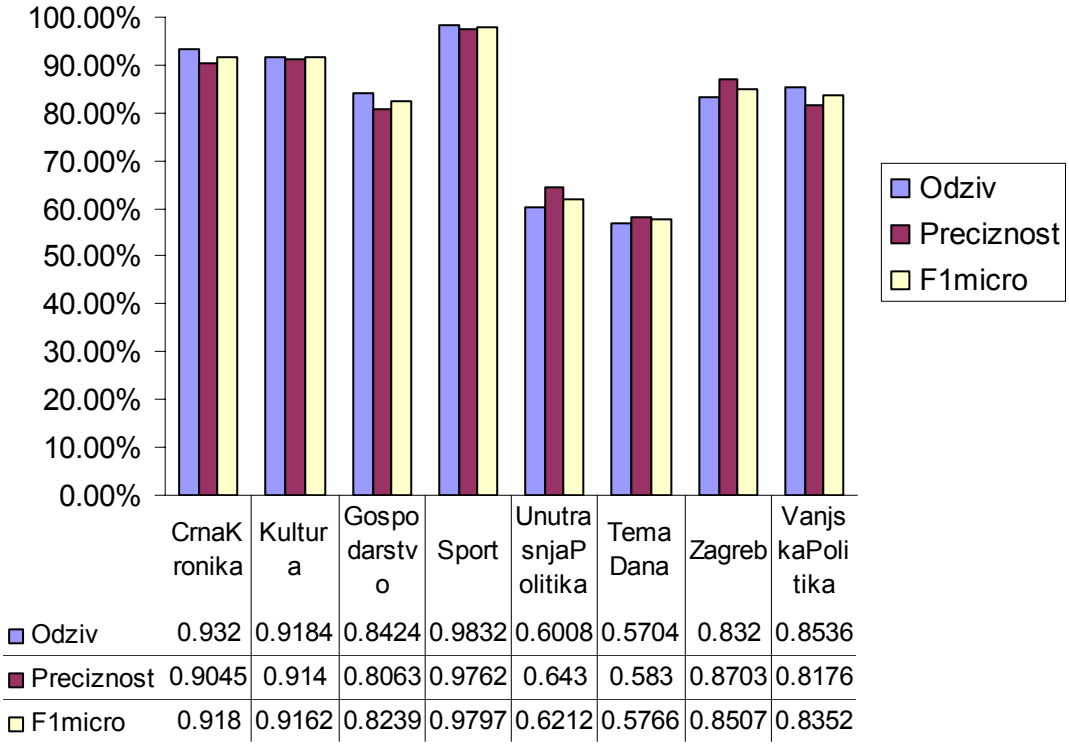
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	TemaDana	Zagreb	VanjskaPolitika
Odziv	0.928	0.928	0.8456	0.9808	0.6152	0.5832	0.8376	0.856
Preciznost	0.9063	0.9091	0.8162	0.9785	0.635	0.6136	0.8798	0.8187
F1micro	0.917	0.9184	0.8306	0.9796	0.6249	0.598	0.8582	0.8369

Slika 39. Mjera $F1^{micro}$ za flektivnu morfološku normalizaciju, tfidf



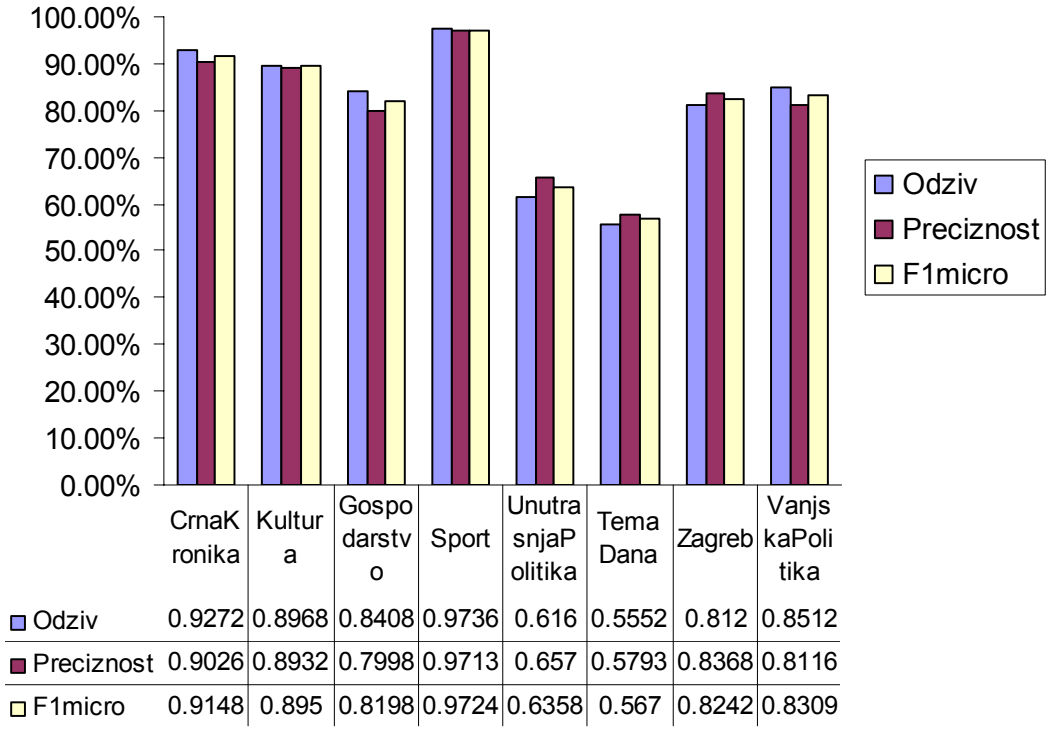
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	TemaDana	Zagreb	VanjskaPolitika
Odziv	0.924	0.9048	0.8392	0.9712	0.6	0.5728	0.8088	0.856
Preciznost	0.903	0.8899	0.7858	0.9735	0.6454	0.6146	0.828	0.8106
F1micro	0.9134	0.8973	0.8116	0.9724	0.6219	0.593	0.8183	0.8327

Slika 40. Mjera $F1^{micro}$ za derivacijsku morfološku normalizaciju, frekvencija



