Automatsko grupisanje reči po semantičkoj sličnosti pomoću k-Means algoritma

Atila Farkaš, *Fakultet tehničkih nauka u Novom Sadu*

***Sadržaj* — Za razumevanje prirodnog jezika od strane nekog automatskog sistema potrebno je predstavljanje značenja na taj način da ona bude upotrebljiva od strane automatskog sistema. Automatsko pronalaženje mere sličnosti neke nove reči u odnosu na neku poznatu reč je mnogo jednostavnije nego određivanje njenog tačnog značenja. U radu je opisana primena k-Means algoritma za automatsko grupisanje reči po semantičkoj sličnosti. Argumenti k-Means algoritma su vektori koji predstavljaju semantičku povezanosti reči.**

***Ključne reči* — Obrada prirodnog jezika, automatsko grupisanje po semantičkoj sličnosti, k-Means algoritam.**

I. UVOD

OBRADA prirodnog jezika je oblast veštačke inteligencije i lingvistike, koja se bavi proučavanjem automatskog generisanja i razumevanja prirodnog ljudskog jezika. Sistemi za generisanje prirodnog jezika pretvaraju informacije iz računarske baze podataka u ljudski jezik koji prirodno zvuči a sistemi za razumevanje prirodnog jezika pretvaraju primere ljudskog jezika u više formalne predstave sa kojima računarski programi lakše manipulišu. Obrada prirodnog jezika je veoma privlačan metod interakcije između čoveka i računara. Programi koji se bave obradom prirodnog jezika mogu da služe i samo kao interfejs nekom primarnom programu [1]. Npr. prirodno jezički interfejs za sistem baza podataka prevodi ulaz od strane korisnika u formalni upit u baze i zatim sistem nastavlja obradu bez dalje potrebe za tehnikama obrade prirodnog jezika.

Veštačka inteligencija kao pojam u širem smislu, označava kapacitet jedne veštačke tvorevine za realizovanje funkcija koje su karakteristika ljudskog razmišljanja, kao npr. razumevanje prirodnih (i veštačkih) jezika. U procesiranju informacija koncentriše se na programe koji nastoje da osposobe računar za razumevanje pisane i verbalne informacije, stvaranje rezimea, davanje odgovara na određena pitanja ili redistribuciju podataka korisnicima zainteresovanim za određene delove tih informacija. U tim programima, od suštinskog je značaja, kapacitet sistema za stvaranjem gramatički korektnih rečenica i uspostavljanje veze između reči i ideja, odnosno identifikacija značenja. Dok je problem strukturne logike jezika, odnosno njegove sintakse, moguće rešiti programiranjem odgovarajućih algoritama, problem značenja, ili semantike, je mnogo dublji i ide u pravcu autentične veštačke inteligencije.

Sistem mora da komunicira sa čovekom i drugim inteligentnim sistemima na "prijateljski način" - zato treba da upotrebljava prirodni jezik i govor. Takva komunikacija podrazumeva baratanje i dvosmislenostima i gramatički neispravnim rečenicama, tolerisanje grešaka i nejasnoća u komunikaciji [2]. Praktični sistemi za obradu prirodnih jezika moraju donositi ne dvosmislene odluke kod značenja reči, kategorija reči, kod sintaksne strukture i u semantičkom domenu.

Atila Farkaš, Veljka Vlahovića 18, 21220 Bečej, Srbija; (telefon: +381-63-863-77-36; e-mail: atilaf@yahoo.com).

II. KLASTEROVANJE (GRUPISANJE)

Klasterovanje predstavlja particionisanje skupa objekata u podskupove (grupe – klastere), tako da objekti u podskupovima dele neko zajedničko obeležje. Cilj algoritma za klasterovanje je da svrsta objekte sa sličnim osobinama u istu grupu.

Jedan od najjednostavnijih algoritama je k-Means algoritam koji vrši nehijerarhijsko hard klasterovanje objekta, što znači da jedan objekat može pripadati samo jednom klasteru.

Klasteri su definisani centrima koje određuju njihovi članovi. Prvo se proizvoljno izaberu inicijalni centri klastera. Zatim se prolazi kroz nekoliko iteracija tokom kojih se objekti pridružuju klasterima čijim centrima su najbliži. U svakoj iteraciji, kada su svi objekti pridruženi, preračunava se centar svakog klastera kao srednja vrednost njegovih članova:

$$\vec{\mu} = \frac{1}{\left|c_j\right|} \sum_{\vec{x} \in c_j} \vec{x} \qquad (1)$$

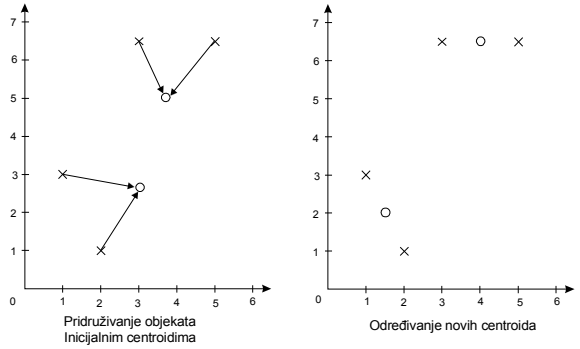
Članovi (objekti) su predstavljeni pomoću vektora $\vec{x} = (x_1, x_2, ..., x_n)$

a $\left|c_j\right|$ je broj elemenata j-tog klastera, c_j.

