Support Vector Machine

Tim 6_23

- Anastasija Samčović, SW44/2019
- Strahinja Popović, SW51/2019
- Srđan Đurić, SW63/2019

Obrada podataka

Prvi korak u rešavanju datog problema bila je pregled podataka iz .tsv fajla u kome se nalaze recenzije. Među recenzijama je utvrđeno da postoji dosta reči koje ne nose težinu u odlučivanju da li je recenzija pozitivna ili negativna. Prvo smo umanjili sva slova. U tekstu je uočeno postojanje emoji karaktera. Emoji karaktere smo prvo zamenili sa praznim „space" karakterom. Međutim, utvrdili smo da bolje rezultate dobijamo kada tužan emoji zamenimo sa tekstom „lose", a srećne emoji-e sa tekstom „odlicno".

Nakon toga smo iz ulaznih podataka uklonili sve brojeve, specijalne karaktere i znakove interpunkcije. Eksperimentisanjem smo utvrdili da naš model daje bolje rezultate kada karaktere srpske latinice koji imaju kvačicu zamenimo sa „ošišanom" latinicom(npr. š -> s).

Reč nikad/nikada više smo grupisali u jednu „nikadvise" jer smo primetili da se ova reč jako često koristi u negativnoj konotaciji. Izbacili smo određeni broj reči koje imaju manje od 4 karaktera za koje smo utvrdili da ne nose bitnu ulogu u zaključivanju. Reči koje su u negativnoj konotaciji(npr. nije dobro, nisam zadovoljan) smo spojili kako ne bi davale suprotno značenje. Reč dobro se javlja u većini slučajeva u pozitivnim recenzijama, te ovim spajanjem dobijamo reč „nijedobro", na koju neće uticati broj javljanja reči dobro iz pozitivnih recenzija.

Problem upotrebe različitih lica smo rešili tako što smo određeni skup reči sveli na njihove korene ili nešto približno korenu. Primeri:

Bezobrazan/bezobrazna -> bezobraz

Najbolja/najbolji/najbolje -> najbolj

Testirani algoritmi

Skup podataka korišćen za testiranje predstavlja 30% datog trening skupa, a skup podataka korišćen za treniranje modela predstavlja 70% datog trening skupa.
Primenili smo vektorizaciju kako bi konvertovali sirove tekstualne podatke u numeričke. Prvo smo probali Bag of Words pristup, ali smo utvrdili da nam TF-IDF pristup daje bolje rezultate. TF-IDF pristup se koristi kada je potrebno dublje razumeti semantiku teksta, što je bilo potrebno u našem slučaju.

Isprobali smo linear, polynomial i RBF pristupe kod SVM algoritma. Polynomial pristup smo probali sa stepenima od 1 do 5, najbolje rezultate smo dobili sa prvim stepenom, što se ne razlikuje od rezultata linear pristupa. Među različitim pristupima nije postojalo veliko odstupanje u dobijenom krajnjem rezultatu(razlika je bila u decimalama). Utvrdili smo da SVM sa RBF kernelom sa hiper-parametrima **gamma=0.62** i **C=1.4** dobijamo najbolje rezultate.