SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 5179

Primjena stroja s potpornim vektorima za analizu sentimenta korisničkih recenzija

Dominik Stanojević

Zagreb, lipanj 2017.

Umjesto ove stranice umetnite izvornik Vašeg rada.
Kako biste uklonili ovu stranicu, obrišite naredbu `\izvornik`.

SADRŽAJ

1. Uvod

Klasifikacijski i regresijski problemi jedni su od najvažnijih problema strojnog učenja. Modeli poput linearne i logističke regresije pogodni su za jednostavnije probleme. Zahvaljujući sve većoj dostupnosti podataka i povećanju procesorske moći današnjih računala, pojavljuju se složeniji zadaci za koje navedene metode nisu efikasne.

Pojava složenijih zadataka rezultirala je i pojavom složenijih metoda koje mogu doskočiti istima. Modeli poput slučajnih šuma i modeli iz skupine dubokog učenja u mogućnosti su rješavati i složenije, nelinearne probleme.

Osim navedenih modela, još jedan model koji je sposoban efikasno obraditi nelinearne podatke je **stroj s potpornim vektorima** (engl. *Support Vector Machine*, u nastavku SVM). Koristeći jezgreni trik, stroj s potpornim vektorima uspješno razdvaja linearno nerazdvojive podatke. Iako su temeljne ideje modela predstavljene prije više od pola stoljeća, stroj s potpornim vektorima i danas je jedan od najrobusnijih modela za klasifikaciju i regresiju.

Jedan od zanimljivih problema koji dobro prikazuje robusnost SVM-a je **analiza sentimenta** (engl. *Sentiment Analysis*). Subjektivnost emocija, kontekst te velika količina podataka svakako predstavljaju izazove u rješavanju problema. Koristeći SVM, uz uvjet kvalitetnog pretprocesiranja podataka, mogu se postići zavidni rezultati u polju analize sentimenta.

U radu je predstavljen model stroja s potpornim vektorima te problem analize sentimenta. U poglavlju 2 bit će predstavljen pregled područja, povijest modela stroja s potpornim vektorima te problem analize sentimenta. U poglavlju 3 detaljnije će se obraditi model SVM. Bit će opisana motivacija i interpretacija modela. Nadalje, detaljnije će se pojasniti algoritmi optimizacije modela. U poglavlju 4 formalizirat će se problem analize sentimenta. Prikazat će se postupak pretprocesiranja podataka koji će podatke pretvoriti u oblik razumljiv SVM-u. U petom poglavlju, provest će se eksperiment analize korisničkih recenzija uporabom opisanih metoda. Ukratko će se analizirati dobiveni rezultati. Poglavlje 6 sadrži zaključak i ideje za daljnje istraživanje.

2. Pregled područja

Godina začetka ideje modela stroja s potpornim vektorima može se smatrati 1963. kada sovjetski matematičar Vladimir Vapnik zajedno s Aleksandrom Lernerom izdaje publikaciju u kojoj je opisan algoritam generaliziranog portreta (engl. *Generalized Portrait Algorithm* [22]). Stroj s potpornim vektorima smatra se nelinearnom generalizacijom upravo tog algoritma. Godinu kasnije Aizerman, Rozonoer i Braverman uvode pojam jezgrenih trikova [1]. Corver 1965. godine iznosi ideju optimalno razdvajajućih hiperravnina [5]. 1992. godine objavljuje se rad u kojem se izlaže model stroja s potpornim vektorima koji je veoma blizak današnjem modelu [2]. Jedna od najvažnijih godina za model SVM je 1995. kada je riješen problem razdvajanja linearno nerazdvojivih podataka. Iste godine razvija se model stroja s potpornim vektorima za primjenu na regresijske probleme [4].

U novijoj literaturi, model stroja s potpornim vektorima dobro je opisan u knjizi *Elements Of Statistical Learning* [8] i materijalima s predavanja sveučilišta Stanford [17].

Za razliku od modela stroja s potpornim vektorima koji je star već nekoliko desetljeća, problem analize sentimenta je relativno novi problem. Izraz *analiza sentimenta* prvi put se pojavljuje 2003. godine [16]. Sama ideja analize sentimenta pojavila se nešto ranije [19], [23]. Područje analize sentimenta jedno je od najpopularnijih problema prirodnog jezika te se nova otkrića i ideje javljaju gotovo svakodnevno. Jedna od najboljih uvodnih knjiga za ovaj problem je svakako knjiga *Sentiment Analysis and Opinion Mining* autora Bing Liu [12].

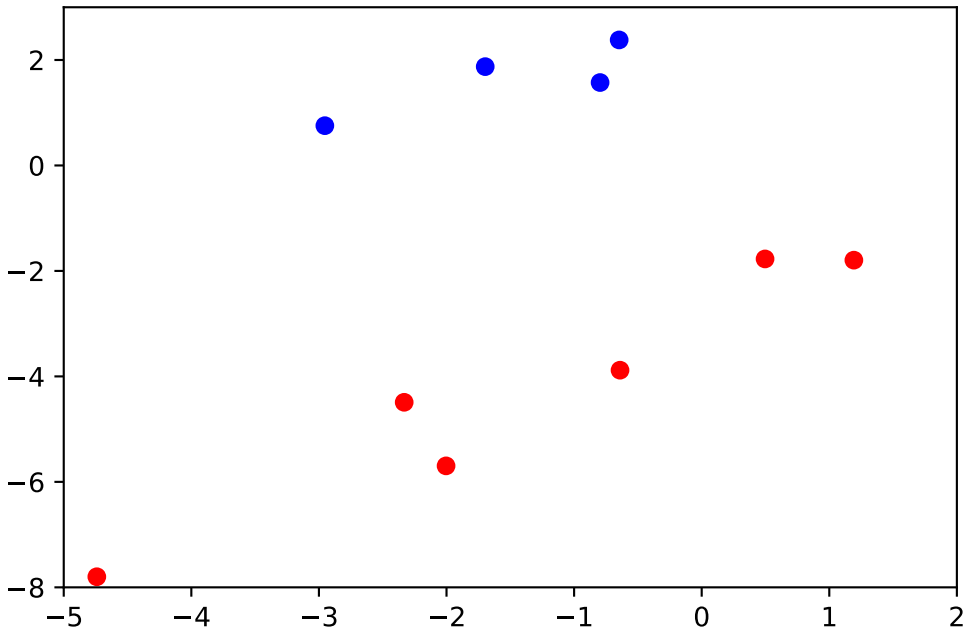
3. Stroj s potpornim vektorima

U ovom poglavlju bit će predstavljen model stroja s potpornim vektorima. Potpoglavlje 3.1 definira pojam klasifikacije i pojašnjava razliku između klasifikacije i regresije. U potpoglavlju 3.2 pojasnit će se ideja razdvajajuće hiperravnine. Potpoglavlje 3.3 predstavit će pojam margine razdvajajuće hiperravnine i njenu važnost u izgradnji klasifikatora. Koristeći ideje iz prethodnih potpoglavlja, potpoglavlje 3.4 definira optimalnu razdvajajuću hiperravninu, metodu koju koristi SVM prilikom klasifikacije podataka. U potpoglavlju 3.5 daje se ideja regularizacije. Ova metoda omogućuje da stroj s potpornim vektorima pronađe optimalnu hiperravninu u slučaju linearno nerazdvojivih podataka. Potpoglavlja 3.6 i 3.7 postavljaju primarni i dualni optimizacijski problem te koristeći dane ideje rješavaju problem optimizacije stroja s potpornim vektorima. Potpoglavlje 3.8 opisuje transformaciju prostora značajki koristeći jezgrene trikove. Jezgreni trikovi su efikasne metode koje omogućuju razdvajanje originalno linearno nerazdvojivih podataka. Potpoglavlje 3.9 posvećeno je metodama višerazredne klasifikacije koje omogućuju klasifikaciju podataka s više od dva razreda. U potpoglavlju 3.10 proširuje se model stroja s potpornim vektorima za primjenu kod regresijskih problema.

3.1. Klasifikacija

Problemi nadziranog učenja uobičajeno se dijele na dvije podskupine - klasifikaciju i regresiju. Neka vektorski prostor *X* dimenzije n, primjerice $\mathbb{R}^n$, predstavlja skup primjeraka. Pojedini primjerak može se zadati vektorom: $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. Ako pojedinom primjerku $\mathbf{x}$ pridružimo oznaku razreda y, tada se govori o **klasifikaciji**. Pojednostavljeno, postupkom klasifikacije određuje se razred kojem određen primjerak $\mathbf{x}$ pripada. Skup svih razreda *C* je konačan, a broj razreda dan je kardinalitetom $|C|$.

Primjer klasifikacijskog problema prikazan je slikom 3.1. Prostor primjeraka je $\mathbb{R}^2$, a skup razreda je dvočlani skup tj. $|C| = 2$. Klasifikacija podataka u dvočlane skupove naziva se **binarna klasifikacija**. Upravo je stroj s potpornim vektorima primjer binarnog klasifikatora, no postoje metode koje pružaju mogućnost višerazredne klasifikacije. Osim gore navedenog primjera, još neki primjeri klasifikacije su otkrivanje neželjene pošte, prepozna-



Slika 3.1: Primjer klasifikacijskog problema

vanje rukopisa, prepoznavanje prometnih znakova, itd.

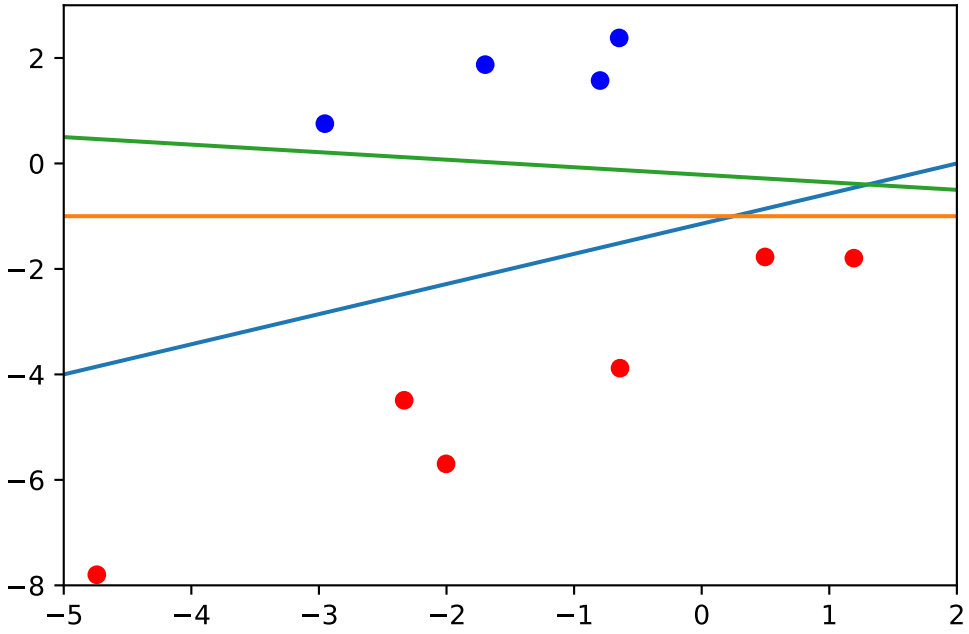
Za razliku od problema klasifikacije u kojem je varijabli y pridružena vrijednost iz konačnog skupa, kod problema regresije primjerku pridružujemo vrijednost iz nekog beskonačnog skupa, primjerice $\mathbb{R}$. Postoji modifikacija stroja s potpornim vektorima koji omogućuje rješavanje regresijskih problema. Primjeri regresije su predviđanje iznosa plaće u ovisnosti o spolu, obrazovanju i sl., predviđanje broja prodanih primjeraka nekog prijenosnog računala, itd.

3.2. Razdvajajuća hiperravnina

Interpretaciju modela stroja s potpornim vektorima potrebno je započeti s pojmom koji nije strogo vezan uz sam model. Primjerice model logističke regresije, iako temeljen na vjerojatnosti, u konačnici pronalazi hiperravninu kojom razdvaja podatke.

Neka je zadan vektorski prostor X dimenzije n, primjerice $\mathbb{R}^n$. Tada je **hiperravnina** definirana kao potprostor dimenzije $n-1$ unutar prostora X. Primjerice, u jednodimenzijskom prostoru hiperravnina je točka, u dvodimenzijskom prostoru hiperravninu predstavlja bilo koji pravac koji leži u ravnini, a u trodimenzijskom prostoru hiperravnina je predstavljena ravninom. Analogno, pojam hiperravnine vrijedi i za prostore većih dimenzija.

Za hiperravninu zadanom izrazom $f(\mathbf{x}) = b + \mathbf{w}^T\mathbf{x} = 0$ vrijede sljedeća svojstva:



Slika 3.2: Razdvajajuće hiperravnine

1. za svaku točku T na hiperravnini vrijedi: $b = -\mathbf{w}^T\mathbf{x}$,
2. jedinični vektor normale je zadan izrazom: $\mathbf{n} = \frac{\mathbf{w}}{\|\mathbf{w}\|}$,
3. udaljenost točke P od hiperravnine iznosi: $d = \frac{|f(\mathbf{x})|}{\|\mathbf{w}\|}$.

