SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1095

Jezični model temeljen na neuronskoj mreži

Florijan Stamenković

Zagreb, svibanj 2015.

SADRŽAJ

1. Uvod

2. Jezični model

Jezični model je računalni sustav koji može ocijeniti "smislenost" teksta napisanog prirodnim jezikom. Glavna sposobnost sustava je da razlikuje smislene nizove riječi od besmislenih. Primjerice, model hrvatskog jezika trebao bi biti sposoban ocijeniti da je niz *"dan je lijep"* više smislen od *"lijepo dana biti"*.

Smislenost niza riječi može se promatrati iz pravopisne, gramatičke i semantičke perspektive. Dakle, model treba biti sposoban nizu *"lijep dan"* dati bolju ocjenu nego nizu *"ljep dan"*. Nadalje, treba bolje ocijeniti niz *"lijep dan"* od niza *"lijepo dani"*. Konačno, treba bolje ocijeniti niz *"lijep dan"* od niza *"staklen plan"*.

2.1. Primjena

Jezični modeli imaju velik broj primjena. Sustavi za automatsko prevođenje koriste ih kako bi odabrali što smisleniji među više potencijalnih prijevoda. Sustavi za prepoznavanje govora koriste ih kako bi prepoznate slogove i riječi pretvorili u konačni tekst. Sustavi za pretraživanje teksta mogu ih koristiti za mjerenje sličnosti između termina pretrage i dokumenta. Ovo su samo neke od mnogih primjena, povećanjem dostupne količine informacija i mogućnosti njihove računalne obrade, moguće primjene i kvaliteta jezičnih modela imaju trend povećanja.

2.2. Probabilistička definicija

Pošto prirodni jezik omogućava praktično neiscrpne mogućnosti kombiniranja riječi u tekst, smisleno je jezični model razmatrati statistički. Na temelju što većeg korpusa teksta gradi se sustav koji za proizvoljni niz označava koliko je sličan tekstu iz korpusa. Kako bi se takav sustav izgradio, potrebno je smislenost teksta definirati probabilistički. Vjerojatnost niza riječi možemo zapisati na sljedeći način.

$$P(\text{niz}) = P(w_1, w_2, ..., w_m)$$

Pri tome je $P(w_1, w_2, ..., w_m)$ vjerojatnost slijeda od m riječi u kojem je i-ta riječ upravo w_i. Tako definirana vjerojatnost može se faktorizirati.

$$\begin{aligned} P(w_1, w_2, ..., w_m) &= P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)...P(w_m|w_1, w_2, ..., w_{m-1}) \\ &= \prod_{i=1}^{m} P(w_i|w_1, ..., w_{i-1}) \end{aligned}$$

Pri tome je $P(w_i|w_1, ..., w_{i-1})$ uvjetna vjerojatnost pojavljivanja riječi w_i na i-toj poziciji u nizu, ako su joj prethodile riječi w_1, ... w_{i-1} na pozicijama 1, ... , $i - 1$. Primjetimo kako je ovakva faktorizacija u skladu sa čovjekovim čitanjem teksta (slijedno po riječima).

Iako egzaktna, dobivena faktorizacija vjerojatnosti niza praktična jer je vjerojatnost riječi na poziciji i uvjetovana svim prethodnim riječima. Pošto tekst može biti proizvoljne duljine, u jezičnim modelima se najčešće koristi kontekst ograničene duljine od n riječi. Primjerice, za kontektst duljine $n = 3$, svaka riječ se razmatra samo s obzirom na prethodne dvije.

$$\begin{aligned} P(w_1, w_2, ..., w_m) &\approx P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)...P(w_m|w_{m-2}, w_{m-1}) \\ &\approx \prod_{i=1}^{m} P(w_i|w_{i-2}, w_{i-1}) \end{aligned}$$

Općenito, za proizvoljnu duljinu konteksta *n* faktorizacija je sljedeća.

$$\begin{aligned} P(w_1, w_2, ..., w_m) &\approx P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)...P(w_m|w_{m-(n-1)}, w_{m-(n-2)}, ..., w_{m-1}) \\ &\approx \prod_{i=1}^{m} P(w_i|w_{i-(n-1)}, ..., w_{i-1}) \end{aligned}$$

Ovakav jezični model više nije probabilistični egzaktan, ali se može puno lakše implementirati kao računalni sustav.