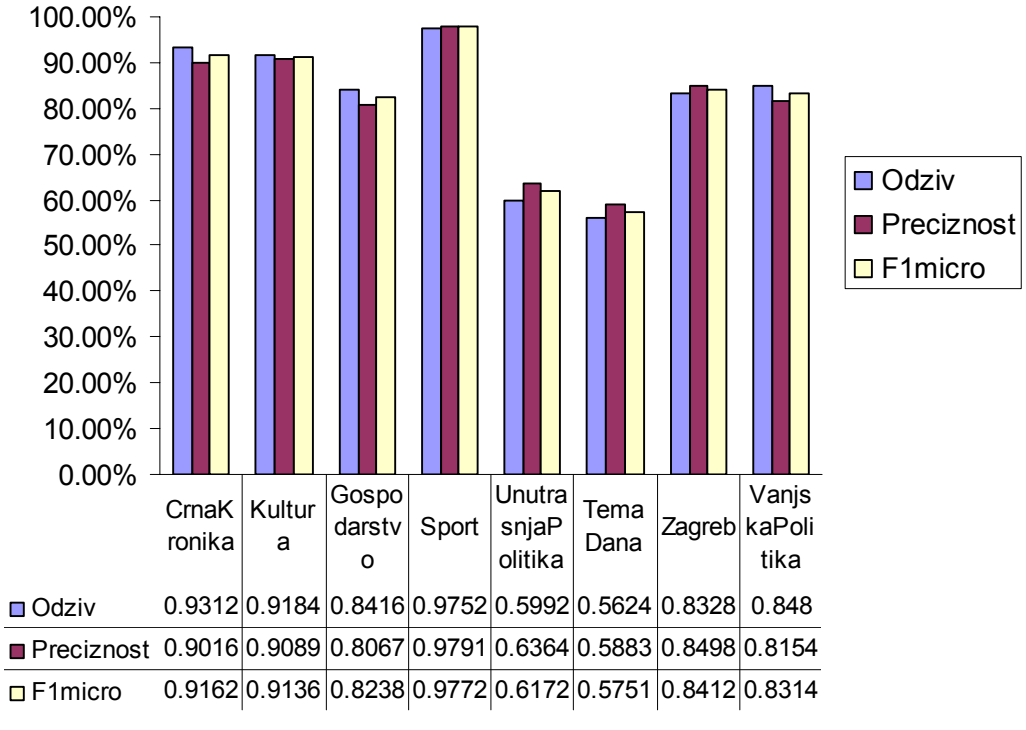
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	TemaDana	Zagreb	VanjskaPolitika
Odziv	0.932	0.9184	0.8424	0.9832	0.6008	0.5704	0.832	0.8536
Preciznost	0.9045	0.914	0.8063	0.9762	0.643	0.583	0.8703	0.8176
F1micro	0.918	0.9162	0.8239	0.9797	0.6212	0.5766	0.8507	0.8352

Slika 41. Mjera $F1^{micro}$ za derivacijsku morfološku normalizaciju, tfidf



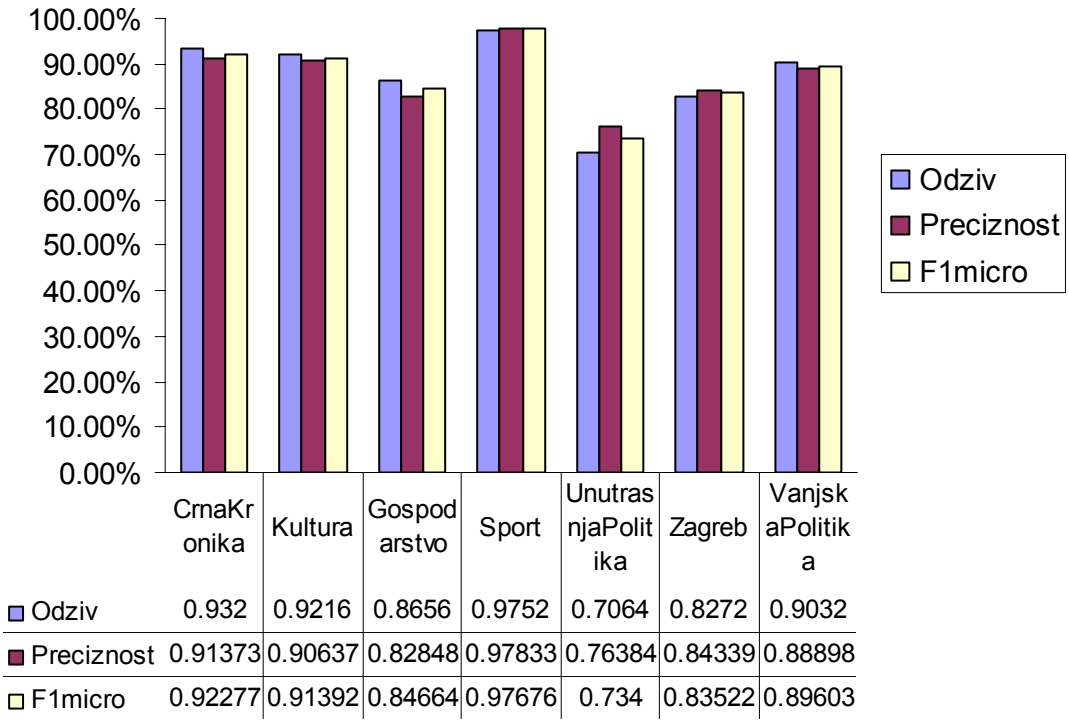
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	TemaDana	Zagreb	VanjskaPolitika
Odziv	0.9272	0.8968	0.8408	0.9736	0.616	0.5552	0.812	0.8512
Preciznost	0.9026	0.8932	0.7998	0.9713	0.657	0.5793	0.8368	0.8116
F1micro	0.9148	0.895	0.8198	0.9724	0.6358	0.567	0.8242	0.8309

Slika 42. Mjera $F1^{micro}$ za terminirajuću morfološku normalizaciju, frekvencija