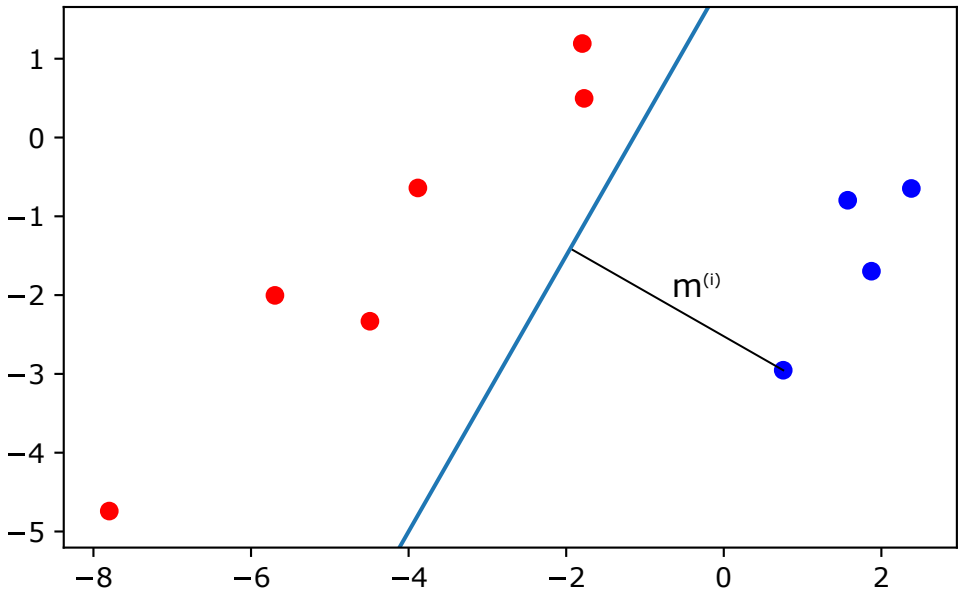
Hiperravnina ovisi o vektoru težina $\mathbf{w}$ i slobodnom članu b. U daljnjem tekstu hiperravnina bit će zadana svojim parametrima, $\langle \mathbf{w}, b \rangle$.

Hiperravnine same po sebi nisu pretjerano interesantne. No, za klasifikaciju interesantan je određen podskup hiperravnina. Hiperravnina koja razdvaja dva razreda podataka naziva se **razdvajajuća hiperravnina.** Uz pretpostavku $y \in \{-1, 1\}$, za razdvajajuću hiperravninu vrijedi:

$$y^{(i)} = sgn(b + \mathbf{w}^T\mathbf{x}^{(i)}), \forall i \tag{3.1}$$

gdje je $\mathbf{x}^{(i)}$ primjer iz skupa podataka, a $y^{(i)}$ je oznaka razreda pridružena primjeru $\mathbf{x}^{(i)}$.

Na slici 3.2 prikazani su primjerci jednaki onima sa slike 3.1. Također, prikazane su i tri razdvajajuće hiperravnine. Valja primijetiti kako je moguće konstruirati beskonačno mnogo razdvajajućih hiperravnina.



Slika 3.3: Margina hiperravnine s obzirom na primjerak $(\mathbf{x}^{(i)}, y^{(i)})$

3.3. Margina razdvajajuće hiperravnine

Za razliku od drugih klasifikatora koji traže bilo koju razdvajajuću hiperravninu kako bi klasificirali podatke, stroj s potpornim vektorima uzima u obzir i udaljenosti primjeraka od hiperravnine. Intuitivno se može zaključiti kako je sigurnije odrediti razred za one primjerke koji su udaljeniji od hiperravnine. Udaljenost primjerka od hiperravnine nazivamo **margina**.

Neka je zadan i-ti primjerak $(\mathbf{x}^{(i)}, y^{(i)})$ gdje je $\mathbf{x}^{(i)}$ vektor značajki, a $y^{(i)}$ pripadajuća oznaka razreda. Također, neka je $\mathbf{r^{(i)}}$ radij-vektor točke T koja se nalazi na hiperravnini i najmanje je udaljena od primjerka. Udaljenost i-tog primjerka od hiperravnine iznosi $m^{(i)}$. Vrijede dvije jednadžbe:

$$\mathbf{r}^{(i)} = \mathbf{x}^{(i)} - y^{(i)} m^{(i)} \frac{\mathbf{w}}{\|\mathbf{w}\|},$$

$$b + \mathbf{w}^T \mathbf{r^{(i)}} = 0.$$

Rješavanjem ovog sustava po $m^{(i)}$ dobiva se margina hiperravnine $\langle \mathbf{w}, b \rangle$ s obzirom na primjerak $(\mathbf{x}^{(i)}, y^{(i)})$:

$$m^{(i)} = \frac{y^{(i)}(b + \mathbf{w}^T \mathbf{x}^{(i)})}{\|\mathbf{w}\|}. \tag{3.2}$$

Potrebno je napomenuti kako $y^{(i)}$ poprima vrijednosti iz skupa $\{-1, 1\}$ stoga nije važno nalazi li se $y^{(i)}$ u brojniku ili u nazivniku.

Na slici 3.3 prikazana je udaljenost primjerka od razdvajajuće hiperravnine. Valja uočiti kako za pozitivne oznake razreda, $y^{(i)} = 1$, vrijednost $b + \mathbf{w}^T\mathbf{x}^{(i)}$ je pozitivna. Analogno, za $y^{(i)} = -1$ vrijednost $b + \mathbf{w}^T\mathbf{x}^{(i)}$ je negativna. Može se zaključiti kako je vrijednost margine za svaki primjerak strogo pozitivna. U slučaju hiperravnine koja ne razdvaja podatke to svojstvo ne vrijedi.

Osim svojstva pozitivnosti, za marginu je zanimljivo i svojstvo otpornosti na skaliranje. Neka je hiperravnina $\langle \mathbf{w}, b \rangle$ skalirana nekim faktorom k. Za marginu $m^{'(i)}$ vrijedi:

$$m^{'(i)} = \frac{y^{(i)}(kb + k\mathbf{w}^T\mathbf{x}^{(i)})}{\|k\mathbf{w}\|} = \frac{y^{(i)}k(b + \mathbf{w}^T\mathbf{x}^{(i)})}{k\|\mathbf{w}\|} = m^{(i)}.$$

Ovo svojstvo omogućuje da duljina vektora težina bude proizvoljna što će se pokazati veoma korisnim kod postavljanja optimizacijskog problema.

Nakon definiranja margine hiperravnine za pojedini primjerak, potrebno je definirati i marginu hiperravnine uzimajući u obzir cijeli skup podataka za učenje. **Margina hiperravnine** u odnosu na skup podataka za učenje je margina onog primjerka koji je najbliži hiperravnini tj.

$$M = \min_i m^{(i)}.$$

3.4. Optimalna razdvajajuća hiperravnina

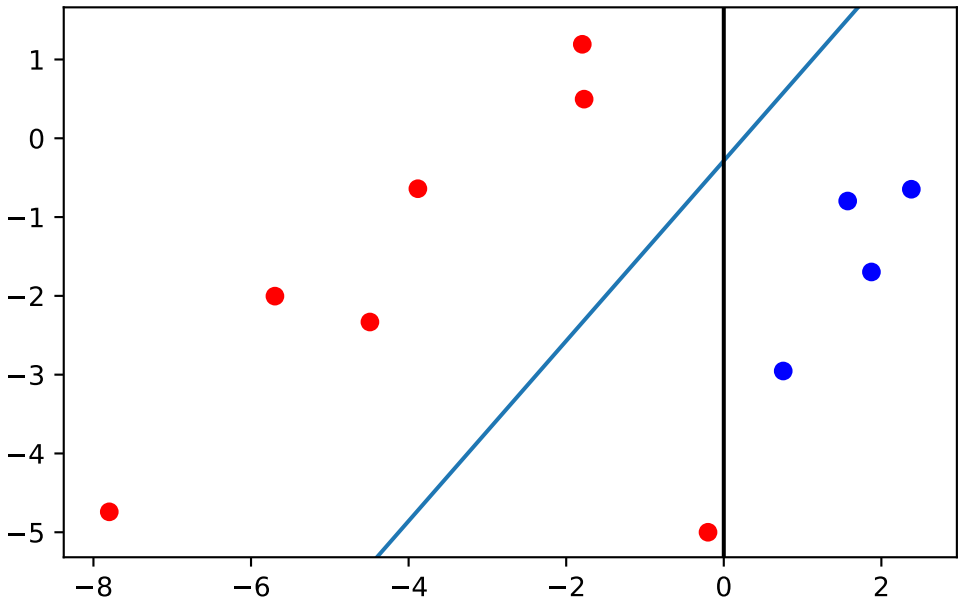
Nakon definicije margine, sljedeći cilj je pronaći razdvajajuću hiperravninu koja najbolje razdvaja podatke. Uzimajući u obzir činjenicu da veća udaljenost primjerka od hiperravnine pruža veću sigurnost za ispravnu klasifikaciju, intuitivno se može zaključiti kako će optimalna razdvajajuća hiperravnina biti ona koja maksimizira marginu hiperravnine s obzirom na skup primjeraka za učenje. **Optimalna razdvajajuća hiperravnina** zadana je sljedećim optimizacijskim problemom:

$$\begin{aligned} &\max_{\mathbf{w},b} && M \\ &\text{s obzirom na} && m^{(i)} \geq M,\ i = 1, \ldots, N \end{aligned} \tag{3.3}$$

Ograničenje $m^{(i)} \geq M$ zahtjeva da svi primjerci iz skupa za učenje budu udaljeni za minimalno M od razdvajajuće hiperravnine. No, izraz 3.2 nije konveksan stoga se redefinira na sljedeći način:

$$y^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq M\|\mathbf{w}\|,\ i = 1, \ldots, N.$$

Nadalje, budući da duljina vektora težina ne utječe na marginu i klasifikaciju, moguće je proizvoljno odabrati njegovu duljinu. Za $\|\mathbf{w}\| = \frac{1}{M}$ cilj optimizacije se svodi na pronalazak maksimuma funkcije $f^*(\mathbf{w}) = \frac{1}{\|\mathbf{w}\|}$. Znajući kako se maksimizacija ove funkcije može svesti na minimizaciju kvadrata norme, može se pisati:



Slika 3.4: Utjecaj stršeće vrijednosti na odabir hiperravnine

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{s obzirom na} \quad & y^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1, \ i = 1, \ldots, N. \end{aligned} \tag{3.4}$$

Koristeći nekoliko različitih transformacija, problem je redefiniran na jednostavniji način. Budući da je dobivena konveksna kvadratna funkcija cilja uz linearne uvjete ovaj problem je rješiv metodama kvadratnog programiranja. Kasnije će se u radu ovaj optimizacijski problem dodatno transformirati kako bi se iskoristili algoritmi koji rješavaju problem efikasnije od općenitih programskih paketa koji koji se koriste za rješavanje ovakvih optimizacijskih problema.

3.5. Regularizacija

U dosadašnjem radu klasificiralo se linearno razdvojive podatke. Međutim, podaci najčešće nisu linearno razdvojivi ili optimalna razdvajajuća hiperravnina nije najbolji klasifikator budući da nije otporna na stršeće vrijednosti. Slučaj kada stršeća vrijednost utječe na izbor hiperravnine prikazan je na slici 3.4. Vidljivo je kako hiperravnina nacrtana plavom bojom dobro razdvaja podatke osim stršeće vrijednosti označene točkom T. Druga hiperravnina označena crnom bojom, iako je razdvajajuća, nije najsretniji izbor za klasifikaciju.

Optimizacijski problem je napisan tako da bira razdvajajuću hiperravninu u bilo kojem

slučaju, čak i kada ona nije dobar izbor. Kao rješenje ovog problema uvodi se metoda regularizacije. **Regularizacija** je metoda kojom se sprječava pretreniranost modela koristeći funkciju kazne. Iako je hiperravnina nacrtana crnom bojom na slici 3.4 dobro prilagođena danim primjercima, kod klasificiranja novih primjerka neće dati dobre rezultate kao hiperravnina označena plavom bojom. Ovime se kažnjavaju modeli koji pretjerano slijede sve primjerke, ne vodeći računa o generalnoj strukturi podataka.

Valja navesti dvije najosnovnije funkcije kazne koje se koriste za regularizaciju SVM modela. Funkcija kazne kod L1-SVM modela zadana je izrazom $\max(1 - y^{(i)}\mathbf{w}^T\mathbf{x}^{(\mathbf{i})}, 0)$, a kod L2-SVM funkcija kazne je zadana izrazom $\max(1 - y^{(i)}\mathbf{w}^T\mathbf{x}^{(\mathbf{i})}, 0)^2$. U nastavku će se koristiti L1-SVM.

Nakon uvođenja regularizacije, optimizacijski problem može se redefinirati na sljedeći način:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i \\ \text{s obzirom na} \quad & y^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1 - \xi_i,\ i = 1, \ldots, N \\ & \xi_i \geq 0,\ i = 1, \ldots, N. \end{aligned} \tag{3.5}$$

Valja obratiti pažnju na regularizacijski parametar C čiji iznos izražava spremnost na pogrešnu klasifikaciju primjerka iz skupa za učenje. Za velike iznose parametra C, optimizacija će inzistirati na ispravnoj klasifikaciji primjeraka iz skupa za učenje, makar pod cijenu pretreniranosti. Za manje iznose, optimizacija će pogrešno klasificirati neke od primjeraka iz skupa za učenje kako bi se pronašla hiperravnina koja dobro opisuje generalnu strukturu podataka.

