MODELI JEZIKA

IVAN POSAVČEVIĆ,

ZAGREB, SIJEČANJ 2015.

Obrada prirodnog jezika

- Homo sapiens – sposobnost govora
- Prije 100000 godina govor, prije 7000 godina pismo
- Komunicikacija preko neograničenog broja kvalitativno različitih poruka
- Turingov test temelji se na jeziku
- Želimo da kompjuterski agenti mogu obrađivati prirodni jezik – komunikacija & informacije
- Pitamo se kako agenti pribavljaju informacije iz pisanog jezika
- Agent koji želi steći znanje mora razumjeti dvosmislene i zbrkane jezike
- Modeli jezika – modeli koji predviđaju vjerojatnosnu razdiobu jezičnih izraza

Modeli jezika (1)

- Formalni jezici (npr. Phyton) imaju precizno definirane modele jezika
- Jezik može biti definiran kao skup strinogova
- Primjer: „print(2+2)” je legalan program u Phytonu, dok „2)+(2 print”
- Legalni programi određeni su skupom pravila koji se zove gramatika
- Postoje pravila koja definiraju semantiku programa
- Primjer: značenje od „2+2” jest 4, značenje od „1/0” jest greška

Modeli jezika (2)

- Prirodni jezici ne mogu biti karakterizirani kao konačan skup rečenica
- Primjer: „Moj prijatelj je dobar.” → to je rečenica hrvatskog jezika, ali nije gramatički točna
- Bolje je definirati prirodan model jezika kao vjerojatnosnu razdiobu nad rečenicama
- Primjer: umjesto da ispitujemo pripada li string „prijatelj” skupu koji definira jezik, ispitujemo koja je vjerojatnost da je slučajno izabrana rečenica „prijatelj”
- Dvosmislenost prirodnih jezika: npr. „Idem van s Ivanom.”
- Vjerojatnosna razdioba nad mogućim značenjima
- Opsežnost i promjenljivost prirodnih jezika – problem!
- Naši modeli jezika uglavnom su aproksimacije

N-gram znakovni modeli (1)

- Napisani tekst sastavljen je od znakova (slova, brojke, interpunkcija, razmaci…)
- Vjerojatnosna distribucija nad nizovima znakova
- Pišemo $P(c_{1:N})$ za vjerojatnost rečenice od N znakova $c_1 - c_N$
- Primjer: na jednoj web stranici na engleskom jeziku jest P(„the”) = 0.027 (realno!)
- N-gram = niz zapisanih simbola duljine N
- Posebni nazivi: unigram, bigram, trigram
- N-gram model = model za vjerojatnosnu distribuciju za N-slovnu rečenicu

N-gram znakovni modeli (2)

- N-gram model definiran je kao Markovljev lanac reda n-1
- Markovljeva pretpostavka: vjerojatnost znaka c_i ovisi samo o neposredno prethodnim znakovima
- Primjer: u trigram modelu je $P(c_i | c_{1:i-1}) = P(c_i | c_{i-2|i-1})$
- U trigram modelu: $P(c_{1:N}) =$ (lančano pravilo) $= \prod_{i=1}^{N} P(c_i | c_{1:i-1}) =$ (Markovljeva pretpostavka)
  $= \prod_{i=1}^{N} P(c_i | c_{i-2:i-1})$
- Za trigram znakovni model u jeziku od 100 znakova, $P(c_i | c_{i-2|i-1})$ ima 10^6 ulaza i može biti točno procijenjen brojanjem nizova znakova u tijelu teksta od 10 milijuna znakova ili više
- Tijelo teksta = corpus (lat.)
- „Trening" corpus

Identifikacija jezika (1)

- Relativno lagan zadatak za koji su pogodni N-gram modeli
- Problem: blisko povezani jezici, npr. hrvatski i srpski
- Prvo se konstruira trigram znakovni model za svaki jezik koji je kandidat
- $P(c_i | c_{i-2|i-1}, L)$, gdje varijabla L varira od jezika do jezika
- Za svaki L model se konstruira brojanjem trigrama u corpusu toga jezika
- Potrebno oko 100000 znakova iz svakoga jezika
- Dobivamo model P(tekst|jezik)