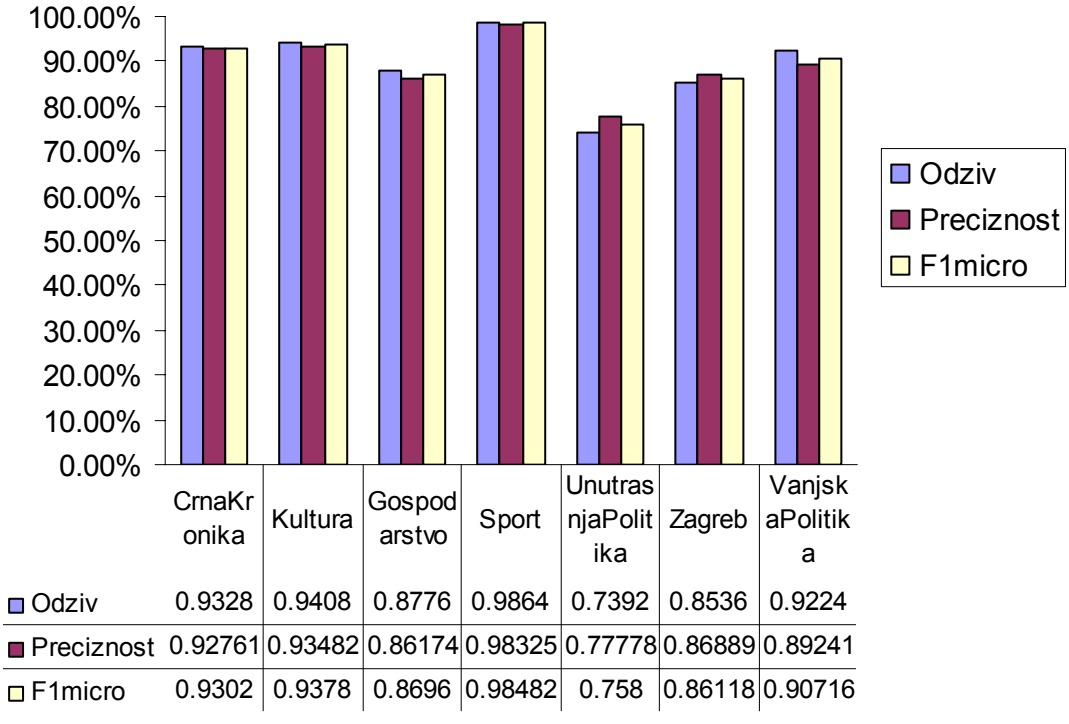
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	TemaDana	Zagreb	VanjskaPolitika
Odziv	0.9312	0.9184	0.8416	0.9752	0.5992	0.5624	0.8328	0.848
Preciznost	0.9016	0.9089	0.8067	0.9791	0.6364	0.5883	0.8498	0.8154
F1micro	0.9162	0.9136	0.8238	0.9772	0.6172	0.5751	0.8412	0.8314

Slika 43. Mjera $F1^{micro}$ za terminirajuću morfološku normalizaciju, tfidf



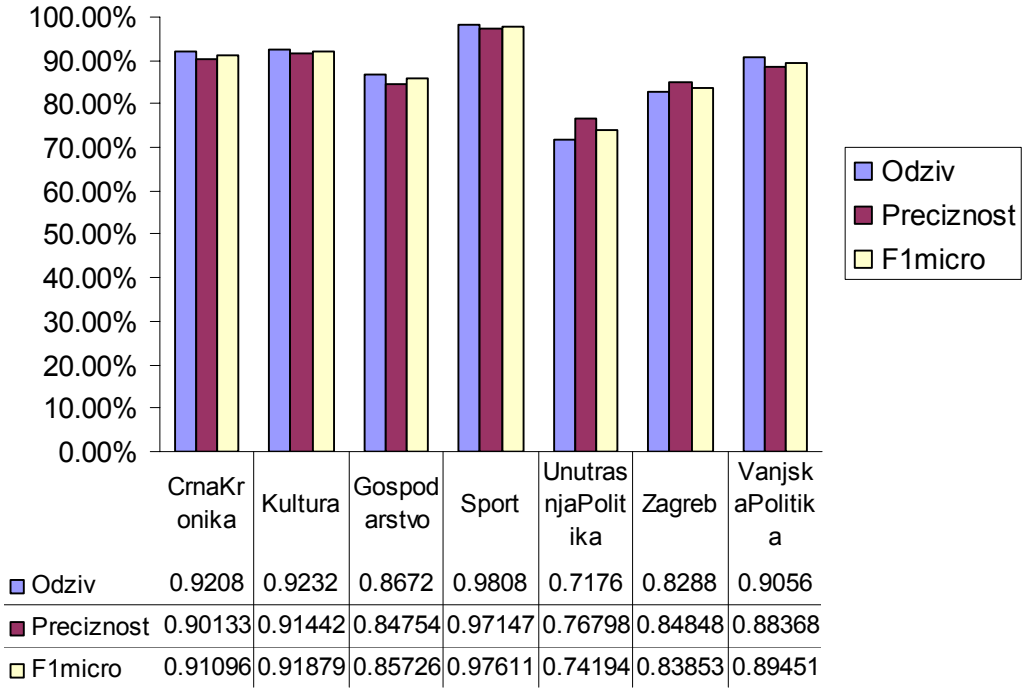
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	Zagreb	VanjskaPolitika
Odziv	0.932	0.9216	0.8656	0.9752	0.7064	0.8272	0.9032
Preciznost	0.91373	0.90637	0.82848	0.97833	0.76384	0.84339	0.88898
F1micro	0.92277	0.91392	0.84664	0.97676	0.734	0.83522	0.89603

Slika 44. Mjera $F1^{micro}$ za flektivnu morfološku normalizaciju, frekvencija, bez teme dana



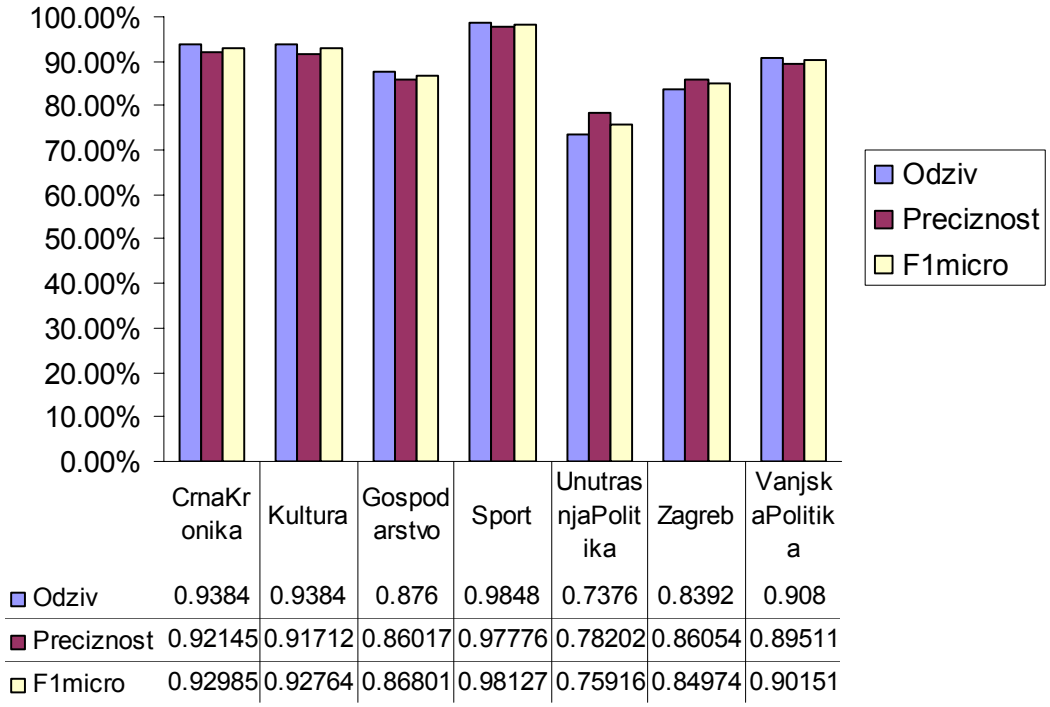
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	Zagreb	VanjskaPolitika
Odziv	0.9328	0.9408	0.8776	0.9864	0.7392	0.8536	0.9224
Preciznost	0.92761	0.93482	0.86174	0.98325	0.77778	0.86889	0.89241
F1micro	0.9302	0.9378	0.8696	0.98482	0.758	0.86118	0.90716