U nastavku ovog rada se slijed od *n* riječi označava *n*-gram. Za $n = 1, 2, 3$ govorimo o "unigramu", "bigramu", odnosno "trigramu", dok za $n = 4$ o "4-gramu" itd.

Kada se koristi ograničeni kontekst, javlja se pitanje optimalne duljine. S obzirom da na konačno značenje niza mogu utjecati riječi koje su od promatrane riječi proizvoljno daleko, može se pretpostaviti da je veći kontekst bolji od manjeg. Iako ovo u idealnom slučaju vrijedi, u praksi se koriste relativno male duljine konteksta, iz sljedećeg

razloga. Vjerojatnost pojavljivanja nekog konkretnog niza riječi duljine n pada eksponencijalno kako n raste. Primjerice, ako promatramo bigrame, tada će se većina smislenih kombinacija od dvije riječi pojaviti u korpusu teksta na kojem se model bazira. U drugu ruku, ako razmatramo 100-grame, većina kombinacija od 100 riječi nikada se neće pojaviti u ograničenom korpusu teksta, što uključuje i smislene 100-grame. Kada se potom takav 100-gram pojavi pri korištenju modela na konkretnom zadatku, model će mu neopravdano pridjeliti vrlo malo vjerojatnost. U praksi se stoga često koriste konteksti duljine do četiri riječi, ovisno o primjeni. Modeli bazirani na velikom korpusu teksta mogu povećati kontekst do šest ili sedam riječi, rijetko više.

Alternativni pristup ograničavanju veličine konteksta je kompresija odnosno aproksimacija konteksta pune duljine. Konkretno, ako kontekst prikažemo kao funkciju prethodnih riječi, moguća je sljedeća faktorizacija.

$$\begin{aligned} P(w_1, w_2, ..., w_m) &\approx P(w_1)P(w_2|f(w_1))P(w_3|f(w_1, w_2))...P(w_m|f(w_1, w_2, ..., w_{n-1})) \\ &\approx \prod_{i=1}^{n} P(w_i|f(w_1, ..., w_{i-1})) \end{aligned}$$

Pri tome funkcija $f(...)$ preslikava kontekst proizvoljne duljine u zapis fiksne duljine. Ovakva formulacija koristi se primjerice u izvedbi jezičnih modela korištenjem rekurzivnih neuronskih mreža. Pošto se u ovom radu ne koriste implementacije bazirane na toj formulaciji, neće se u ostatku teksta spominjati.

Postoje varijante jezičnih modela koje kao kontekst ne promatraju riječi koje doslovno prethode promatranoj, već riječi iz šire okoline. Neki modeli čak razmatraju "koncepte" sačinjene od nekoliko riječi koje se mogu pojaviti bilo gdje u tekstu. Ovakvi modeli se neće razmatrati u nastavku.

Konačno, bitno je spomenuti jezične modele bazirane na manjim lingvističkim jedinicama. Jezični modeli bazirani na morfemima i znakovima su od nedavno, pogotovo sa sve većom primjenom dubokih neuronskih mreža, postali uspooredivi i čak u nekim primjenama bolji od leksičkih modela.

2.3. Načini izvedbe

Postoje brojne implementacije jezičnih modela. Većina njih izvedena je iz probabilističke formulacije modela, ili tu formulaciju aproksimiraju. Tehnike korištene za izradu modela variraju. U ovom diplomskom radu biti će uspoređene izvedbe prebrojavan-

jem *n*-grama, oblik neuronske mreže i log-bilinearni probabilistički model. Njihova definicija i opis slijede.