Identifikacija jezika (2)

- Želimo odabrati jezik koji ima najveću vjerojatnost za dani tekst
- $\mathrm{L}^* = \underset{L}{\operatorname{argmax}}\, P(L|\mathrm{c}_{1:\mathrm{N}}) = \underset{L}{\operatorname{argmax}}\, P(L) * P(\mathrm{c}_{1:\mathrm{N}}|\mathrm{L}) = \underset{L}{\operatorname{argmax}}\, P(L) * \prod_{i=1}^{N} P(c_i|c_{i-2:i-1}, L)$
- U gornjoj formuli koristili smo Bayesovo pravilo i Markovljevu pretpostavku
- Problem: Kako odrediti početnu vrijednost P(L)?
- Za slučajno odabranu web-stranicu znamo da je engleski najvjerojatniji jezik i P(albanski) < 0.01
- Odgovor: Točan broj za P(L) nije bitan jer trigram model obično bira jezik koji je nekoliko redova veličine više vjerojatan nego ijedan drugi

Druge zadaće za znakovne modele

- Uključuju pravopisni ispravak, klasifikaciju žanra i prepoznavanje imena
- Klasifikacija žanra – kojem žanru pripada dani tekst
- Otvoren problem: brojanje interpunkcijskih znakova i drugih obilježja N-gram znakova
- Prepoznavanje imena = zadaća pronalaženja imena objekata u dokumentu i određivanja kojoj klasi pripadaju
- Primjer: „Mr. Sopersteen was prescribed aciphex.” → za ovu zadaću dobri su tzv. modeli znakovnih razina

N-gram modeli poravnavanja (1)

- Trening corpus omogućuje samo procjenu stvarne vjerojatnosne razdiobe
- Primjer: „_th” i „_ht” → ako stavimo P(„_ht”) = 0, onda bi tekst koji uključuje riječ „http” imao za engleski jezik vjerojatnost 0
- Zahtjev: modeli jezika trebaju se dobro poopćavati na tekstove koje još nisu vidjeli
- Prilagodba modela: nizovima koji imaju broj 0 u trening corpusu dodjeljujemo malu ne-nula vjerojatnost, a ostale vrijednosti prilagodimo blago silazno tako da vjerojatnost u sumi daje 1
- Poravnavanje (engl. smoothing) = proces prilagođavanja vjerojatnosti za vrijednosti s niskom frekvencijom

N-gram modeli poravnavanja (2)

- Laplaceov tip poravnavanja: ako je proizvoljna Booleova varijabla X bila false u svih N opažanja dotad, onda procjena za P(X = true) treba biti $\frac{1}{N+2}$ (ovo pojasniti)
- Poznato i pod imenom „dodaj-jedan" poravnavanje
- Bolji pristup → tzv. backoff model: krećemo od procjena N-gram vrijednosti, ali za svaki niz koji ima malenu (ili 0) vrijednost, vraćamo se na (N-1)-grame
- Još bolje: podrezivanje linearnom interpolacijom – backoff model koji kombinira unigram, bigram i trigram modele linearnom interpolacijom (vidi idući slajd)

Podrezivanje linearnom interpolacijom

- $P^*(c_i | c_{i-2:i-1}) = \lambda_3 P(c_i | c_{i-2:i-1}) + \lambda_2 P(c_i | c_{i-1}) + \lambda_1 P(c_i)$
- $\lambda_3+\lambda_2+\lambda_1 = 1$
- Vrijednost parametara λ_i mogu biti fiksne ili određene algoritmom „očekivanje-maksimizacija”
- Moguće je da vrijednosti λ_i-ova ovise o tome koliki je broj trigrama/bigrama/unigrama prisutan u danom tekstu

N-gram modeli poravnavanja (3)