Slika 45. Mjera $F1^{micro}$ za flektivnu morfološku normalizaciju, tfidf, bez teme dana



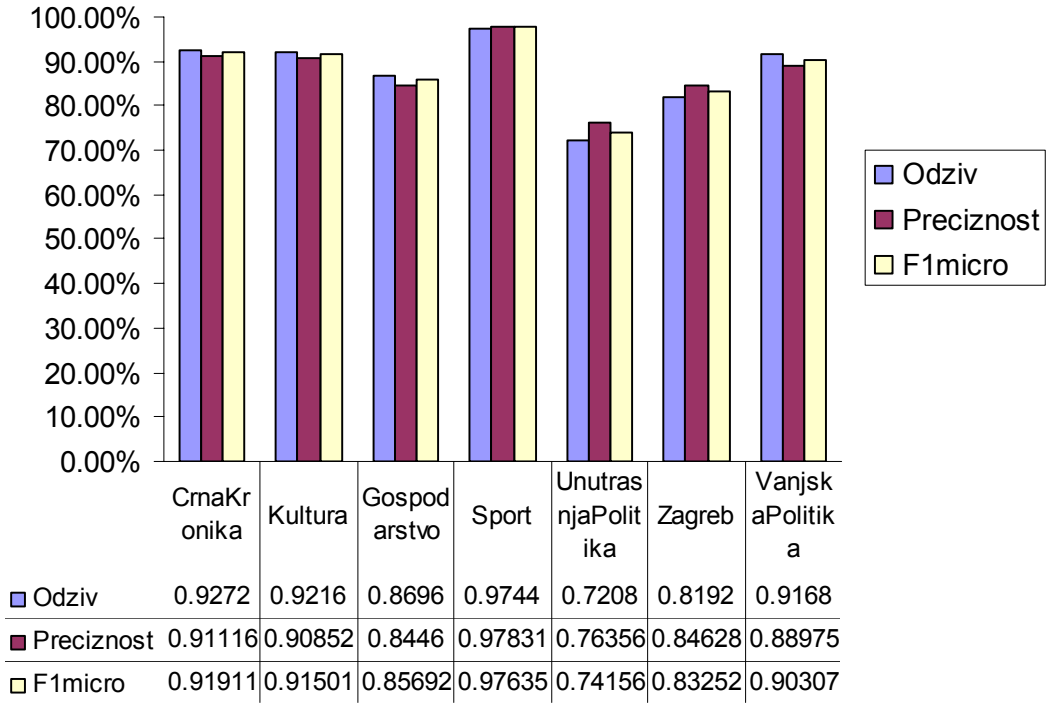
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	Zagreb	VanjskaPolitika
Odziv	0.9208	0.9232	0.8672	0.9808	0.7176	0.8288	0.9056
Preciznost	0.90133	0.91442	0.84754	0.97147	0.76798	0.84848	0.88368
F1micro	0.91096	0.91879	0.85726	0.97611	0.74194	0.83853	0.89451

Slika 46. Mjera $F1^{micro}$ za derivacijsku morfološku normalizaciju, frekvencija, bez teme dana



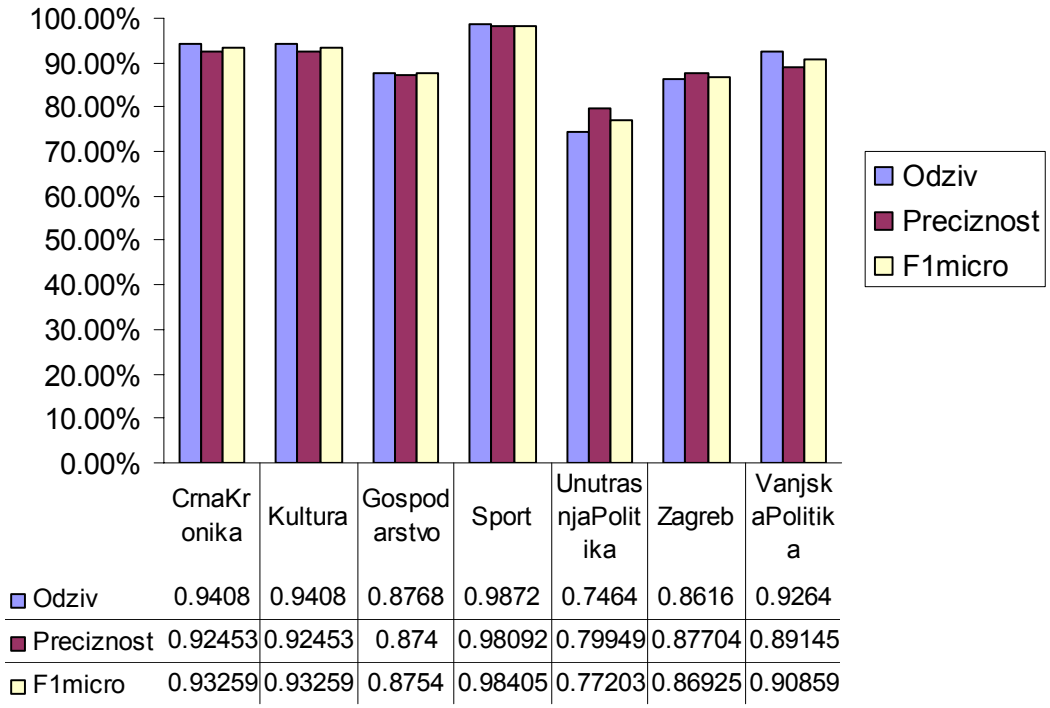
	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	Zagreb	VanjskaPolitika
Odziv	0.9384	0.9384	0.876	0.9848	0.7376	0.8392	0.908
Preciznost	0.92145	0.91712	0.86017	0.97776	0.78202	0.86054	0.89511
F1micro	0.92985	0.92764	0.86801	0.98127	0.75916	0.84974	0.90151

Slika 47. Mjera $F1^{micro}$ za derivacijsku morfološku normalizaciju, tfidf, bez teme dana



	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	Zagreb	VanjskaPolitika
Odziv	0.9272	0.9216	0.8696	0.9744	0.7208	0.8192	0.9168
Preciznost	0.91116	0.90852	0.8446	0.97831	0.76356	0.84628	0.88975
F1micro	0.91911	0.91501	0.85692	0.97635	0.74156	0.83252	0.90307