3.6. Primarni i dualni optimizacijski problem

Nakon što je optimizacijski problem postavljen, vrijedi ga pokušati i riješiti. Postupak koji će se koristiti u rješavanju ovog problema je metoda Lagrangeovih multiplikatora. Neka je zadan **primarni** optimizacijski problem oblika:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s obzirom na} \quad & g_i(x) \leq 0,\ i = 1, \ldots, m \\ & h_i(x) = 0,\ i = 1, \ldots, n. \end{aligned} \tag{3.6}$$

Za dani optimizacijski problem, može se postaviti Lagrangeova funkcija oblika:

$$\mathcal{L}(x, \alpha, \beta) = f(x) + \sum_{i=1}^{m}\alpha_i g_i(x) + \sum_{i=1}^{n}\beta_i h_i(x) \tag{3.7}$$

gdje su α_i i β_i Lagrangeovi multiplikatori. Budući da se traži ekstrem funkcije, Lagrangeova funkcija se parcijalno derivira po varijablama x, α i β te se parcijalne derivacije izjednače s nulom.

Nakon definiranja Lagrangeove funkcije, valja definirati i još jednu vrijednost:

$$\theta_{\mathcal{P}}(x) = \max_{\alpha,\beta,\alpha_i \geq 0} \quad \mathcal{L}(x, \alpha, \beta). \tag{3.8}$$

Može se pokazati kako je iznos $\theta_{\mathcal{P}}(x)$ jednak vrijednosti funkcije $f(x)$ u slučaju kada su svi uvjeti zadovoljeni. U slučaju da neki od uvjeta nisu zadovoljeni, $\theta_{\mathcal{P}}(x)$ iznosi nula. Primjerice neka je $g_i(x) > 0$. Tada se može izabrati α_i za koji desna strana jednadžbe 3.6 iznosi ∞. Analogno vrijedi i za $h_i(x) \neq 0$.

Minimizacijom vrijednosti $\theta_{\mathcal{P}}(x)$ dobije se problem jednak primarnom. Vrijednost primarnog problema dana je izrazom:

$$p = \min_{x} \quad \theta_{\mathcal{P}}(x). \tag{3.9}$$

Za pronalazak $\theta_{\mathcal{P}}(x)$ interesantan je bio maksimum Lagrangeove funkcije u odnosu na parametre α i β. Umjesto toga, problem se može modificirati da umjesto traženja maksimuma u odnosu na parametre α i β, traži se minimum u odnosu na x. **Dualni** problem definira se na sljedeći način:

$$\max_{\alpha,\beta,\alpha_i \geq 0} \theta_{\mathcal{D}}(x) = \max_{\alpha,\beta,\alpha_i \geq 0} \min_{x} \mathcal{L}(x, \alpha, \beta). \tag{3.10}$$

Valja primijetiti kako je dualni problem jednak primarnom, uz zamjenu poretka funkcije min i funkcije max. Vrijednost $\theta_{\mathcal{D}}(x)$ je dana izrazom:

$$d = \max_{\alpha,\beta,\alpha_i \geq 0} \quad \theta_{\mathcal{D}}(x). \tag{3.11}$$

Uzimajući u obzir odnos između funkcija max i min jasno je da vrijedi relacija $d \leq p$. No, posebno su interesantni slučajevi gdje su vrijednosti primarnog i dualnog problema jednake.

Kako bi vrijednost primarnog i dualnog problema bila jednaka, potrebno je postaviti neka ograničenja na funkcije f, g_i i h_i. Neka je f konveksna funkcija. Nadalje, neka su g_i konveksne funkcije i neka za svaku od njih vrijedi $g_i(x) < 0$. Također, neka je h_i linearna funkcija. Ako su ti uvjeti zadovoljeni, tada postoje α, β i x za koje vrijedi sljedeća jednakost:

$$p = d = \mathcal{L}(x, \alpha, \beta). \tag{3.12}$$

Osim gornje jednakosti, za parametre vrijede i Karush-Kuhn-Tucker (KKT) uvjeti:

$$\frac{\partial \mathcal{L}(x, \alpha, \beta)}{\partial x_i} = 0, \ i = 1, \ldots, p \tag{3.13}$$

$$\frac{\partial \mathcal{L}(x, \alpha, \beta)}{\partial \beta_i} = 0,\ i = 1, \ldots, n \tag{3.14}$$

$$\alpha_i g_i(x) = 0,\ i = 1, \ldots, m \tag{3.15}$$

$$g_i(x) \leq 0,\ i = 1, \ldots, m \tag{3.16}$$

$$\alpha_i \geq 0,\ i = 1, \ldots, m \tag{3.17}$$

Valja napomenuti kako se kod optimizacije stroja s potpornim vektorima u pravilu koristi dualni problem. Također, za SVM interesantan je uvjet zadan jednakošću 3.15. U pravilu $\alpha_i \neq 0$ iz čega slijedi $g_i(x) = 0$. Taj uvjet je ključan za pronalazak potpornih vektora.

3.7. Optimizacija stroja s potpornim vektorima

Primjenjujući prethodno potpoglavlje na optimizacijski problem iz potpoglavlja 3.5, primarna Lagrangeova funkcija može se zapisati na sljedeći način:

$$\mathcal{L}(\mathbf{w}, b, \xi, \alpha, \mu) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i(y^{(i)}(\mathbf{w}^T\mathbf{x^{(i)}} + b) - (1 - \xi_i)) - \sum_{i=1}^{N}\mu_i\xi_i. \tag{3.18}$$

Vidljivo je kako je ograničenje $y^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1 - \xi_i$ zapisano kao:

$$g_i(\mathbf{w}, b) = -(y^{(i)}(\mathbf{w}^T\mathbf{x^{(i)}} + b) - (1 - \xi_i)) \leq 0.$$

Gore navedeno ograničenje ima zanimljivo svojstvo. U slučaju da vrijedi $g_i(\mathbf{w}, b) = 0$ za neki primjerak $(x^{(i)}, y^{(i)})$ tada se taj primjerak naziva **potpornim vektorom**. Može se pretpostaviti kako je potpornih vektora relativno malo u odnosu na cijeli skup primjeraka za učenje. Vodeći se tom pretpostavkom te uvjetom 3.15 može se zaključiti kako samo za potporne vektore vrijedi $\alpha_i \neq 0$ dok za ostale primjerke vrijedi $\alpha_i = 0$.

Na slici 3.5 prikazana je maksimalna razdvajajuća hiperravnina te pravci paralelni s istom na kojima leže potporni vektori. Valja primijetiti kako su pravci jednako udaljeni od hiperravnine za iznos M.

Primarni problem dan izrazom 3.18 može se derivirati po $\mathbf{w}, b$ i ξ_i te izjednačiti s nulom:

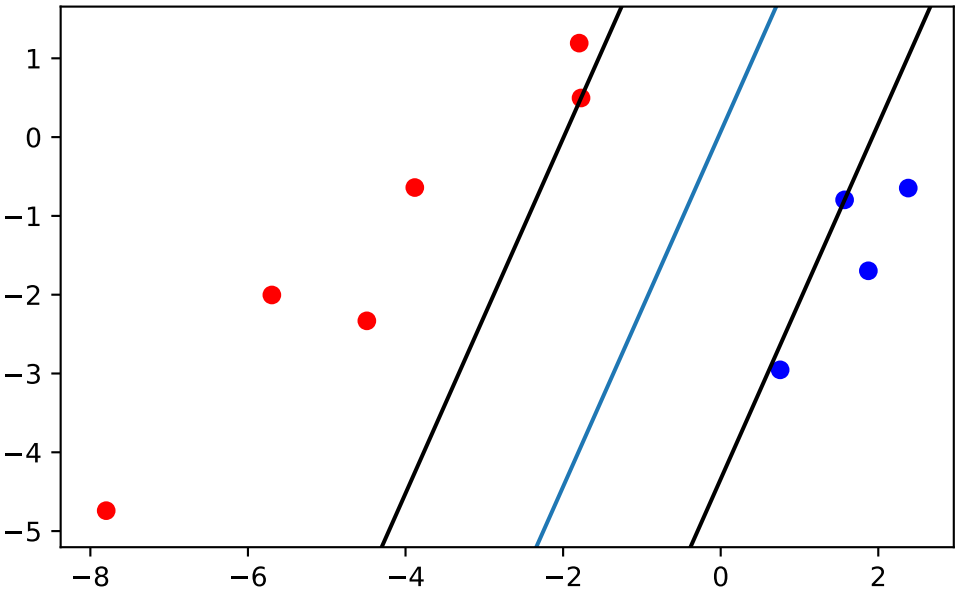
$$\mathbf{w} = \sum_{i=1}^{N}\alpha_i y^{(i)} x^{(i)} \tag{3.19}$$

$$0 = \sum_{i=1}^{N}\alpha_i y^{(i)} \tag{3.20}$$

$$\alpha_i = C - \mu_i, \forall i. \tag{3.21}$$

Ubacivanjem gornjih izraza u 3.18, dolazi se do dualne Lagrangeove funkcije:

$$\mathcal{L}(\alpha) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y^{(i)} y^{(j)} x^{(i)T} x^{(j)}. \tag{3.22}$$



Slika 3.5: Maksimalna razdvajajuća hiperravnina te pravci na kojima leže potporni vektori

Uz ograničenje 3.20 te ograničenje $0 \leq \alpha_i \leq C$ dualni problem može se zapisati na sljedeći način:

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y^{(i)} y^{(j)} x^{(i)T} x^{(j)} \\ \text{s obzirom na} \quad & 0 \leq \alpha_i \leq C, \ i = 1, \ldots, N \\ & \sum_{i=1}^{N} \alpha_i y^{(i)} = 0. \end{aligned} \tag{3.23}$$

Ako se pogleda uvjet 3.20 može se zaključiti kako samo potporni vektori utječu na odabir hiperravnine. U slučaju linearno nerazdvojivih podataka za potporne vektore koji se nalaze na margini, Lagrangeov multiplikator iznosi $0 < \alpha_i < C$. U slučaju da se potporni vektor ne nalazi na margini, vrijednost pripadajućeg multiplikatora iznosi C.

3.8. Jezgreni trikovi

Dosad je optimizacija stroja s potpornim vektorima mogla pronaći razdvajajuću hiperravninu jedino u slučaju linearno razdvojivih podataka. Želja je izgraditi klasifikator koji će moći ispravno klasificirati podatke koji su nisu linearno razdvojivi.

Kod metoda poput logističke regresije svaki vektor značajki može se, koristeći mapiranje značajki, preslikati u neki drugi vektor značajki. Neka je $\mathbf{x} = (x_1, \ldots, x_p)$ vektor značajki. Radi jednostavnosti neka postoji samo jedna značajka tj. $\mathbf{x} = (x_1)$. Neka funkcija ϕ vrši polinomijalno mapiranje značajki do trećeg stupnja tj. $\phi(x) = (x, x^2, x^3)$. Tada vektor

značajki $\mathbf{x}$ može se preslikati u $\mathbf{x}' = \phi(x_1) = (x_1, x_1^2, x_1^3)$. Ova ideja može se iskoristiti i kod stroja s potpornim vektorima. Svaku pojavu vektora značajki $\mathbf{x}$ moguće je zamijeniti funkcijom $\phi(\mathbf{x})$.

Valja se na trenutak vratiti na dualnu Lagrangeovu funkciju zadanu jednakošću 3.22. Može se primijetiti kako dio funkcije zadan izrazom $x^{(i)T}x^{(j)}$ je zapravo skalarni umnožak vektora značajki dvaju primjera. Nadalje će se pisati $\langle x^{(i)}, x^{(j)} \rangle$. Vektore značajki moguće je preslikati u neke nove vektore značajki koristeći funkciju ϕ. Vrijedi:

$$\mathcal{L}(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y^{(i)} y^{(j)} \langle \phi(x^{(i)}), \phi(x^{(j)}) \rangle.$$

Primjerak $(x^{(i)}, y^{(i)})$ tada se klasificira na sljedeći način:

$$\hat{y} = sgn(\sum_{i=1}^{N} \alpha_i y^{(i)} \langle \phi(x), \phi(x^{(i)}) \rangle + b).$$

Vidljivo je kako i dualni problem i funkcija klasifikacije koriste funkciju oblika $\langle \phi(x), \phi(x') \rangle$, a ne $\phi(x)$. Može se definirati **jezgrena funkcija** oblika:

$$K(x, x') = \langle \phi(x), \phi(x') \rangle. \tag{3.24}$$

Zamjena skalarnog umnoška vektora x i x' s jezgrenom funkcijom $K(x, x')$ naziva se **jezgreni trik**. Najčešće korištene jezgrene funkcije su radijalna bazna funkcija (RBF), polinomijalna i sigmoidalna jezgrena funkcija.

Na slici 3.6 prikazana je kalsifikacija podataka bez korištenja jezgrenog trika i s korištenjem radijalne bazne funkcije. Jasno je kako linarni model nije dovoljno moćan klasificirati ovaj skup linearno nerazdvojivih podataka. Radijalni bazni model je puno uspješniji u tom zadatku.