- Jedna skupina znanstvenika razvija što je moguće više sofisticirane modele poravnavanja
- Druga skupina predlaže stvaranje što većeg corpusa tako da i najjednostavniji modeli rade dobro
- Cilj obje skupine = smanjiti varijancu u modelu jezika
- Problem: izraz $P(c_i | c_{i-2:i-1})$ zahtijeva $P(c_1 | c_{-1:0})$ za i=1, ali ne postoje znakovi prije c_1
- Uvođenje artificijelnih znakova: npr. c_0 definiramo kao razmak ili poseban znak za početak teksta
- Alternativno, možemo definirati $c_{-1:0}$ da bude prazan niz pa je tada $P(c_1 | c_{-1:0}) = P(c_1)$

Evaluacijski model

- Kako znamo koji od N-gram modela izabrati?
- Ocjena modela pomoću tzv. unakrsne validacije: podijelimo corpus u trening corpus i validacijski corpus → odredimo parametre modela iz trening corpusa → ocjenjujemo model na validacijskom
- Ta ocjena može biti specifična za zadatak, kao što je mjerenje točnosti pri identifikaciji jezika
- Možemo imati model kvalitete jezika neovisan o zadatku – izračunati vjerojatnost pridruženu validacijskom corpusu prema modelu
- Perplexitiy (prijevod: zbunjenost) = mjera koja označava koliko dobro vjerojatnosna razdioba ili vjerojatnosni model predviđa uzorak
- Perplexitiy($c_{1:N}$) = $P(c_{1:N})^{(1/N)}$ → recipročna vrijednost vjerojatnosti, normalizirana prema duljini uzorka; ponderirani prosječni faktor grananja modela

N-gram modeli riječi (1)

- Rječnik (vokabular) – skup simbola koji tovre corpus i model
- Većina jezika ima samo oko 100 znakova
- Ponekad konstruiramo znakovne modele koji su još više restriktivni, npr. treitramo „A” i „a” kao isti simbol ili tretiramo sve interpunkcijske znakove kao isti simbol
- S druge strane, kod modela riječi imamo barem na desetke tisuća simbola → nije jasno što točno tvori riječ u pojedinom jeziku
- N-gram modeli riječi moraju se „suočavati” s riječima koje su izvan rječnika

N-gram modeli riječi (2)

- Mogućnost pojave nove riječi koja nije bila viđena u trening corpusu
- U rječnik se dodaje jedna nova riječ koja predstavlja nepoznatu riječ: <UNK>
- Kako procjeniti N-gram vrijednosti <UNK>?
- Odgovor: prolazimo kroz trening corpus i kad se bilo koja riječ pojavi prvi puta, ona je prethodno nepoznata pa je zamijenimo sa <UNK>. Sve sljedeće pojave te riječi ostaju nepromijenjene. Zatim prebrojimo N-gram vrijednosti u corpusu kao inače, tretirajući <UNK> kao bilo koju drugu riječ. Ako se neka nepoznata riječ pojavi u testnom skupu, tražimo njezinu vjerojatnost pod <UNK>.
- Ponekad se koriste višestruki simboli za nepoznate riječi
- Primjer: proizvoljan string znamenaka zamijenimo sa <NUM>, a e-mail adrese sa <EMAIL>

N-gram modeli riječi (3)

- Da bismo dobili osjećaj što modeli riječi mogu raditi, autori Knjige konstruirali su unigram, bigram i trigram modele nad riječima u Knjizi, a zatim nasumično odabrali nizove riječi iz modela.
- Rezultati:
  - Unigram: logical are as are confusion a may right tries agent goal the was…
  - Bigram: systems are very similar computational approach would be represented…
  - Trigram: planning and scheduling are integrated the success of naive bayes model is…
- Unigram model slabo aproksimira engleski jezik i sadržaj Knjige
- Bigram i trigram modeli puno su bolji

Završna riječ

- U ovom kratkom pregledu utvrdili smo osnove N-gram modela – modela znakova i modela riječi
- Naglasak poglavlja = kako pribavljati informacije iz pisanoga jezika
- Modeli jezika dobar su pristup u realizaciji te zadaće
- Sljedeći je korak pogledati neke jezične zadatke

Literatura

- S. J. Russell, P. Norvig: Artificial Intelligence, A modern Approach, Third Edition
- N. Sarapa: Teorija vjerojatnosti
- www.math.pmf.unizg.hr/~singer/ui - Materijali za predavanja i vježbe
- www.wikipedia.org