Dobijeni centri su poznati kao centroidi i oni se najčešće ne poklapaju sa objektima unutar te klase

Sl. 1. prikazuje primer jedne iteracije k-Means algoritma. Prvo se objekti pridruže klasterima čijim centrima su najbliži. Zatim ponovo određuju centri kao srednja vrednost njihovih članova. U ovom slučaju, dalja iteracija ne bi promenila klastere, s obzirom da

pridruživanje novim centrima ne menja pripadnost klasteru ni jednog objekta. To znači da se ni centri ne bi promenili u sledećem preračunavanju. Ali ovo nije opšti slučaj. Najčešće je potrebno nekoliko iteracija da bi algoritam konvergirao [3].



Sl. 1. Jedna iteracija k-Means algoritma

Kao rastojanje može se koristiti i kosinus ugla između vektora kojima su opisani objekti i vektora koji predstavlja centroid:

$$\cos(\vec{x},\vec{y}) = \frac{\vec{x}\cdot\vec{y}}{|\vec{x}|\cdot|\vec{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}} \qquad (2)$$

Određivanje klastera sa što boljim rasporedom objekata moguće je minimiziranjem nekih od sledećih metrika:

- maksimalno rastojanje objekata od svojih centroida
- zbir srednjih vrednosti rastojanja od centrioda za sve kalstere
- zbir varijansi po svim klasterima
- ukupno rastojanje objekata od svojih centroida

Često se ne zna unapred tačan broj klastera. Gore navedene metrike mogu posluže i kod određivanja broja klastera n, tako što će se algoritam izvršavati više puta za različito n dok mera za kvalitet ne bude zadovoljavajuća.

Kod k-Means algoritma inicijalni centri se biraju proizvoljno. U zavisnosti od strukture skupa objekata ovaj izbor može biti važan. Male promene kod inicijalnih centroida mogu dati skroz različite rezultate klasterovanja [4]. Izvršavanje algoritma se može ponoviti više puta za različit izbor početnih centorida i da se na kraju izabere najbolji rezultat, za koji je npr. minimalna neka od gore navedenih metrika. Na rezultat takođe može da utiče i redosled objekata koje grupišu.

A. Primena k-Means algoritma za grupisanje reči

k-Means algoritam za klasterovanje može da se primeni i na rečima prirodnog jezika. Reči, čije značenje je na neki način povezano, svrstavaju se u istu grupu. Kriterijum "sličnosti" može biti semantička sličnost. Pod semantičkoj sličnosti se ne misli samo na sinonime, već se smatra da su neke reči slične ako pripadaju istoj semantičkoj kategoriji (sadržini). Npr. lekar, ambulanta, operacija, mogu pripadati jednoj grupi semantički sličnih reči.

Smatraće se da su reči slične ukoliko postoji verovatnoća da se one pojave zajedno. Pored velikog broja načina određivanja mere semantičke sličnosti, najbolje može da se shvati pomoću sličnosti vektora. Reči, čija semantička sličnost želi da se odredi, predstavljaju se pomoću vektora u višedimenzionalnom prostoru.

Algoritam treba da bude potpuno automatski i da se oslanja na statističku analizu vrlo velikog broja tekstova. Nakon analize, formira se matrica na osnovu broja dokumenta u kojima se pojavljuju obe reči Vektor vrste ove matrice predstavljaju reči kao vektore. Primer matrice prikazan je u Tabeli 1. Element a_{ij} predstavlja broj dokumenta u kojima su se i *i*-ta i *j*-ta reč pojavili u isto vreme. Reči se smatraju sličnim ako se one pojavljuju zajedno u dokumentima. Međutim, reči mogu biti slične i kada se one ne pojavljuju zajedno, ali se javljaju u društvu neke treće reči, koja na taj način čini vezu između njih.

TABELA 1: MATRICA ZAJEDNIČKOG POJAVLJIVANJA REČI

i \ j	parlament	ministar	vlast	lopta	utakmica	turnir
parlament	30	18	17	0	0	0
ministar	18	47	15	0	1	0
vlast	17	15	33	0	0	0
lopta	0	0	0	21	16	4
utakmica	0	1	0	16	64	21
turnir	0	0	0	4	21	36

Zajednička pojavljivanja mogu da se definišu nad dokumentima, paragrafima ili nad nekim drugim jedinicama. Različit način prikaza reči u vektorskom prostoru, daje različite semantičke sličnosti.