Radijalna bazna funkcija korištena u prethodnom primjeru ima oblik:

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

gdje je σ parametar funkcije. Sigmoidalna jezgrena funkcija dana je oblikom:

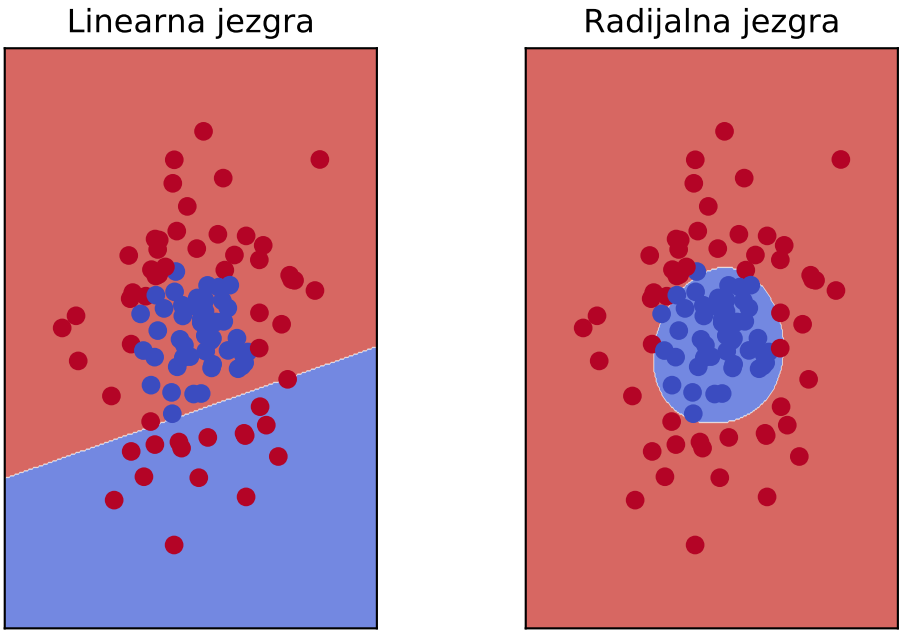
$$K(\mathbf{x}, \mathbf{x}') = \tanh(\gamma \mathbf{x}^T \mathbf{x}' + r)$$

gdje su γ i r parametri funkcije. Polinomijalna jezgrena funkcija zadana je jednadžbom:

$$K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + k)^d,$$

gdje je k proizvoljna konstanta, a d stupanj polinoma. Na primjer, zadan je vektor značajki $\mathbf{x} = (x_1, x_2)$. Neka je $k = 1$ i $d = 2$. Jezgrena funkcija tada iznosi:

$$K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + 1)^2 = 1 + 2x_1 x_1' + 2x_2 x_2' + (x_1 x_1')^2 + (x_2 x_2')^2 + 2x_1 x_1' x_2 x_2'.$$



Slika 3.6: Klasifikacija podataka bez jezgrenog trika i s radijalim jezgrenim trikom

Vidljivo je kako smo vektor značajki s dva svojstva mapirali na vektor značajki sa šest svojstava. Općenito, polinomijalna jezgrena funkcija d-tog reda, preslika vektor značajki duljine p u vektor značajki duljine $\binom{n+d}{d}$. Iz povećanja dimenzionalnosti proizlazi i jedan od problema jezgrenih funkcija. Naime, neka su podaci linearno razdvojivi u slučaju prostora koji sadrži interakcijsku značajku $x_1 x_1'$. Tada je moguće, koristeći gore navedenu polinomijalnu funkciju drugog stupnja, pronaći razdvajajuću hiperravninu. No, uz pronalazak težine za tu značajku optimizacija traži i težine za ostalih pet značajki. Ovaj problem s povećanjem reda jezgrene funkcije postaje sve izraženiji, pogotovo ako je inicijalni prostor svih primjeraka velike dimenzije.

3.9. Višerazredna klasifikacija

Stroj s potpornim vektorima je binarni klasifikator te kao takav nije u mogućnosti klasificirati primjerke u više od dva razreda. To ograničenje je intuitivno budući da hiperravnina dijeli prostor u dva potprostora. Za rješenje ovog problema nude se dvije često korištene metode višerazredne klasifikacije.

Prva metoda višerazredne klasifikacije je metoda "jedan-naspram-ostalih" (engl. *one-vs-all*). Koristeći ovu metodu gradi se sustav klasifikatora. Ako je zadano N različitih razreda, sustav će biti izgrađen od $N - 1$ klasifikatora. Ovom metodom problem se razbija u više

binarnih klasifikacija. Svaki stroj vrši klasifikaciju za pojedini razred. Vrijednost 1 označava pripadnost tom razredu dok vrijednost -1 kaže kako primjerak ne pripada razredu. U slučaju da svih $N-1$ klasifikatora daju negativan izlaz, tada primjerak pripada zadnjem, N-tom razredu. Problem kod ove metode je mogućnost pozitivnog izlaza za više klasifikatora. Tada sustav nije u mogućnosti odrediti razred kojemu primjerak pripada.

Druga metoda višerazredne klasifikacije je metoda "jedan-naspram-jedan" (engl. *one-vs-one*). Kod ove metode također se gradi sustav klasifikatora, ali kod ove metoda jedan klasifikator vrši usporedbu između dvaju razreda. Ukupan broj izgrađenih klasifikatora iznosi $\frac{N(N-1)}{2}$. Očito je kako je ova metoda vremenski složenija od "jedan-naspram-ostalih" metode. No, ova metoda je robusnija u slučaju linearno nerazdvojivih podataka te je robusnija na gore navedeni problem koji se javlja kod "jedan-naspram-ostalih" metode. Sustav radi na principu glasanja. Prilikom klasifikacije, svaki klasifikator daje glas nekom od relevantnih razreda. Primjerak pripada nekom razredu u slučaju da je taj razred dobio najviše glasova.

3.10. Primjena SVM-a kod regresijskih problema

Stroj s potpornim vektorima je primarno binarni klasifikator. No, Drucker je 1997. godine predložio proširenje stroja s potpornim vektorima na probleme regresije [7].

Neka je zadan linearan model oblika:

$$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b. \tag{3.25}$$

Moguće je zapisati funkciju optimizacije na sljedeći način:

$$f(\mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N} L(y^{(i)} - f(\mathbf{x}^{(i)})) \tag{3.26}$$

gdje je L funkcija kazne, a C regularizacijski parametar. Funkciju kazne L moguće je izabrati na nekoliko načina. Neka je d razlika između dane vrijednosti $y^{(i)}$ i vrijednosti izračunate modelom $f(\mathbf{x}^{(i)})$ za neki primjerak $(\mathbf{x}^{(i)}, y^{(i)})$. U slučaju da je greška manja od neke proizvoljne vrijednosti ϵ tada taj primjerak ne pridonosi ukupnoj grešci. U slučaju pogreške veće od ϵ, primjerak pridonosi ukupnoj grešci s $|d| - \epsilon$. Uspoređujući ovu funkciju kazne s načinom rada klasifikatora može se ustvrditi sličnost. Za primjerke koji su jako udaljeni od hiperravnine, klasifikator je veoma siguran da su oni ispravno klasificirani te ne utječu na optimizaciju, za razliku od potpornih vektora. Kod regresije, primjerci koji su relativno blizu svojoj očekivanoj vrijednosti ne pridonose povećanju iznosa kazne. Može se pisati:

$$L(d) = \begin{cases} 0 & |d| < \epsilon \\ |d| - \epsilon & \text{inače.} \end{cases} \tag{3.27}$$

Koristeći gore navedene podatke, može se napisati optimizacijska funkcija:

$$f(\alpha, \alpha') = \epsilon \sum_{i=1}^{N} (\alpha_i' + \alpha_i) - \sum_{i=1}^{N} y^{(i)} (\alpha_i' - \alpha_i) + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\alpha_i' - \alpha_i)(\alpha_j' - \alpha_j) \langle x^{(i)}, x^{(j)} \rangle$$

i pripadajući optimizacijski problem:

$$\begin{aligned} \min_{\alpha, \alpha'} \quad & f(\alpha, \alpha') \\ \text{s obzirom na} \quad & \alpha_i \geq 0, \\ & \sum_{i=1}^{N} (\alpha_i' - \alpha_i) = 0, \\ & \alpha_i' \leq C, \\ & \alpha_i \alpha_i' = 0. \end{aligned} \tag{3.28}$$

U ovom slučaju primjerci za koje vrijedi $\alpha_i' - \alpha_i \neq 0$ nazivaju se potpornim vektorima. Metode koje se koriste za optimizaciju klasifikacijskog stroja mogu se koristiti i za optimizaciju regresijskog stroja.

4. Analiza sentimenta

Ovo poglavlje je posvećeno problemu analize sentimenta. Potpoglavlje 4.1 iznosi definiciju problema analize sentimenta. U potpoglavlju 4.2 pojašnjeni su tipovi mišljenja u analizi sentimenta. Potpoglavlje 4.3 daje pregled razina u analizi sentimenta, a potpoglavlje 4.4 fokusira se na razinu dokumenta. Zadnje potpoglavlje 4.5 daje pregled problema s kojima se susreće prilikom analize sentimenta.

4.1. Definicija

Analiza sentimenta (engl. *Sentiment Analysis*) je područje posvećeno prikupljanju, obradi i analizi subjektivnih informacija, posebice stavova. Subjektivne informacije podrazumijevaju kratkoročna stanja poput emocija i raspoloženja, dugoročna stanja poput stavova pa čak i psihičkih osobina. S druge strane, objektivne rečenice izriču činjenice. Primjerice objektivna rečenica *"Danas je sunčano."* izražava činjenicu te se iz nje ne mogu iščitati emocije, stavovi i slično. Rečenica poput *"Volim sunčan dan."* govori kako izvor mišljenja ima pozitivan stav prema danima koji sunčani te je stoga ova rečenica subjektivna. Moguće je da subjektivna rečenica ne izriče stav. Rečenica *"Mislim da će sutra biti sunčan dan."* ne predstavlja nikakvu činjenicu, već subjektivni dojam, no ne izriče nikakav sentiment.

Najčešće promatrana stanja su stavovi pojedinca koji predstavljaju enormni izvor korisnih informacija za zainteresirani subjekt. Koristeći ove informacije, političari mogu osvajati i gubiti izbore, poduzeća plasirati svoje usluge i proizvode, a potrošači informirati se o istima. Uporabom metoda obrade prirodnog teksta i metoda strojnog učenja, cilj je automatizirati obradu i analizu subjektivnih informacija. Upravo subjektivnost čini ovo područje teškim i istovremeno zanimljivim.

Problem analize sentimenta može se zadati i na formalniji način. Neka je dana uređena petorka (o, k, i, s, v) kojom se može definirati sentiment [12]. Varijabla o predstavlja promatrani objekt. Primjerice, ako se analizira kvaliteta mobilnih uređaja, promatrani objekt upravo bi bili mobilni uređaji. Varijablom k dana je karakteristika promatranog objekta. U slučaju mobilnih uređaja to bi mogla biti kamera, ekran, zvuk, itd. Ako se analizira uređaj u cjelini, tada ova varijabla može biti zanemarena. Varijabla i predstavlja izvor mišljenja,

najčešće korisnika neke usluge ili proizvoda. Ključni dio problema, sentiment, dan je varijablom s. Sentiment predstavlja mišljenje korisnika i o nekoj karakteristici k proizvoda o. Mišljenja, stavovi i emocije mijenjaju se s vremenom pa je važno uključiti i vrijeme u sam problem. Vrijeme je predstavljeno varijablom v.

Evo i jednog primjera korisničke recenzije operacijskog sutava[1]:
Užasan operacijski sustav. Osim osnovnih alata, upakiran je i s gomilu programa koji samo usporavaju rad sustava. Nadam se da korisnici nisu ljubitelji privatnosti jer ju s ovim operacijskim sustavom sigurno neće imati. Jedina pozitivna opcija je uvođenje radnih sati tako da me sustav rjeđe maltretira s ponovnim pokretanjem. Savjet: potražite alternativu. Marko, 2016.

Nakon čitanja korisničke recenzije, jasno je kako se ovdje radi o negativnoj recenziji proizvoda. Prva rečenica recenzije je najbitnija za određivanje sentimenta prema cijelom proizvodu. Izvor mišljenja, korisnik Marko, ocjenjuje cijeli sustav *užasnim*. Izbor pridjeva, u ovom slučaju *užasan*, sigurno utječe na intenzitet sentimenta. Kad bi se umjesto navedenog pridjeva našao pridjev *loš* i dalje bi izjava bila klasificirana kao negativna, no s puno manjim intenzitetom u odnosu na izjavu koja sadrži pridjev *užasan*. Valja napomenuti kako kontekst također utječe na pozitivnost sentimenta. U ovom primjeru, riječ *užasan* je nositelj negativnog sentimenta. No, u izrazu *užasno pametan* riječ *užasno* ističe poztivinu karakteristiku objekta. Nakon prve rečenice, slijede dvije rečenice negativnog sentimenta usmjerene na pojedine karakteristike sustava. Četvrta rečenica navodi jednu pozitivnu karakteristiku sustava. U slučaju analize sentimenta na razini cijele recenzije ova rečenica uvodi šum te predstavlja jedan od problema u analizi sentimenta. Iz zadnje rečenice, iako svakako manjeg intenziteta od ostalih, također se može iščitati negativan stav prema proizvodu. Valja napomenuti kako je ova recenzija napisana 2016. godine te upravo godina predstavlja vremensku komponentu problema. Moguće je da se s vremenom stav korisnika promijeni. Tada ova recenzija više neće biti aktualna. Ukupno se može iščitati četiri sentimenta:

(Operacijski sustav, "općenito", Marko, negativan, 2016.)
(Operacijski sustav, programski paket, Marko, negativan, 2016.)
(Operacijski sustav, razina privatnosti, Marko, negativan, 2016.)
(Operacijski sustav, vrijeme osvježavanja sustava, Marko, pozitivan, 2016.)