Slika 48. Mjera $F1^{micro}$ za terminirajuću morfološku normalizaciju, frekvencija, bez teme dana



	CrnaKronika	Kultura	Gospodarstvo	Sport	UnutrasnjaPolitika	Zagreb	VanjskaPolitika
Odziv	0.9408	0.9408	0.8768	0.9872	0.7464	0.8616	0.9264
Preciznost	0.92453	0.92453	0.874	0.98092	0.79949	0.87704	0.89145
F1micro	0.93259	0.93259	0.8754	0.98405	0.77203	0.86925	0.90859

Slika 49. Mjera $F1^{micro}$ za terminirajuću morfološku normalizaciju, tfidf, bez teme dana

7 ZAKLJUČAK

Nakon završetka ovog rada, dok pišem zadnju stranicu, glavna misao mi je kako i na čemu je moguće doraditi metode koje sam obuhvatila i proučila. Istovremeno se nadam da će ovaj diplomski biti dobar temelj za istraživanja koja slijede.

Donedavno područje dubinske analize teksta na hrvatskom jeziku nije ni postojalo. Buđenje i nagli razvoj duguje text mining timu sa Fakulteta elektrotehnike i računarstva u Zagrebu, a ponajviše sjajnoj mentorici i voditeljici prof. dr. sc. Bojani Dalbelo-Bašić. Stalno pojavljivanje novih baza svjedoči o potrebama ovog područja, a njihova sve veća veličina tjera na daljnju optimizaciju, istraživanje i usavršavanje ovog područja. Kolegama koje interesira ovo područje ostavljam prvu stepenicu i još mnoga otvorena pitanja koja se mogu poboljšavati. Smatram da je zbog bogatstva hrvatskog jezika napredak na ovom području moguć s razvojem lingvistike, sintakse i semantike i njihovom sve boljom računalnom obradom.

8 LITERATURA

M. Malenica. *Primjena jezgrenih metoda u kategorizaciji teksta*, diplomski rad, FER, Zagreb, rujan 2004.

Marko Tadić. *Hrvatski nacionalni korpus na Internetu*, 2005. [http://www.hnk.ffzg.hr/]

I.S.Dhillon, D.S. Modha. *Concept decomposition for large sparse text data using clustering*, Machine Learning, 42:1, 2001, pp. 143-175

Jing Gao, and Jun Zhang. *Text Retrieval Using Sparsified Concept Decomposition Matrix*, Technical Report 412-04, Department of Computer Science, University of Kentucky, Lexington, KY, 2004. [http://cs.engr.uky.edu/~jzhang/pub/MINING/gao3.ps.gz]

N. Cristianini, J. Shawe-Taylor. *An Introduction to Support Vector Machines (and other kernel based learning methods)*, Cambridge University Press, Cambridge, 2000.

T. Joachims. *Learning to classify text using support vector machines*, Kluwer, Massachusetts, 2001.

M.A. Hearst. *Trends and Controversies: Support Vector Machines*, IEEE Intelligent Systems, 13(4)., 1998 [www.computer.org/intelligent/ex1998/pdf/x4018.pdf]

G. Karypis, E.-H. Han. *Concept indexing: A fast dimensionality reduction algorithm with applications to document retrieval and categorization,* Technical Report TR-00-0016, University of Minnesota, 2000. [http://citeseer.ist.psu.edu/karypis00concept.html]

Golub, G. H.,C. F. Van Loan. *Matrix computations*. Baltimore, MD, USA: The Johns Hopkins University Press, 1996.

Mandelbrot, B. B. *Fractal Geometry of Nature*. W. H. Freeman & Company, 1988.

E. Xing, A. Ng, M. Jordan, and S. Russell. *Distance metric learning, with application to clustering with side-information*. Advances in Neural InformationProcessing Systems 15, 2003. [http://books.nips.cc/papers/files/nips15/AA03.pdf]

Gang Wu, Navneet Panda, Edward Y. Chang. *Formulating Distance Functions via the Kernel Trick*, ACM International Conference on Knowledge Discovery and Data Mining (KDD), Chicago, August 2005 (future plan)

A. Bar-hillel, T. Hertz, N. Shental, and D. Weinshall. *Learning distance functions using equivalence relations*. In Proceedings of the Twentieth International Conference on Machine Learning, August 2003.[http://www.hpl.hp.com/conferences/icml2003/papers/88.pdf]

C. C. Aggarwal. *Towards systematic design of distance functions for data mining applications.* The Ninth ACM SIGKDD International Conference on Knowledge Discovery in Data and Data Mining, 2003.[http://citeseer.ist.psu.edu/717882.html]

Nello Cristianini, John Shawe-Taylor, Andre Elisseeff, Jaz Kandola. *On Kernel-target alignment,* Neural Information Processing Systems 2001. [www.support-vector.net/papers/nips01_alignment.ps]

Daichi Mochihashi, Genichiro Kikui, Kenji Kita. *Learning Nonstructural Distance Metric by Minimum Cluster Distortions,* Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004), pp.341-348, Barcelona, July 2004. [acl.ldc.upenn.edu/acl2004/emnlp/pdf/Mochihashi.pdf]

Rong Zhang and Alexander I. Rudnicky, *A Large Scale Clustering Scheme for Kernel K-Means*, *icpr*, vol. 04, no. 4, p. 40289, 16 2002.

Z. Zhang. *Learning metrics via discriminant kernels and multidimensional scaling : Towards expected Euclidean representation*. In Proceedings of the Twentieth International Conference on Machine Learning, August 2003.

J. T. Kwok and I. W. Tsang. *Learning with idealized kernels.* In Proceedings of the Twentieth International Conference on Machine Learning, pages 400–407, August 2003.