U navedenom primeru uzete su samo 6 reči koje se mogu podeliti u dve grupe - sport ili politika. Algoritam, međutim ne treba da zna koje kategorije predstavljaju ove grupe. Algoritmu takođe nije poznat ni broj grupa, već se ona zadaje ili određuje tako da se dobije najbolji rezultat koji zadovoljava neki kriterijum ili minimizira neku od pomenutih metrika. Najbolji izbor za broj klastera (grupa) može da se odredi i ekperimentisanjem. Pri analizi rezultata klasterovanja za različiti broj klastera, dolazi se do zaključka da se optimalnim izborom broja klasetra smanjuje uticaj početnog izbora centroida kao i redosleda reči koji se grupišu, na rezultat klastrovanja k-Means algoritmom.

Tabela 1. dobijena je analizom 250 dokumenata, u proseku sa oko 600 reči po dokumentu. Vektori koji predstavljaju reči *parlament, ministar, vlast, lopta, utakmica, turnir* su redom:

$$\vec{x}_1 = (30,18,17,0,0,0) \qquad \vec{x}_4 = (0,0,0,21,16,4)$$
$$\vec{x}_2 = (18,47,15,0,1,0) \qquad \vec{x}_5 = (0,1,0,16,64,21)$$
$$\vec{x}_3 = (17,15,33,0,0,0) \qquad \vec{x}_6 = (0,0,0,4,21,36)$$

Izbor inicijalnih centorida je proizvoljan i oni mogu da se poklapaju npr. sa vektorima $\vec{x}_1$ i $\vec{x}_2$:

$$\vec{\mu}_1 = (30,18,17,0,0,0)$$
$$\vec{\mu}_2 = (18,47,15,0,1,0)$$

Nakon prve iteracije, vektori $\vec{x}_1$ i $\vec{x}_3$ će pripasti prvom klasteru c_1, pošto će ugao između ovih vektora i prvog centroida $\vec{\mu}_1$ biti manji od ugla koji zaklapaju su sa drugim centroidom (fromula (2)). Analogno će vektori $\vec{x}_2$, $\vec{x}_4$, $\vec{x}_5$, $\vec{x}_6$ pripasti drugom klasteru c_2.

Pomoću formule (1) odrede se novi centroidi:

$$\vec{\mu}_1 = (23,16,25,0,0,0)$$
$$\vec{\mu}_2 = (4,12,3,10,25,15)$$

Nakon druge iteracije raspored postaje:

$$c_1 = \{\vec{x}_1, \vec{x}_2, \vec{x}_3\} \text{ i } c_2 = \{\vec{x}_4, \vec{x}_5, \vec{x}_6\}$$

a novi centroidi su:

$$\vec{\mu}_1 = (21,26,21,0,0,0)$$
$$\vec{\mu}_2 = (0,0,0,13,33,20)$$

U svakoj narednoj iteraciji centroidi ostaju isti, pošto vektori (reči) ne menjaju dalje pripadnost klasterima i time se i završava iteracija. Rezultat klasterovanja u dve grupe je dakle:

1. grupa: parlament, ministar, vlast
2. grupa: lopta, utakmica, turnir.

III. ZAKLJUČAK

Određivanjem pripadnosti reči klasterima, dobijaju se bliže informacije o značenju neke reči. Većina osobina reči koja su interesantna za obradu prirodnog jezika, nisu u potpunosti pokrivena u rečnicima koje mašine mogu da čitaju. Razlog je produktivnost prirodnog jezika. Stalno se pronalaze nove reči ili se stare reči koriste sa novim značenjem. Čak i kada bi se napravio rečnik koji bi potpuno pokrivao jezik današnjice, za nekoliko meseci taj rečnik bi neizbežno postao nekompletan. Grupisanjem se pomaže pronalaženje osobina i značenja reči, što je značajno u inteligentnim računarskim sistemima, gde je lakše računaru opisati povezanost sa drugim rečima nego značenje same reči.

LITERATURA

[1] Thomas C. Rindflesch: "Natural Language Processing", Sematic Knowledge Representation research paper, USA, 1996.

[2] L. Budin, B.Dalbelo Bašić, S. Ribarić, N. Pavešić: "Inteligentni sustavi", međunarodna konferencija MIPRO Opatija, 2001.

[3] Christopher D. Manning, Hinrich Schütze: „Foundation of Statistical Natural Language Processing", Massachusetts Institute of Technology, London, Second printing with correction 2000.

[4] Ian H.Witten and Eibe Frank: „Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann Publishers, Second Edition; 2005.

[5] Dan W. Patterson: „Introduction to Artificial Intelligence and Expert Systems", Prentice Hall, 1990.

ABSTRACT

For understanding of natural language by automatic system, the meaning has to be represented in way which can be used by the system. Finding measure of similarity of words is more easier then determine the exact meaning. This paper describe implementation of k-Means algorithm for grouping words based on semantic similarity.

AUTOMATIC CLUSTERING OF SEMANTICALLY SIMILAR WORDS USING k-MEANS ALGORITHM

Atila Farkaš.