4.2. Tipovi mišljenja

Postoji nekoliko podjela mišljenja. Najčešće korišten tip mišljenja u analizi sentimenta je regularno mišljenje. Uz regularno mišljenje, izdvaja se i komparativno mišljenje. Alternativna

[1]Napomena: Iako nije dano ime operacijskog sustava, čitatelj može s jednostavnošću zaključiti o kojem se operacijskom sustavu radi

podjela je podjela mišljenja na implicitno i eksplicitno mišljenje.

1. **Regularno mišljenje** je najčešći tip mišljenja te unutar njega postoje dva podtipa mišljenja. Izravno mišljenje je mišljenje u kojem korisnik jasno izražava svoje mišljenje. Rečenica *Volim kolače.* je primjer izravnog mišljenja. S druge strane, neizravno mišljenje karakterizira neizravno iskazivanje mišljenja o objektu. Najčešće izvor mišljenja iskazuje osjećaj prouzročen utjecajem objekta na njega. Primjer neizravnog mišljenja je rečenica *"Konzumiranje kolača povećava moje zadovoljstvo."*. U istraživanju, naglasak je stavljen na izravno mišljenje budući da ga je lakše uočiti i analizirati.

2. **Komparativno mišljenje** iskazuje odnos između dva objekta ili karakteristike. Često se kod komparativnog mišljenja koristi stupnjevanje pridjeva kako bi se definirao odnos. Rečenica *"Iako volim voćne kolače, draži su mi čokoladni."* je primjer komparativnog mišljenja. U ovoj rečenici iskazan je pozitivan sentiment prema voćnim i čokoladnim kolačima, no očito je kako su mišljenja različitih intenziteta.

3. **Eksplicitno mišljenje** je tip mišljenja iz kojeg je direktno moguće iščitati regularno ili komparativno mišljenje. Rečenica *"Volim kolače."* primjer je eksplicitnog mišljenja.

4. **Implicitno mišljenje** je objektivna izjava koja implicira regularno ili komparativno mišljenje. Ovakav tip mišljenja kroz činjenice ističe sentiment korisnika. Primjer implicitnog mišljenja je rečenica *"Baterija na mobitelu ne traje dovoljno dugo."*.

Može se zaključiti kako je tanka granica između ovih podjela, posebno u slučaju izravnih i neizravnih mišljenja te eksplicitnih i implicitnih mišljenja. U pravilu fokus je na izravnim i eksplicitnim mišljenjima budući da je detekcija i analiza lakša u odnosu na neizravna i implicitna mišljenja.

4.3. Razine analize sentimenta

Sentiment korisnika može se proučavati na više razina. U pravilu sentiment korisnika proučava se na sljedeće tri razine:

1. **Analiza na razini dokumenta**. Cilj ove razine je pronaći generalni sentiment u sklopu cijelog teksta. Provodi se najčešće kada zainteresirane strane ne zanima sentiment neke karakteristike proizvoda ili usluge već generalan stav. Također, pretpostavlja se da korisnik ocjenjuje samo jedan proizvod budući da iz njega nije moguće iščitati mišljenje o više proizvoda. Primjer analize na razini dokumenta je analiza korisničkih recenzija filmova [19].

2. **Analiza na razini rečenica**. Cilj analize na razini rečenica je odrediti pozitivnost, neutralnost ili negativnost svake rečenice. Za ovu razinu analize važno je razlikovati objektivne i subjektivne rečenice [23]. No, postoje iznimke kada rečenice, iako objektivne, nose sentiment. Primjerice, *"Kapacitet baterije ovoga mobitela drastično opada već nakon dva mjeseca."* predstavlja objektivnu rečenicu, ali i implicitno izlaže nezadovoljstvo korisnika.

3. **Analiza na razini karakteristika**. Najdetaljnija i najkompleksnija analiza sentimenta vrši se na razini karakteristika. Pretpostavka je da svako mišljenje sadrži sentiment i objekt te da mišljenje nije striktno vezano uz formu tj. uz rečenicu, paragraf, cijeli dokument [10]. Na primjer, rečenica *"Zadovoljan sam s mobitelom marke A, no nije kao onaj marke B."* izražava pozitivan stav prema oba mobitela, no intenzitet sentimenta nije jednak prema oba mobitela. Vidljivo je kako postoji dva sentimenta i dva različita objekta što se ne može utvrditi na razini rečenice, a pogotovo ne na razini dokumenta. Rečenica *"Auto je brz, ali troši previše goriva."* izražava sentiment o jednom objektu, autu, ali za više njegovih karakteristika. Upravo je pronalazak karakteristika i objekata veoma složen problem koji čini ovu razinu još kompleksnijom u odnosu na razine dokumenta i rečenice.

Iako su sve tri razine analize sentimenta zanimljive i složene, u ovom radu fokus je stavljen na razinu dokumenta.

4.4. Sentiment na razini dokumenta

Analiza sentimenta na razini dokumenta najčešće je obrađivan problem u ovom području. Cilj analize na razini dokumenta je pronaći sentiment osnovne informacijske jedinice, u ovom slučaju cijelog dokumenta. U pravilu ova razina se najčešće bavi sentimentom korisnika prema različitim proizvodima i uslugama. Budući da se analizom dolazi do generalnog sentimenta, ova razina analize nije pogodna za pronalazak sentimenta u odnosu na karakteristike proizvoda ili sentimenta za više različitih proizvoda.

Problem analize sentimenta na razini dokumenta može se definirati i na formalniji način. Cilj analize sentimenta na razini dokumenta je pronaći sentiment s izvora mišljenja i o objektu o pri čemu se ne analiziraju karakteristike objekta [12]. Problem se predstavlja uređenom petorkom: $(\_, \text{"općenito"}, \_, s, \_)$. Pretpostavka je da su vrijeme mišljenja, izvor mišljenja te objekt znani ili nebitni za analizu. Prilikom analize sentimenta na ovoj razini ključna je pretpostavka kako u analiziranom dokumentu jedan izvor mišljenja izlaže stav o jednom objektu [11]. U pravilu, recenzije proizvoda i usluga zadovoljavaju ovu pretpos-

tavku. S druge strane, transkripti političkih rasprava su primjer dokumenata koji ne zadovoljavaju pretpostavku.

U srži problema analize sentimenta su značajke teksta i izraza unutar njega. Postoji nekoliko značajki koje se uzimaju u obzir prilikom analize sentimenta:

1. **Subjektivne riječi i izrazi** su ključni za ispravnu klasifikaciju sentimenta. Riječi poput *sretan*, *razočaran* i *odličan* uvelike određuju sentiment. U pravilu pridjevi i prilozi su nosioci sentimenta dok imenice i glagoli najčešće ne iskazuju sentiment. Čest postupak koji se koristi prilikom analize sentimenta je **označavanje** vrsta riječi. Primjer pridruživanja kategorija riječima su Pennove banke stabala [15].

2. **Frekvencija izraza** nastavlja se na ideju pronalaska subjektivnih riječi. Osim što je važna prisutnost subjektivnih izraza, važna je i njihova učestalost. Dokument koji obiluje pozitivnim subjektivnim izrazima poput *dobar*, *sretan*, itd. vjerojatno će biti klasificiran kao pozitivan. Uz frekvenciju izraza, često se koristi i relevantnost izraza koja daje mjeru o važnosti pojedinog izraza za neki dokument.

3. **Negacija** je izdvojena kao posebna značajka budući da njena prisutnost drastično mijenja sentiment. Rečenica *"Ovo prijenosno računalo je dobro."* bit će klasificirana kao pozitivna. No, rečenica *"Ovo prijenosno računalno nije dobro."* klasificirat će se kao negativna upravo zbog negacije *nije* koja utječe na pridjev *dobar* dajući mu suprotno značenje.

Za kraj ovog poglavlja, valja u nekoliko rečenica dati i kratki osvrt na klasifikacijske metode prilikom analize sentimenta na razini dokumenta. U pravilu se za klasifikaciju sentimenta koriste metode nadziranog strojnog učenja. Predstavnici navedenih metoda su naivni Bayesov klasifikator i stroj s potpornim vektorima. U pravilu stroj s potpornim vektorima je nešto bolji u odnosu na Bayesov klasifikator, no obije metode postižu zadovoljavajuće rezultate [19]. Osim klasičnog strojnog učenja, javljaju se i neke druge metode poput vrednovanja koje pridružuje ocjenu skupu značajki te pridruženu ocjenu koristi za klasifikaciju [6]. Popularizacijom metoda dubokog učenja, istraživanja su krenula i u tom pravcu [21]. Valja napomenuti kako se osim binarne klasifikacije koja određuje je li tekst pozitivan ili negativan koristi i klasifikacija koja dokumentima pridaje ocjenu, najčešće od 1 do 5. Pritom ocjene 1 i 2 su negativne, 3 predstavlja neutralan sentiment, a 4 i 5 predstavlja pozitivan sentiment. Pokazalo se kako je regresijski stroj s potpornim vektorima bolji izbor kod ovakve formulacije problema u odnosu na sustave SVM-a koristeći višerazrednu klasifikaciju [18].

4.5. Problemi kod analize sentimenta

Prilikom analize sentimenta često se javljaju problemi vezani uz prirodnu obradu jezika te uz same korisnike. Za neke od njih, na žalost, još ne postoji efikasna metoda rješavanja istih. Neki od najčešćih problema su:

1. **Dvoznačnost** je jedan od velikih problema analize sentimenta često utječući na kvalitetu klasifikacije. Primjerice rečenica *"Ova kamera ubija."* predstavlja pozitivan sentiment dok rečenica *"Droga ubija!"* implicitno izražava korisnikov negativan stav prema psihoaktivnim supstancama.

2. Na problem dvoznačnosti naslanja se i problem **nositelja sentimenta**. Korisnik u pravilu koristi izjavne rečenice kako bi izrazio svoj sentiment. U rečenici *"Ovaj film je dosadan."* riječ *dosadan* je nositelj sentimenta. S druge strane, u rečenici *"Ovaj film je dosadan?"* riječ više ne nosi nikakav sentiment budući da se više ne radi o izjavnoj, već o upitnoj rečenici.

3. Zanimljiv problem na koji se nailazi prilikom analize sentimenta je **sarkazam**. Sarkazam se najčešće ne pojavljuje u recenzijama proizvoda i usluga, no čest je u raspravama. Primjer sarkazma je rečenica *"Baš si dobar košarkaš."* gdje izvor mišljenja daje negativan stav o igračkoj kvaliteti sugovornika. Valja primijetiti kako je teško detektirati sarkazam u slučaju teksta budući da je ton govornika ključni element sarkazma.

4. Ranije je kao jedan oblik mišljenja bilo navedeno implicitno mišljenje. Često rečenice ovakvoga tipa ne sadrže nositelje sentimenta. Rečenica *"Kupio sam spor automobil."* ne sadrži nosioce sentimenta, no s druge strane implicira negativan sentiment korisnika prema kupljenom automobilu.

5. Za kraj valja spomenuti i neželjena mišljenja korisnika (engl. *spam*). U pravilu su korisničke recenzije i ocjene proizvoda vrijedan izvor informacija. No, često se znaju javljati, pogotovo u raspravama velikog intenziteta sentimenta, pristrana i lažna mišljenja. Primjer takvih rasprava su političke rasprave. Cilj korisnika koji daju pristana mišljenja je promovirati svoj proizvod ili stav, ili diskreditirati tuđi. Problem otkrivanja neželjenog mišljenja također je jedan od interesantnih problema strojnog učenja.

5. Implementacija i rezultati

U ovom poglavlju iznesen je postupak primjene stroja s potpornim vektorima u analizi sentimenta korisničkih recenzija. Potpoglavlje 5.1 pojašnjava metodu dualnog koordinatnog spusta koja se koristi za optimizaciju stroja s potpornim vektorima. U potpoglavlju 5.2 dan je postupak vektorizacije teksta koji korisničke recenzije pretvara u vektor značajki, oblik pogodan za provođenje postupka klasifikacije koristeći SVM. Potpoglavlje 5.3 prikazuje rezultate klasifikacije i analizira ih.

5.1. Algoritam dualnog koordinatnog spusta

Algoritam **dualnog koordinatnog spusta** je efikasan algoritam za rješavanje dualnog optimizacijskog problema linearnog SVM-a [9]. Ovaj algoritam je izabran za implementaciju budući da za probleme analize sentimenta u pravilu nije potrebno korištenje jezgrenih trikova budući da je vektorski prostor primjeraka jako velikih dimenzija. U pravilu jednu značajku čini frekvencija ili relevantnost riječi ili izraza pa je za očekivati da će za dobru klasifikaciju biti potreban vektor značajki velikih dimenzija. Odabir linearnog klasifikatora omogućuje korištenje algoritma specijaliziranog za isključivo linearan slučaj. Prednost je vrijeme izvođenja u odnosu na primjerice algoritam sekvencijalne minimalne optimizacije (engl. *Sequential minimal optimization*, skraćeno SMO). Nedostatak je nemogućnost odabira jezgrenog trika, što u konačnici onemogućuje razdvajanje linearno nerazdvojivih podataka.

Ideja algoritma dualnog koordinatnog spusta je veoma slična ideji gradijentnog spusta. Za razliku od gradijentnog spusta koji prilikom jedne iteracije osvježava vrijednosti svih težina, algoritam dualnog koordinatnog spusta u jednoj iteraciji osvježava vrijednost samo jedne težine.