Alexei Vinokourov, John Shawe-Taylor, Nello Cristianini. *Inferring a Semantic Representation of Text via Cross-Language Correlation Analysis*. In Advances of Neural Information Processing Systems 15, 2002.[www.ecs.soton.ac.uk/~av/alexei_nips02.pdf]

Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a Library for Support Vector Machines*, 2005. [http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf]

Blaž Fortuna. *Kernel Canonical Correlation Analysis With Applications.* SIKDD 2004 at multiconference IS 2004, 12-15 Oct 2004, Ljubljana, Slovenia 2004. [eprints.pascal-network.org/ archive/00000736/01/BlazFortuna-Kcca.pdf]

J.Shawe-Taylor, N.Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

C.-W. Hsu and C.-J. Lin. *A comparison of methods for multi-class support vector machines.* IEEE Transactions on Neural Networks, 13(2):415–425, 2002. [http://citeseer.ist.psu.edu/hsu01comparison.html]

Lehoucq, R.B., D.C. Sorensen, and C. Yang, *ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, SIAM Publications, Philadelphia, 1998.[http://www.caam.rice.edu/software/ARPACK/UG/ug.html]

Steve Gunn. *Support Vector Machiens for Classification and Regression*. ISIS Technical Report ISIS-1-98, Image Speech & Intelligent Systems Research Group, University of Southapton, May. 1998

http://www.kernel-methods.net/index.html Dodatak knjige J.Shawe-Taylor, N.Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

http://www.zemris.fer.hr/projects/textmining Stranice text mining projekta na Fakultetu elektrotehnike i računarstva u Zagrebu, voditelj projekta: Prof. dr. sc. Bojana Dalbelo-Bašić

9 PRIMJER RADA METODE GLAVNIH KOMPONENATA I KANONSKE KORELACIJSKE ANALIZE

Smisao ovog poglavlja je približiti navedene metode čitatelju. Kako se ne bi udaljavali od podataka korištenih pri analizi odabrani su neki od naslova članaka iz kategorije politika i kategorije gospodarstvo baze čanaka Croatia Weekly. Slijede naslovi tih članaka zajedno sa kategorijom i prijevodom na engleski jezik:

Skup za učenje:

1 go	Svjetska financijska kriza mimoići će Hrvatsku	World financial crisis passing Croatia
2 go	Zagrebačka banka dobila veliki kredit ebrd-a	Zagreb bank secures loan of ebdr
3 go	Svjetska banka odobrila 101 dolara kredita hrvatske željeznice	World bank lends millions for croatia railroad company
4 go	Svjetska banka potpora hrvatskoj 200 dolara	World bank million of support for Croatia
5 go	Bankarska kriza mora biti riješena sljedeće godine	Financial crisis must be resolved next year
6 go	Hrvatsko informatičko tržište vrijedi 300 dolara	Croatian information technology market worth million usd
7 go	Kredit ebrd-a 168 dem za sanaciju zagrebačkog odlagališta otpada	Ebrd grants 168 million dm loan for Zagreb landfill rehabilitation
8	Najmanje 330 dem od prodaje pbz-a	330 million dm for state coffers from pbz

go	proračunu do kraja godine	sale
9 po	Sporazum otvaranju graničnih prijelaza mora se poštivati	Agreement opening border crossing observed
10 po	Bošnjaci odbili potpisati parafirani sporazum	Bosniaks refuse to sign agreement
11 po	Preko graničnih prijelaza u Istri ušlo dosta turista	Border crossing istria tourists
12 po	Uskršnja najezda stranih turista na Jadran	Eastern invasion of foreign tourists
13 po	Bošnjaci pristaju na dogovor o humanom preseljenju	Bosniaks agree to humane relocation
14 po	Dan potpisivanja daytonskog sporazuma državni praznik	Day of signing daython agreement is state holiday
15 po	Pardew zagovara reviziju daytonskog sporazuma	Pardew jolted revision of daython agreement
16 po	Mjesec dana za dogovor o izbornom zakonu	Month day for agreement about election

Skup za testiranje:

1 go	Zagrebačka banka prodaje tvornicu duhana zagreb	Zagreb bank sale factory of tabaco Zagreb
2 go	Svjetska banka strukturne promjene preduvjet gospodarskog rasta	World bank structural prerequisite economic growth
3 po	U srpnju sporazum o imovinsko-pravnim odnosima	Agreement about property rights relations in july
4 po	Oess-ov sporazum sa srj spominje icty	Osce agreement with Yugoslavia fails to mention icty

Crvenom bojom označene su riječi koje se pojavljuju više od jedanput. Za izgradnju dokument-riječ matrice potrebno je parsirati i lematizirati riječi koje se nalaze u skupu za učenje. Odbacuju se stop riječi, brojevi i interpunkcijski znakovi. Prolaskom kroz sve primjere iz skupa za učenje generira se lista svih izraza koji se pojavljuju te im se dodjeljuje broj koji će ih dalje zamjenjivati kao indeks u dokument-riječ matrici. Ovakav postupak potreban je za dokumente na hrvatskom i posebno na engleskom jeziku. Broj riječi u engleskom i hrvatskom neće biti isti jer prijevod nije dostavan.

1. *svjetska*
2. *financijska*
3. *kriza*
4. *mimioći*
5. *hrvatska*
6. *zagrebačka*
7. *banka*
8. *dobiti*
9. *veliki*
10. *kredit*
11. *erdb-a*
12. *…*

1. *world*
2. *finincial*
3. *crisis*
4. *passing*
5. *croatia*
6. *zagreb*
7. *bank*
8. *secures*
9. *loan*
10. *ebdr*
11. *…*

Na temelju rječnika tvore se vektori za svaki dokument. Dimenzionalnost vektora je broj riječi koje su se pojavile u skupu za učenje, a svaki element vektora predstavlja frekvenciju pojavljivanja riječi u razmatranom dokumentu. Kako se u naslovima na hrvatskom jeziku pojavljuje 65 riječi, 65 je indeks zadnje riječi(“zakon”) i dimenzionalnost svih vektora. Slaganjem dokument-vektora u matricu dobijemo dokument-riječ matricu.

	svjetska	financijsk	kriza	mimoići	hrvatska	Zagrebač		zakon
1	1	1	1	1	1	0		0
2	0	0	0	0	0	1		0
3	1	0	0	0	0	0		0
4	1	0	0	0	0	0		0
5	0	0	1	0	0	0		0
6	0	0	0	0	0	0		0
7	0	0	0	0	0	1		0
8	0	0	0	0	0	0		0
9	0	0	0	0	0	0		0
10	0	0	0	0	0	0		0
11	0	0	0	0	0	0		0
12	0	0	0	0	0	0		0