Uvjet optimizacijskog problema $0 \leq \alpha_i \leq C$ stavlja ograničenja na vrijednost Lagrangeovih multiplikatora. Ovaj uvjet također je važan prilikom optimizacije te će prilikom osvježavanja vrijednosti multiplikatora vraćati u zadane granice.

Valja malo formalnije opisati postupak optimizacije. Neka je zadan optimizacijski pro-

blem sličan 3.23:

$$\begin{aligned} \min_{\alpha} \quad & f(\boldsymbol{\alpha}) = \frac{1}{2}\boldsymbol{\alpha}^T\bar{\mathbf{Q}}\boldsymbol{\alpha} - \mathbf{e}^T\boldsymbol{\alpha} \\ \text{s obzirom na} \quad & 0 \leq \alpha_i \leq U,\ i = 1, \ldots, N. \end{aligned} \tag{5.1}$$

Razlika u odnosu na problem zadan izrazom 3.23 je minimalna. Naime, funkcija optimizacije je negirana pa se ne traži maksimum već minimum te je vektorizirana. Prvi uvjet je zapisan koristeći novu varijablu U budući da za L1-SVM vrijedi uvjet $0 \leq \alpha_i \leq C$ dok za L2-SVM vrijedi $\alpha_i \geq 0$. Iz ovog slijedi da za L1-SVM vrijedi $U = C$ dok za L2-SVM vrijedi $U = \infty$. Za matricu $\bar{\mathbf{Q}}$ vrijedi izraz $\bar{\mathbf{Q}} = \mathbf{Q} + \mathbf{D}$. Elementi matrice $\mathbf{Q}$ su $Q_{ij} = \alpha_i\alpha_j y^{(i)}y^{(j)}x^{(i)T}x^{(j)}$, dok je $\mathbf{D}$ dijagonalna matrica. Za L1-SVM $\mathbf{D}_{ii} = 0$, a za L2-SVM $D_{ii} = \frac{1}{2C}$. Valja napomenuti kako je slobodan član b maknut, a umjesto njega je dodana još jedna dimenzija vektoru težina $\mathbf{w}$. Tada vektor težina ima oblik $[\mathbf{w}, 1]$, a vektor značajki $\mathbf{x}$ postaje $[\mathbf{x}, 1]$. U ovom zapisu uglate zagrade predstavljaju konkatenaciju.

Algoritam slijedno osvježava sve vrijednosti multiplikatora. Postupak kojim se osvježava jedna težina zove se unutarnja iteracija, a postupak kojim se osvježavaju sve težine naziva se vanjska iteracija. Valja primijetiti kako jednu vanjsku iteraciju čini N unutarnjih. Vrijednost Lagrangeovih multiplikatora za vanjsku iteraciju k i unutarnju iteraciju i označavat će se s $\boldsymbol{\alpha}_i^k$. Za jednu unutarnju iteraciju rješava se potproblem:

$$\begin{aligned} \min_{d} \quad & f(\boldsymbol{\alpha}_i^k + d\mathbf{e}_i) = \frac{1}{2}\bar{Q}_{ii}d^2 + \nabla_i f(\boldsymbol{\alpha}_i^k)d + C \\ \text{s obzirom na} \quad & 0 \leq \alpha_i + d \leq U,\ i = 1, \ldots, N \end{aligned} \tag{5.2}$$

gdje je $\mathbf{e}_i$ vektor čiji je i-ti element jednak jedan dok su ostali članovi jednaki nuli. ∇_i je gradijent funkcije u odnosu na i-ti element, a C je konstanta. Odabire se $d = 0$ budući da za tu vrijednost nema potrebe osvježiti α_i. Deriviranjem funkcije po d te izjednačavanjem derivacije s nulom dobije se izraz:

$$\nabla_i f(\boldsymbol{\alpha}_i^k) = y^{(i)}\mathbf{w}^T\mathbf{x}^{(i)} - 1 + D_{ii}\alpha_i = 0. \tag{5.3}$$

Ako vrijedi izraz 5.3 tada vrijednosti $\boldsymbol{\alpha}_i^k$ ostaje nepromijenjena i u sljedećoj iteraciji. No, uz ovaj izraz, potrebno je uzeti u obzir i ograničenje $0 \leq \alpha_i \leq U$. Ovo ograničenje vraća vrijednost $\nabla_i f(\boldsymbol{\alpha}_i^k)$ u dozvoljeni raspon na sljedeći način:

$$PG_i^k = \begin{cases} \nabla_i f(\boldsymbol{\alpha}_i^k) & \text{za } 0 \leq \alpha_i \leq U \\ \min(0, \nabla_i f(\boldsymbol{\alpha}_i^k)) & \text{za } \alpha_i = 0 \\ \max(0, \nabla_i f(\boldsymbol{\alpha}_i^k)) & \text{za } \alpha_i = U. \end{cases} \tag{5.4}$$

Konačno, može se zaključiti kako se osvježavanje ne događa u slučaju $PG = 0$. U slučaju $PG \neq 0$, Lagrangeov multiplikator mijenja vrijednost na sljedeći način:

$$\alpha_i = \min(\max(0, \alpha_i - \frac{G}{\bar{Q}_{ii}}), U) \tag{5.5}$$

Za kraj iteracije, budući da se radi o optimizaciji linearnog stroja s potpornim vektorima, moguće je osvježiti vrijednost težina prema izrazu:

$$\mathbf{w} = \mathbf{w} + \delta y^{(i)} \mathbf{x}^{(i)} \tag{5.6}$$

gdje je δ razlika vrijednosti i-tog Lagrangeovog multiplikatora prije i poslije iteracije.

Trenutno je redoslijed osvježavanja Lagrangeovih multiplikatora poprilično deterministički. Naime, jedna vanjska iteracija algoritma uvijek istim redoslijedom osvježava vrijednosti algoritma. Pokazuje se kako nasumičan izbor redoslijeda osvježavanja multiplikatora dovodi do bolje konvergencije algoritma.

Često tijekom iteracija vrijednosti multiplikatora ostaju nepromijenjene, posebice u slučaju $\alpha_i = 0$ ili $\alpha_i = U$. Konstantno računanje gradijenta i pokušaj osvježavanja troši dragocjeno vrijeme. Kao odgovor na ovaj problem, uvodi se **aktivni skup** u kojem se nalaze multiplikatori koji za koje vrijedi $0 < \alpha_i < U$. Lagrangeovi multiplikatori se izbacuju iz aktivnog skupa ako vrijedi:

$$\begin{aligned} \alpha_i^k = 0 \text{ i } \nabla_i f(\boldsymbol{\alpha}_i^k)) > \bar{M}^{k-1} \text{ ili} \\ \alpha_i^k = U \text{ i } \nabla_i f(\boldsymbol{\alpha}_i^k)) < \bar{m}^{k-1} \end{aligned} \tag{5.7}$$

gdje su $\bar{M}^{k-1}$ i $\bar{m}^{k-1}$:

$$\bar{M}^{k-1} = \begin{cases} \max\limits_j PG_j^{k-1} & \text{za } \max\limits_j PG_j^{k-1} > 0 \\ \infty & \text{inače} \end{cases}$$

$$\bar{m}^{k-1} = \begin{cases} \min\limits_j PG_j^{k-1} & \text{za } \min\limits_j PG_j^{k-1} < 0 \\ -\infty & \text{inače.} \end{cases}$$

Vrijednosti $M^{k-1} = \max\limits_j PG_j^{k-1}$ i $m^{k-1} = \max\limits_j PG_j^{k-1}$ poslužit će i za određivanje zaustavnog uvjeta. U slučaju da vrijedi $M - m < \epsilon$ i aktivni skup sadrži sve multiplikatore, tada su svi multiplikatori postigli optimalne vrijednosti te algoritam zaustavlja s radom. U slučaju da aktivni skup ne sadrži sve multiplikatore, vrijednosti se vraćaju na početne, $M = \infty, m = -\infty$, te se u aktivni skup vraćaju svi multiplikatori.

5.2. Postupak vektorizacije teksta

5.2.1. Leksička analiza teksta

Postupak vektorizacije teksta započinje leksičkom analizom. **Leksička analiza** ili tokenizacija je postupak razdvajanja prepoznatljivih riječi, fraza i simbola. Jedna leksička jedinica zove se leksem. Leksičkom analizom tekst se pretvara u slijed leksema.

Najjednostavnija metoda leksičke analize je razdvajanje leksema po razmacima. Primjerice, iz rečenice *"Danas je lijep i sunčan dan."* mogu se izdvojiti sedam leksema: [*Danas*, *je*, *lijep*, *i*, *sunčan*, *dan*, .]. Za osnovni oblik rečenice ova metoda je sasvim dostatna. No, budući da je fokus ovog rada na analizi sentimenta, valja primijeniti ovaj postupak na problem analize sentimenta. Neka je zadana rečenica *"Ne volim ovaj film."*. Leksički analizator će razdvojiti ovu rečenicu na sljedeće lekseme: [*Ne*, *volim*, *ovaj*, *film*, .]. Nositelj sentimenta u ovoj rečenici je pridjev *volim* tj. negacija tog pridjeva. Problem je lako uočljiv - nositelj sentimenta je, strogo gledajući, pozitivan leksem, a rečenica u cjelini je negativna. Ovo je samo jedan od problema osnovnog modela. Primjerice, danas se često analiziraju podaci prikupljeni s interneta i društvenih mreža koji zahtijevaju modifikaciju osnovne leksičke analize.

Christopher Potts izradio je predložak leksičke analize primjerene za probleme analize sentimenta [3]. U tom predlošku opisan je način obrade teksta s društvenih mreža te obradu podataka prikupljenih metodama rudarenja.

Prvi korak u leksičkoj analizi je obrada HTML i XML oznaka. U pravilu, HTML i XML oznake se brišu iz dokumenta. Iznimka su oznake koje mogu nositi informaciju poput <*strong*>, <*b*> i sl. Primjerice, HTML oznaka u izrazu <*strong*>*Volim ovaj sladoled.*</*strong*> pojačava pozitivan sentiment korisnika prema sladoledu. Ove iznimke obradit će se tako da svi leksemi koji se nalaze unutar oznaka budu napisani velikim slovima. Nakon leksičke analize prethodno napisana rečenica dat će sljedeće lekseme: [*VOLIM*, *OVAJ*, *SLADOLED*, .]. Također, imenovani identiteti, primjerice *<*, zamjenjuju se svojim standardnim oblicima (u ovom primjeru identitet *<* zamjenjuje se znakom <).

Nakon obrade HTML i XML oznaka, slijedi izrada leksema. Osim standardnog razdvajanja riječi po razmacima, ovaj leksički analizator brine se i drugim različitim oblicima poput korisničkog imena na društvenoj mreži *Twitter*, telefonskog broja, emotikona, itd. Ovaj napredniji sustav prepoznavanja leksema omogućuje izdvajanje više značajka teksta koje mogu biti od velike važnosti za analizu sentimenta. Važno je napomenuti kako se svi leksemi zapisuju malim slovima, osim onih unutar posebnih oznaka. Interpunkcijski znakovi će kroz postupak tokenizacije biti zadržani.

Sljedeći korak u leksičkoj analizi je obrada negacije. Potrebno je izgraditi rječnik negacija - skup riječi koje nositeljima sentimenta pridaju suprotno značenje. Nailaskom na negaciju, ona se kao leksem briše i aktivira poseban način rada analizatora. Svakoj riječi nakon negacije dodaje se prefiks *"NEG_"*. Iz tog posebnog načina rada izlazi se nakon n broja leksema ili ako se naiđe na interpunkcijski znak. Primjerice, rečenica *"Ovo nije pozitivna rečenica."* nakon leksičke analize izgleda ovako: [*Ovo*, *NEG_pozitivna*, *NEG_rečenica*, .]. Ovime se postiže da izrazi koji sadrže nositelje sentimenta u kombinaciji s negacijom, ne budu klasificirani na isti način kao oni izrazi koji ne sadrže negaciju.

5.2.2. Lematizacija

Lematizacija je postupak svođenja leksema na kanonski oblik tj. lemu. Za glagole lema je infinitiv, za pridjeve to je muški rod jednine. Primjerice, riječ *radim* nakon postupka lematizacije postaje *raditi*, a riječ *najlošiji* postaje riječ *loš*. Sustav koji provodi postupak lematizacije zove se lematizator. Uz lematizator veže se i stemer. Oba sustava pokušavaju pronaći korijen dane riječi, no za razliku od lematizatora, stemer ne uzima u obzir vrstu riječi. Budući da su lematizatori i stemeri često preagresivni u svojim zadaćama, njihova korisnost u analizi sentimenta je upitna.

5.2.3. Izbacivanje zaustavnih riječi i interpunkcijskih znakova

Zaustavne riječi poput veznika *i*, *ili*, priloga *danas*, *gdje* i sl. u pravilu ne nose nikakvu informaciju stoga ih se u ovom postupku izbacuje iz liste leksema. Isto se radi i s interpunkcijskim znakovima. Primjerice, rečenica *"Danas je lijep i sunčan dan."* nakon leksičke analize i izbacivanja zaustavnih riječi i interpunkcijskih znakova tvori listu leksema [*lijep*, *sunčan*, *dan*].