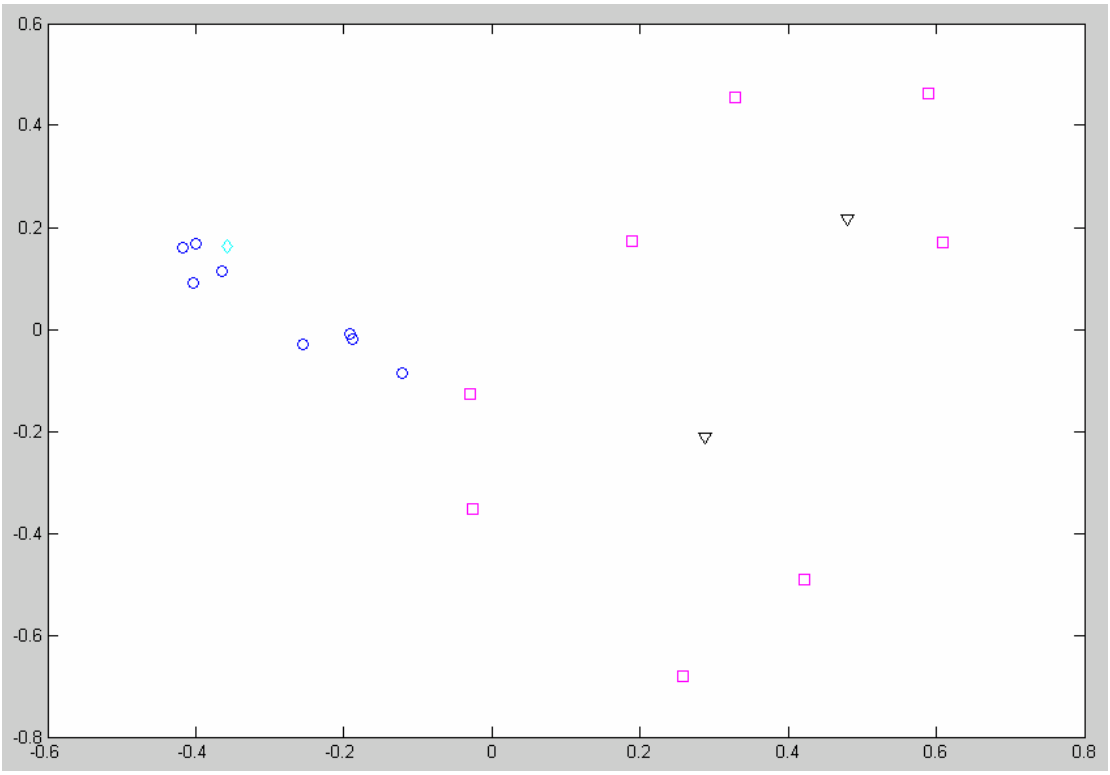
13	0	0	0	0	0	0		0
14	0	0	0	0	0	0		0
15	0	0	0	0	0	0		0
16	0	0	0	0	0	0		1

Dimenzionalnost matrice dokument-riječ za odabrane naslove Croatia Weekly

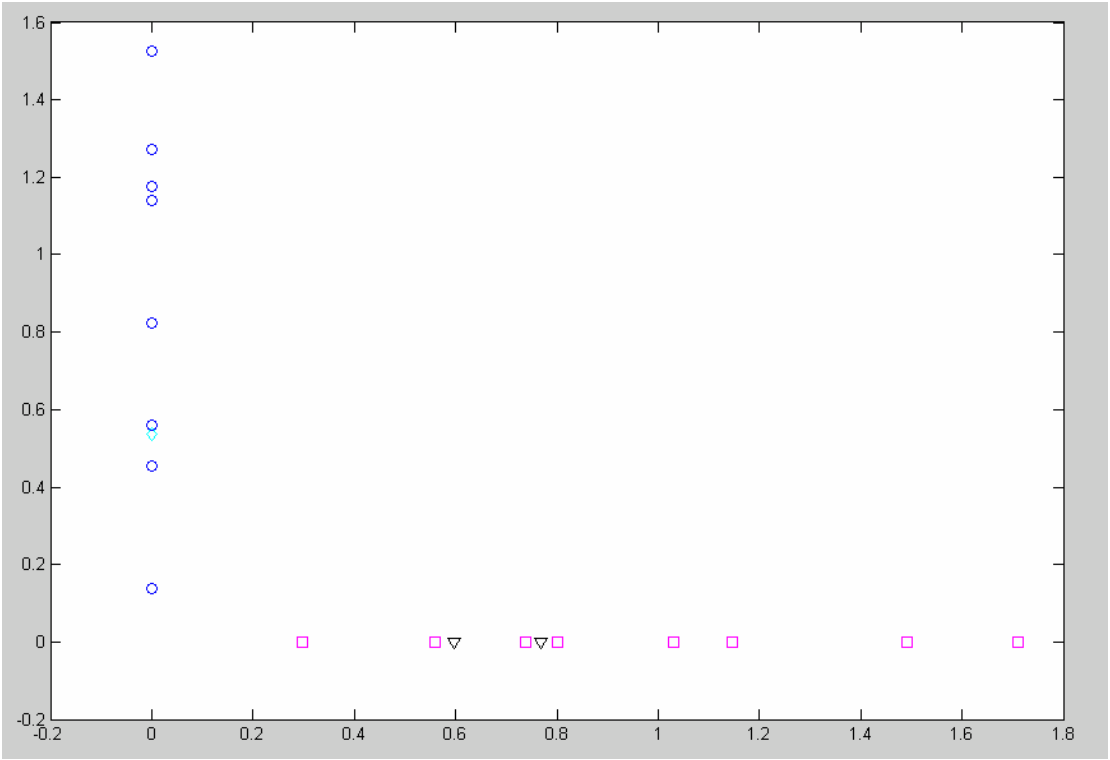
	Hrvatski	**Engleski**
Skup za učenje	16 x 65	16 x 58
Skup za treniranje	4 x 65	4 x 58

Ovakve matrice ulaz su u metodu glavnih komponenti i kanonsku korelacijsku analizu, metode za redukciju dimenzionalnosti. Radi boljeg prikaza i razumljivosti preslikavat ćemo dokument-vektore u dvodimenzionalni prostor. Potrebno je pronaći dva vektora u višedimenzionalnom prostoru riječi na koje će se dokument-vektori preslikati. Svaka metoda ima svoj način pronalaska takvih vektora. Vidjet ćemo koliko će vektori pronađeni metodom glavnih komponenata zadovoljiti naše potrebe. Kako ta metoda nema pristupa oznakama dokumenata(kategorijama) preostaje joj da potraži smjerove maksimalne varijance podataka i nada se da smjerovi male varijance predstavljaju samo šum. Kanonska korelacijska analiza traži smjerove maksimalne korelacije među prikazima dokumenta na dva jezika. Engleski jezik nam zapravo služi kao kompleksna oznaka kad već nemam pristupa stvarnim oznakama kategorije. Neke riječi kao npr. dogovor i sporazum u engleskom jeziku se preslikavaju u jednu riječ – agreement, te ne treba spominjati pozitivne promjene koje će to uvesti na ovako malom skupu i povezati dokumente koji nemaju nikakvih sličnosti na hrvatskom jeziku. Ako pažljivije pogledate naslove i njihove prijevode vidjet ćete da engleski nije doslovan prijevod hrvatskog i da se neke riječi dodatno pojavljuju u engleskom i povezuju inače odvojene dokumente.

Na slici su prikazani preslikani dokumenti u novom dvodimenzionalnom prostoru određenom vektorima dobivenim metodom glavnih komponenata i kanonskom korelacijskom analizom. Rozom i plavom prikazani su redom dokumenti kategorije gospodarstvo te kategorije politika iz skupa za učenje. Crno je skup za testiranje kategorije gospodarstvo, a svijetlo plavo skup za testiranje kategorije politika.