5.2.4. N-grami

Osim leksema kao osnovne jedinice, moguće je izgraditi i složenije izraze. N-grami su izrazi koji se sastoje od n slijednih leksema. Koristeći n-grame moguće je dohvatiti informacije koje nisu dostupne jednostavnim izrazima. Primjerice, rečenica *"Njegov savjet uzeo sam sa zrnom soli."* sadrži frazu *"sa zrnom soli"*. Koristeći jednostavno razdvajanje na lekseme tj. 1-grame, dobije se niz: [*savjet*, *uzeti*, *zrno*, *sol*]. Vidljivo je kako 1-grami nisu u mogućnosti izvući informaciju iz teksta. No, u slučaju 2-grama dobiva se sljedeći niz: [*savjet uzeti*, *uzeti zrno*, *zrno soli*]. Koristeći 2-grame, izgrađen je niz izraza koji uspješno izvlači informaciju iz teksta.

5.2.5. Izvlačenje značajki teksta

Stroj s potpornim vektorima ne razumije tekst, niti listu leksema. On kao ulaz očekuje primjerak kojeg čini vektor brojeva. U ovom koraku vrši se preslikavanje leksema u broj pogodan za računalnu obradu.

U ovom radu bit će objašnjena tri različite metode preslikavanja. Metode su temeljne na modelu zbirki značajki (engl. *Bag of words*), a postupno se grade od jednostavnijih ka složenijim.

Najjednostavniji model izgradnje vektora značajki je binarni model. U binarnom modelu gradi se vektor značajki u kojem jedan element poprima vrijednost 0 u slučaju da se riječ

nije pojavila u tekstu, a 1 u slučaju da je riječ zastupljena. Neka su zadane dvije rečenice *"Ovo je pozitivna rečenica."* i *"Ovo je negativna rečenica."*. Korištenjem jednostavnog leksičkog analizatora prva rečenica daje niz [*ovo*, *je*, *pozitivna*, *rečenica*], a druga rečenica daje niz [*ovo*, *je*, *negativna*, *rečenica*]. Sljedeći korak u izgradnji vektora značajki je izgradnja vokabulara. Vokabular je skup riječi koje se pojavljuju u svim primjercima. Za gornji primjer vokabular je skup $\{$*Ovo*, *je*, *pozitivna*, *rečenica*, *negativna*$\}$. Elementi vokabulara tvore vektor značajki, a u slučaju velikog vokabulara potrebno je izvršiti odabir izraza. Odabir izraza je jedan od ključnih problema u izgradnji vektora značajki. U pravilu izbor izraza vrši se prema frekvenciji pojavljivanja tj. češće viđeni izrazi dobivaju prednost u odnosu na rjeđe viđene izraze. Takvim odabirom postiže se generalizacija tj. rijetko viđene riječi nemaju utjecaj na klasifikaciju sentimenta. S druge strane, riječi koje se često pojavljuju u tekstovima ne moraju nužno nositi sentiment. Takvi elementi nisu značajni za klasifikaciju. Budući da u gornjem primjeru vokabular nije velik, moguće je izgraditi vektor značajki sa svim elementima vokabulara. Neka su poredani gore prikazanim redoslijedom. Tada se za prvu rečenicu gradi primjerak $((1, 1, 1, 1, 0), 1)$, a za drugu rečenicu $((1, 1, 0, 1, 1), 0)$. Prvi element primjerka je vrijednost vektora značajki za zadanu rečenicu, a drugi element predstavlja oznaku razreda. Nakon izrade primjerka, moguće ih je dovesti na ulaz klasifikatora te provesti klasifikaciju teksta.

Iako je s osnovnim modelom zbirki značajki moguće provesti klasifikaciju, često ovakav model nije pretjerano uspješan. Jedan od problema gore navedenog modela je frekvencija pojavljivanja riječi. Naime, gore navedeni model ne vodi računa o frekvenciji pojavljivanja, već samo o tome je li se izraz pojavio u tekstu ili ne. Primjerice, neka se u jednom tekstu riječ *dobar* pojavila sto puta, a u nekom drugom tekstu svega jednom. Može se pretpostaviti kako je u prvom tekstu izraženiji pozitivan sentiment upravo zbog čestog korištenja nositelja pozitivnog sentimenta. No, za gore navedeni model nema razlike - vrijednost elementa vektora koji predstavlja navedeni pridjev bit će jednaka i za prvi i za drugi tekst. Stoga, modificira se osnovni model zbirki značajki te se umjesto pojavnosti izraza u tekstu iskazuje frekvencija pojavljivanja izraza. Neka su zadane rečenice: *"Ovo je jako pozitivna rečenica."* i *"Ovo je jako, jako pozitivna rečenica."*. U osnovnom modelu njihov vektor značajki je jednak i iznosi $(1, 1, 1, 1, 1)$. U poboljšanom modelu vektor značajki druge rečenice postaje $(1, 1, 2, 1, 1)$ čime se postiže razlika u odnosu na prvu rečenicu.

Frekvencijski model zbirki značajki donio je poboljšanja u odnosu na binarni model. No, i dalje ostaje problem izraza koji se često pojavljuju u svim dokumentima te nisu pretjerano korisni za analizu sentimenta. Primjer takvih riječi su zaustavne riječi koje se u pravilu brišu iz vokabulara. No, često postoje riječi specifične za određeni problem. Primjerice, u analizi sentimenta korisničkih recenzija filmova očekuje se često ponavljanje riječi *film*. Pretpostavlja se kako će za svaki primjerak riječ *film* biti relevantna te da neće nositi informaciju o

sentimentu. Stoga treba poboljšati model koji će osim relevantnosti riječi za pojedini dokument uzimati u obzir i odnos relevantnosti između dokumenta. Riječi koje se često pojavljuju u skupu dokumenata, smatrat će se manje relevantnima za pojedini dokument. Takav model naziva se TF-IDF model (engl. *term frequency–inverse document frequency*). TF-IDF predstavlja složenicu - TF opisuje relevantnost riječi za dokument, dok IDF - opisuje pojavnost riječi u svim dokumentima. Budući da je model TF-IDF matematički složeniji od prethodna dva modela, valja dati i nekoliko osnovnih izraza koji opisuju model. Postoji nekoliko različitih matematičkih izraza koji definiraju model, a jedna od definicija relevantnosti izraza dana je formulom:

$$\text{tf}(t,d) = 1 + \log(1 + \frac{f_t}{\sum_{t' \in d} f_{t'}}). \tag{5.8}$$

Iz formule je vidljivo kako će izrazi koji se češće pojavljuju u dokumentu imati veću tf-vrijednost od onih koji se pojavljuju rjeđe. IDF komponenta modela može se zadati izrazom:

$$\text{idf}(t,D) = \log(1 + \frac{N}{n_t}) \tag{5.9}$$

gdje je N ukupan broj dokumenata, a n_t broj dokumenata u kojima se pojavljuje izraz t. Može se uočiti kako će veću vrijednost idf-a za fiksni N, imati oni izrazi koji se pojavljuju u što manje dokumenata. Ukupna vrijednost tdidf iznosi:

$$\text{tfidf}(t,d) = \text{tf}(t,d)\text{idf}(t,D) = (1 + \log(1 + \frac{f_t}{\sum_{t' \in d} f_{t'}}))\log(1 + \frac{N}{n_t}). \tag{5.10}$$

Upravo je tfidf vrijednost elementa nekog izraza u vektoru značajki za neki dokument. Vrijednost idf je korisna i za odabir značajki. Postavljanjem granica na vrijednost idf-a, mogu se izbaciti veoma rijetke te jako česte riječi.

5.3. Analiza sentimenta korisničkih recenzija filmova

U sklopu rada bilo je potrebno primijeniti dosad navedene metode i modele na konkretnom primjeru. Izbor je pao na analizu sentimenta korisničkih recenzija filmova. Skup podataka korišten u ovom radu naziva se *Large Movie Review Dataset*, a sačinjavaju ga recenzije preuzete s internetske baze podataka IMDb i obrađene od sveučilišta Stanford [14]. Korišteni podaci sastoje se od pedeset tisuća korisničkih recenzija filmova na engleskom jeziku od čega je polovica pozitivnih recenzija, a ostale recenzije su negativne. Također, prilikom obrade teksta, točnije kod postupka lematizacije i postupka izbacivanja stranih riječi korišten je programski paket NLTK [13]. Za pretvorbu podataka u vektor značajki te za potrebe testiranja implementacije stroja s potpornim vektorima korišten je programski paket scikit-learn [20].

Metoda	Negacija isljućena	Negacija uključena
Bez lematizacije i steminga	89.18%	89.44%
Lematizacija	89.01%	89.29%
Steming	88.9%	89.33%

Tablica 5.1: Točnost klasifikacije recenzija u odnosu na komponente leksičkog analizatora

5.3.1. Testiranje komponenata leksičkog analizatora

Leksički analizator osim što je zadužen za pronalazak leksema, vrši i modifikacije na njima. Točnije, zadužen je za postupak pronalaska kanonskog oblika riječi te negaciju leksema ranije opisanom metodom. Očekuje se kako će upravo ti postupci doprinijeti uspješnijoj klasifikaciji podataka. Rezultati su dani u tablici 5.1.

Iz tablice 5.1 je vidljivo kako negacija riječi ne utječe previše na točnost klasifikatora. Iznenađujuća je činjenica da lematizacija i steming negativno utječu na točnost klasifikatora, iako je razlika veoma malena. Očito su postupci pretvorbe riječi u kanonski oblik preagresivni te se pretvorbom gubi određen dio informacije.

5.3.2. Testiranje dimenzija vektora značajki i metoda vektorizacije

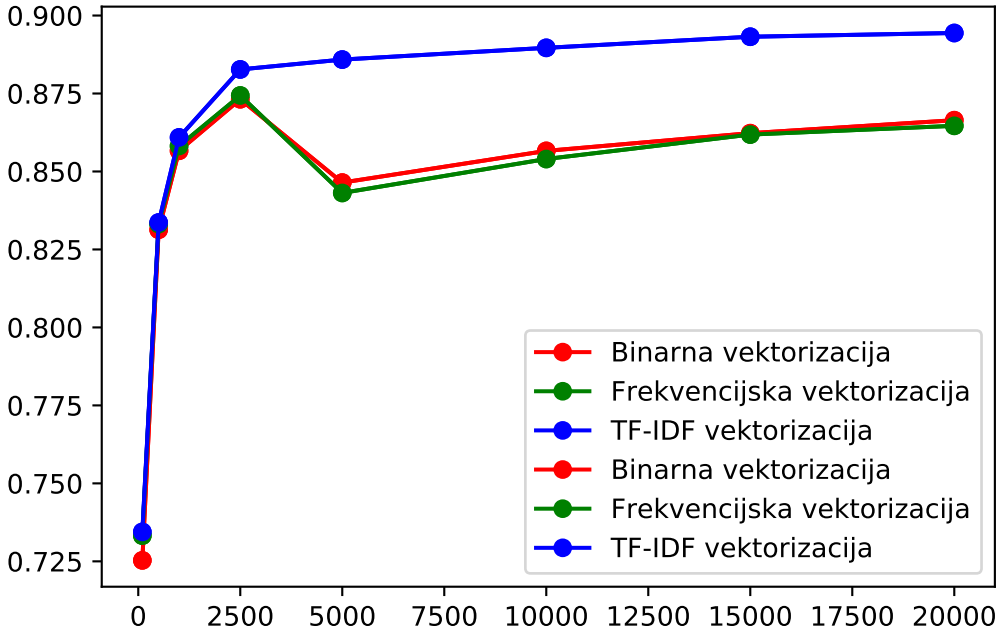
Dimenzija vektora značajki jedan je od najvažnijih parametara prilikom analize sentimenta. Cilj je pronaći vrijednost parametra s kojim klasifikator dobro generalizira podatke, a istovremeno pružajući mu maksimalnu količinu informacije. Također, u ovom dijelu cilj je testirati tri opisane metode vektorizacije. Očekuje se kako će model TF-IDF davati najbolje rezultate budući da je složeniji od ostalih modela.

Sa slike 5.1 može se vidjeti kako model TF-IDF daje najbolje rezultate. Posebice je vidljiva razlika između modela TF-IDF i ostala dva modela za vektore značajki većih dimenzija. Iznenađujuć je odnos između binarnog i frekvencijskog modela. Oba modela postižu slične rezultate, no za neke dimenzije binarni model se ponaša bolje od frekvencijskog.

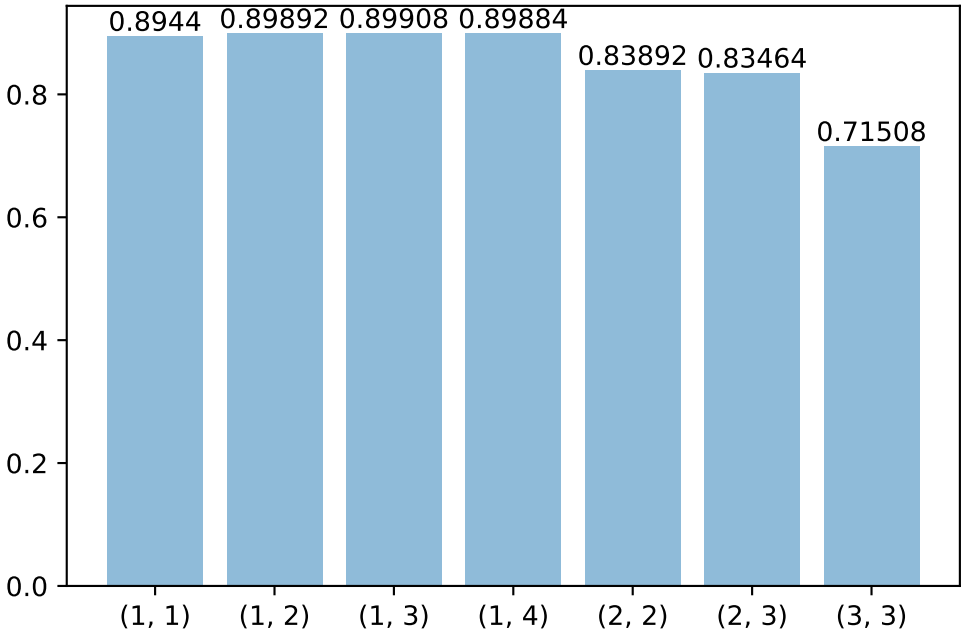
5.3.3. Izbor n-grama

Nakon izbora modela i dimenzije vektora značajki, slijedi odabir raspona n-grama. Dosad u testovima su korišteni unigrami, no cilj ovog testa je provjeriti kako odabir n-grama utječe na točnost klasifikatora. Osim klasifikatora koji koristi samo jednu vrijednost parametra n, može se izgraditi i klasifikator koji primjerice koristi unigrame i bigrame. Rezultati su dani na grafu 5.2, a raspon je zadan uređenim parom (primjerice unigrami, bigrami i trigrami su zadani parom (1, 3)).

Rezultati pokazuju kako unigrami samostalno ili u kombinaciji s drugim n-gramima daju



Slika 5.1: Točnost klasifikacije u odnosu na model vektorizacije i dimenziju vektora značajki



Slika 5.2: Točnost klasifikatora u odnosu na izbor n-grama

Rank	Leksem	Težina	**Rank**	Leksem	Težina
1	worst	-5.097	**11**	fails	-2.966
2	7/10	3.866	**12**	great	2.941
3	awful	-3.821	**13**	disappointment	-2.933
4	bad	-3.495	**14**	8/	2.857
5	excellent	3.456	**15**	disappointing	-2.805
6	dull	-3.418	**16**	poor	-2.788
7	waste	-3.413	**17**	waste_NEG	-2.779
8	4/10	-3.35	**18**	8	2.714
9	boring	-3.208	**19**	unfortunately	-2.686
10	terrible	-3.04	**20**	amazing	2.67

Tablica 5.2: Nositelji sentimenta koji najviše utječu na klasifikaciju

najbolje rezultate. Unigrami, bigrami i trigrami zajedno ostvaruju najbolji rezultat, iako su razlike neznatne. Korištenjem složenijih struktura bez korištenja unigrama, klasifikator gubi na točnosti.

5.3.4. Analiza nositelja sentimenta

Ne utječu svi leksemi na klasifikaciju jednako. Očekuje se da će pridjevi i prilozi nositi sentiment dok glagoli i imenice u pravilu ne utječu na klasifikaciju. Stoga je pametno analizirati i utjecaj leksema na klasifikaciju. Svakom leksemu pridodana je težina dobivena optimizacijom stroja s potpornim vektorima.

Tablica 5.2 prikazuje dvadeset leksema koji najviše utječu na klasifikaciju. Očekivano, tablica obiluje pridjevima. Što se tiče intenziteta sentimenta, posebno iskače riječ *worst* (hrv. *najlošiji*) koja ima najveći negativni intenzitet. Valja uočiti kako prevladavaju nositelji negativnog sentimenta. Samo je šest pozitivnih nositelja sentimenta dok su ostali nositelji negativnog sentimenta. No, ako se uzme svih dvadeset tisuća leksema u obzir, prevladavaju nositelji pozitivnog sentimenta, a našlo se i nekoliko riječi koje ne utječu na sentiment. Zanimljivo je uočiti kako nekoliko leksema poput *4/10* i *7/10* izrazito utječu na klasifikaciju. Ti leksemi predstavljaju korisničke ocjene filmova te su dobar orijentir za klasifikaciju recenzija. Interesantna je i činjenica da su recenzije po prirodi negativne budući da vrijednost slobodnog člana b iznosi -0.048.

Isječak recenzije	Označeno	Klasificirano
...The cinematography is badly lit, with everything looking grainy and ugly. Thesound is so terrible that you can barely hear what people are saying. The worst thing in this movie is the reason you're watching it-the sex....	1	0
...Bad acting, bad direction, bad looking woman, bad sets, bad cinematography, bad sound and bad sex scenes. The filmmakers should learn the difference between raunchy and erotic....	1	0
...there's something compelling and memorable about it. Like another commenter on the film, I saw this in childhood. It's been thirty three years since 1952, but I have never forgotten the story or its ridiculously cumbersome title. See it if you have the opportunity.	0	1
...HOWEVER, understand that the self-indulgent director also had many "funny gags" that totally fell flat and hurt the movie. His "camera tricks" weren't so much tricky but annoying and stupid. IGNORE THESE AND KEEP WATCHING–it does get better. The film is fast paced, funny and worth seeing. In particular, I really liked watching the acting and mugging of Max Linder–he was so expressive and funny! Too bad he is virtually forgotten today.	1	0

Tablica 5.3: Isječci pogrešno klasificiranih recenzija

5.3.5. Analiza pogrešno klasificiranih primjeraka

Nakon postupka klasifikacije, dobra je praksa pogledati i analizirati primjerke koji nisu ispravno klasificirani. Stroj s potpornim vektorima pogriješio je u 2640 primjeraka. Otprilike je podjednako lažno pozitivnih i lažno negativnih primjeraka.

Nekoliko isječaka pogrešno klasificiranih primjeraka recenzija dano je u tablici 5.3. U pravilu, dojam je da klasifikator ispravno klasificira primjerke te da su tekstovi pogrešno označeni. Zadnji primjerak prikazuje jedan od problema obrade teksta - nerazumijevanje konteksta. Iako isječak ima nekoliko nositelja negativnog sentimenta, uz pomoć konteksta ova recenzija je nedvojbeno pozitivna.

6. Zaključak

U ovom radu dan obrađen je model stroja s potpornim vektorima. Pokazana je ideja modela, problem optimizacije te proširenje na višerazrednu klasifikaciju te regresiju. Implementiran je dualni koordinatni spust, algoritam pogodan za učenje linearnog SVM-a. Također, dan je uvod u problem analize sentimenta te postupak vektorizacije teksta. Na konkretnom primjeru analize sentimenta korisničkih recenzija filmova pokazana je robusnost stroja s potpornim vektorima. Rezultati dobiveni klasifikacijom u pravilu su u skladu s očekivanjima.

Budući da je fokus u analizi sentimenta korisničkih recenzija filmova bio pronalazak pozitivnog i negativnog sentimenta, daljnji smjer istraživanja može biti u pravcu analize i predviđanja intenziteta sentimenta. Valjalo bi usporediti sustav stroja s potpornim vektorima koji provodi višerazrednu klasifikaciju te regresijskog modela za predviđanje ocjena. Valjalo bi dati i usporedbu stroja s potpornim vektorima i nekih drugih metoda strojnog učenja te metodama dubokog učenja. Isto tako, valjalo bi se posvetiti problemu konteksta u analizi sentimenta te analizi sentimenta na razini rečenica ili značajki. Pogodan problem koji se dotiče konteksta i razini značajki su političke rasprave.

Na kraju valja zaključiti kako model stroja s potpornim vektorima je jedan od najboljih modela za binarnu klasifikaciju podataka. Njegova robusnost na pretreniranost, jezgreni trikovi koji rješavaju problem linearno nerazdvojivih podataka te mogućnost proširenja na probleme višerazredne klasifikacije te probleme regresije čine ga dobrim izborom za većinu problema iz domene strojnog učenja.

LITERATURA

[1] M. A. Aizerman, E. A. Braverman, i L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. U *Automation and Remote Control,*, broj 25 u Automation and Remote Control,, stranice 821–837, 1964.

[2] Bernhard E. Boser, Isabelle M. Guyon, i Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. U *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, stranice 144–152, New York, NY, USA, 1992. ACM. ISBN 0-89791-497-X. doi: 10.1145/130385.130401. URL http://doi.acm.org/10.1145/130385.130401.

[3] Potts Christopher. Sentiment-aware tokenizer. http://sentiment.christopherpotts.net/tokenizing.html, 2011. Accessed: 2017-06-03.

[4] Corinna Cortes i Vladimir Vapnik. Support-vector networks. *Mach. Learn.*, 20(3):273–297, Rujan 1995. ISSN 0885-6125. doi: 10.1023/A:1022627411411. URL http://dx.doi.org/10.1023/A:1022627411411.

[5] Thomas M. Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *Electronic Computers, IEEE Transactions on*, EC-14(3):326–334, 1965. URL http://hebb.mit.edu/courses/9.641/2002/readings/Cover65.pdf.

[6] Kushal Dave, Steve Lawrence, i David M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. U *Proceedings of the 12th International Conference on World Wide Web*, WWW '03, stranice 519–528, New York, NY, USA, 2003. ACM. ISBN 1-58113-680-3. doi: 10.1145/775152.775226. URL http://doi.acm.org/10.1145/775152.775226.

[7] Harris Drucker, Chris J. C. Burges, Linda Kaufman, Alex Smola, i Vladimir Vapnik. Support vector regression machines. U *Advances in Neural Information Processing Systems 9*, stranice 155–161. MIT Press, 1997.

[8] Trevor Hastie, Robert Tibshirani, i Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[9] Cho-Jui Hsieh, Kai-Wei Chang, Chih-Jen Lin, S. Sathiya Keerthi, i S. Sundararajan. A dual coordinate descent method for large-scale linear svm. U *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, stranice 408–415, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4. doi: 10.1145/1390156.1390208. URL http://doi.acm.org/10.1145/1390156.1390208.

[10] Minqing Hu i Bing Liu. Mining and summarizing customer reviews. U *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, stranice 168–177, New York, NY, USA, 2004. ACM. ISBN 1-58113-888-1. doi: 10.1145/1014052.1014073. URL http://doi.acm.org/10.1145/1014052.1014073.

[11] Bing Liu. Sentiment analysis and subjectivity. *Handbook of Natural Language Processing*, 2nd ed, 2010.

[12] Bing Liu. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012. ISBN 1608458849, 9781608458844.

[13] Edward Loper i Steven Bird. Nltk: The natural language toolkit. U *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ETMTNLP '02, stranice 63–70, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1118108.1118117. URL http://dx.doi.org/10.3115/1118108.1118117.

[14] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, i Christopher Potts. Learning word vectors for sentiment analysis. U *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, stranice 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P11-1015.

[15] Mitchell P. Marcus, Beatrice Santorini, i Mary Ann Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *COMPUTATIONAL LINGUISTICS*, 19(2):313–330, 1993.

[16] Tetsuya Nasukawa i Jeonghee Yi. Sentiment analysis: Capturing favorability using natural language processing. U *Proceedings of the 2Nd International Conference on*

Knowledge Capture, K-CAP '03, stranice 70–77, New York, NY, USA, 2003. ACM. ISBN 1-58113-583-1. doi: 10.1145/945645.945658. URL http://doi.acm.org/10.1145/945645.945658.

[17] Andrew Ng. Cs229 lecture notes. http://cs229.stanford.edu/notes/cs229-notes3.pdf. Accessed: 2017-06-05.

[18] Bo Pang i Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. U *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, stranice 115–124, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics. doi: 10.3115/1219840.1219855. URL https://doi.org/10.3115/1219840.1219855.

[19] Bo Pang, Lillian Lee, i Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. U *Proceedings of EMNLP*, stranice 79–86, 2002.

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

[21] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, i Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. U *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, stranice 1631–1642, Stroudsburg, PA, October 2013. Association for Computational Linguistics.

[22] V. Vapnik i A. Lerner. Pattern recognition using generalized portrait method. *Avtomatika i Telemekhanika*, 24(6):774–780, 1963.

[23] Janyce M. Wiebe, Rebecca F. Bruce, i Thomas P. O'Hara. Development and use of a gold-standard data set for subjectivity classifications. U *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, stranice 246–253, College Park, Maryland, USA, June 1999. Association for Computational Linguistics. doi: 10.3115/1034678.1034721. URL http://www.aclweb.org/anthology/P99-1032.

Primjena stroja s potpornim vektorima za analizu sentimenta korisničkih recenzija

Sažetak

Stroj s potpornim vektorima (engl. Support Vector Machine, SVM) vrlo je općenit postupak koji omogućava rješavanje klasifikacijskih problema te problema funkcijske regresije. Isti je u praksi primjenjivan na niz praktičnih problema. U okviru završnog rada proučen je i opisan stroj s potpornim vektorima. Također, ostvarena je njegova programska implementacija kao i implementacija prikladnog postupka učenja istoga. Proučena je primjena stroja s potpornim vektorima na rješavanje problema analize sentimenta korisničkih recenzija te je ispitan rad algoritma na korisničkim recenzijama filmova. Rezultati dobiveni klasifikacijom recenzija su prikazani i analizirani.

Ključne riječi: stroj s potpornim vektorima, analiza sentimenta, klasifikacija, SVM, strojno učenje, regresija, dualni koordinatni spust

Application of Support Vector Machine for Users' Reviews Sentiment Analysis

Abstract

Support vector machines (SVM) are supervised learning models which can solve regression and classification problems. This thesis describes and analyses the support vector machine model. Moreover, dual coordinate descent, an optimization algorithm for solving linear SVM is implemented. In the second part of the thesis, a brief introduction to sentiment analysis is given. In the last part of the thesis, film reviews are classified using linear support vector machine and the results are interpreted.

Keywords: Support Vector Machine, SVM, Sentiment Analysis, Classification, Regression, Machine Learning, Dual Coordinate Descent