Slika 50. Dvodimenzionalni preslikani dokument-vektori dobiveni metodom glavnih komponenata



Slika 51. Dvodimenzionalni preslikani dokument-vektori dobiveni kanonskom korelacijskom analizom

Skup za učenje i treniranje redukcijom dimenzionalnosti kanonskom korelacijskom analizom postaje jasno i savršeno odvojiv. Za *x = 0* možemo klasificirati dokument u klasu politika sa 100% točnošću, te također za *y = 0* možemo tvrditi da je kategorija dokumenta gospodarstvo. Vektori pronađeni kanonskom korelacijskom analizom jasno su pridijeljeni svaki svojoj kategoriji, te su okomiti na sve dokument-vektore suprotne kategorije. Za metodu glavnih komponenata to nije slučaj. Dokument-vektori su linearna kombinacija vektora pronađenih ovom metodom. Samo sa jednim, prvim vektorom, možemo jasno odrediti klasifikaciju ako postavimo granicu za *x >= 0*. Negativne vrijednosti komponenata vektora dobivene su zbog centriranja ulaznih podataka.

Ako se sjećamo definicije matrice bliskosti i jezgrene funkcije,

$$\tilde{k}(d_1,d_2) = \phi(d_1)\mathbf{U_k U_k'}\phi(d_2)'$$

$$\tilde{k}(d_1,d_2) = \phi(d_1)\mathbf{WW'}\phi(d_2)'$$

vidimo da su matrice **WW'** i $\mathbf{U_k U_k'}$ *riječ x riječ* matrice koja određuju veze između različitih riječi. Osim što želimo povezati dokumente u kojima su se pojavile iste riječi, povezujemo i riječi koje se često pojavljuju u sličnim dokumentima što možemo vidjeti u matricama **WW'** za kanonsku korelacijsku analizu i $\mathbf{U_k U_k'}$ za metodu glavnih komponenata. Matrica **WW'** savršeno odvaja riječi različitih kategorija malim vrijednostima i povezuje riječi iste kategorije. Dvije riječi koje se obje nalaze u istoj kategoriji(npr. svjetska-banka) za kanonsku korelacijsku analizu imaju vrijednosti 0.15557, 0.037886, 0.046873, 0.037886, 0.037886,… dok za usporedbu one koje se nalaze u različitim kategorijama(npr. sporazum-banka) imaju vrijednosti -1.6328e-017, 7.8959e-018, -1.1772e-018… U metodi glavnih komponenata razlika između kategorija nije tako jasno određena.

	Svjetska	financijska	kriza	mimoići
Svjetska	0.15557	0.037886	0.046873	0.037886
Financijska	0.037886	0.0092262	0.011415	0.0092262
Kriza	0.046873	0.011415	0.014122	0.011415
Mimioći	0.037886	0.0092262	0.011415	0.0092262
Hrvatska	0.037886	0.0092262	0.011415	0.0092262
Zagrebačka	0.063146	0.015378	0.019025	0.015378
Banka	0.14712	0.035827	0.044325	0.035827
Dobila	0.029429	0.0071666	0.0088666	0.0071666
Veliki	0.029429	0.0071666	0.0088666	0.0071666
Kredit	0.11953	0.029108	0.036012	0.029108
Ebrd-a	0.063146	0.015378	0.019025	0.015378
…	…	…	…	…
Mjesec	-1.6328e-017	7.8959e-018	-1.1772e-018	7.8959e-018
Izborni	-1.6328e-017	7.8959e-018	-1.1772e-018	7.8959e-018
Zakon	-1.6328e-017	7.8959e-018	-1.1772e-018	7.8959e-018

Matrica bliskosti za prvih par riječi (kanonska korelacijska analiza)

	Svjetska	financijska	kriza	mimoići
Svjetska	0.86791	0.14411	0.081873	0.14411
Financijska	0.14411	0.025203	0.012693	0.025203
Kriza	0.081873	0.012693	0.0083587	0.012693
Mimioći	0.14411	0.025203	0.012693	0.025203
Hrvatska	0.14411	0.025203	0.012693	0.025203
Zagrebačka	-0.21098	-0.066344	0.0021886	-0.06634
Banka	0.68845	0.098315	0.076226	0.098315
Dobila	-0.03536	-0.020587	0.0070463	-0.02058
Veliki	-0.03536	-0.020587	0.0070463	-0.02058
Kredit	0.095499	-0.019134	0.033694	-0.01913
Ebrd-a	-0.21098	-0.066344	0.0021886	-0.06634
…	…	…	…	…
Mjesec	-0.07797	-0.010901	-0.008798	-0.01090
Izborni	-0.07797	-0.010901	-0.008798	-0.01090
Zakon	-0.07797	-0.010901	-0.008798	-0.01090

Matrica bliskosti za prvih par riječi (metoda glavnih komponenata)

Različite boje odvajaju riječi po dokumentima, tako se prvih 5 riječi označenih crvenom bojom nalazi u prvom dokumentu, sljedećih 6 u drugom dokumentu itd. Zelenom bojom obilježene su posljednje 3 riječi koje pripadaju zadnjem dokumentu i bitno je napomenuti da pripadaju kategoriji politika za razliku od prva dva dokumenta koja govore o gospodarstvu. Metoda glavnih komponenata nije uspjela povezati sve riječi koje definiraju kategorije. Tako su samo skup riječi svjetska-banka jako povezani jer se pojavljuju u drugim dokumentima zajedno.

Za kanonsku korelacijsku analizu vrijedi: kako su spoj riječi „svjetska banka“ pojavljuje u mnogo dokumenta, a „zagrebačka banka“ je spomenuta u jednom logično je povezivanje svjetska – zagrebačka. Svjetska - dobila ima manju vrijednosti od svjetska - zagrebačka, svjetska - kredit ima veću jer se eksplicitno pojavljuje u jednom od dokumenata.

Kanonska korelacijska analiza je uspjela točno odvojiti riječi, tj. riječi jednog i drugog svojstvenog vektora se uopće ne miješaju i točno definiraju svoje kategorije jer su svi dokumenti jedne kategorije bili međusobno povezani, na hrvatskom ili ako ne na engleskom prijevodu. Metoda glavnih komponenata je s druge strane kao komponente vektora izmiješala riječi jedne i druge kategorije, tako da postoji veza svjetska - zakon jača od svjetska - veliki iako se svjetska i veliki pojavljuju u kategoriji gospodarstvo, a zakon u politici.

Metoda glavnih komponenata naravno ne gleda oznake pojedinih dokumenta nego traži osi najveće varijance te u ovom primjeru možemo zaključiti kako je kanonska korelacijska analiza kompleksnim oznakama(engleskim jezikom) uspjela odrediti bolji potprostor.