3. Prebrojavanje *n*-grama

Najjednostavniji pristup implementaciji jezičnog modela bazira se na frekvenciji *n*-grama. Definicija za uvjetnu vjerojatnost riječi baziranu na primjerice trigramima je sljedeća.

$$P(w_i|w_{i-2}, w_{i-1}) = \frac{C(w_{i-2}, w_{i-1}, w_i)}{C(w_{i-2}, w_{i-1})}$$

Pri tome je $C(w_a, ..., w_b)$ funkcija prebrojavanja koja za niz $w_a, ..., w_b$ daje broj njegovih pojavljivanja u korpusu teksta nad kojim se model gradi. Uvjetna vjerojatnost pojavljivanja riječi, s obzirom na prethodne dvije, je dakle broj pojavljivanja relevantnog trigrama, podjeljeno s brojem pojavljivanja bigrama koji joj prethode. Intuicija je sljedeća. Ako smo u tekstu obzervirali da se trigram *"dan je lijep"* pojavljuje 10 puta, a bigram *"dan je"* 12 puta, tada je $P("lijep"|"dan","je") = 10/12$.

Ovako definiran procijenitelj vjerojatnosti zapravo je procijenitelj najveće izglednosti (ML-procijenitelj, engl. *maximum likelihood*). Općenito razmatranje statističkih procjenitelja je izvan opsega ovog diplomskog rada, više informacija na temu može se naći u materijalima na temu statistike i strojnog učenja.

Definirana je uvjetna vjerojatnost za kontekst duljine $n = 3$, formulaciju je potrebno poopćiti. Uvjetna vjerojatnost riječi za kontekst proizvoljne duljine n je sljedeća.

$$P(w_i|w_{i-(n-1)}, ..., w_{i-1}) = \frac{C(w_{i-(n-1)}, ..., w_{i-1}, w_i)}{C(w_{i-(n-1)}, ..., w_{i-1})} \tag{3.1}$$

3.1. Zaglađivanje

Već je spomenuto kako je otežavajuć faktor izgradnje jezičnih modela to što se baziraju na nekom konačnom korpusu teksta. Spomenuto je kako je to jedan od ograničavajućih čimbenika za veličinu konteksta modela. Ovaj problem zapravo nije ograničen samo na modele koji koriste veliki kontekst. Čak i za male vrijednost n, primjerice 3, lako se može desiti da neki smisleni trigram ne bude prisutan u korpusu na kojem se model

bazira. Ovo nije začuđujuće, s obzirom da je broj mogućih trigrama $|V|^3$, gdje je V vokabular jezika za koji se model gradi, odnosno skup svih riječi tog jezika. Primjerice, za engleski se jezik procjenjuje da sadrži oko 300000 riječi[1], što znači da je broj mogućih trigrama oko $27 * 10^{15}$. Jasno je da se među njima nalaze mnogi smisleni trigrami koji se u korpusu nad kojim gradimo model ne pojavljuju. U slučaju da se pri korištenju modela evaluira vjerojatnost jednog od njh, po navedenoj definiciji ona će biti 0. Numerički i intuitivno vjerojatnost 0 govori kako je nešto nemoguće, što se svakako želi izbjeći, pogotovo ako se radi smislenom *n*-gramu.

Ovaj problem se rješava zaglađivanjem (engl. *smoothing*). Ideja je da se u račun vjerojatnosti unese pretpostavka kako je korpus na kojem se model bazira samo uzorak jezika. Pretpostavivši da postoje mnogi *n*-grami koji nisu prisutni u korpusu, ali zavređuju određenu vjerojatnost, njen račun se modificira kako bi se vjerojatnosna masa šire rasporedila. Time se disproporcije u vjerojatnostima *n*-grama smanjuju, odnosno zaglađuju.

Postoji više pristupa zaglađivanju, neki od popularnijih su aditivno (Laplace), Good-Turing i Knesser-Ney zaglađivanje. Aditivno zaglađivanje u pravilu daje lošije rezultate, koristi se jer je iznimno jednostavno. Druge dvije spomenute metode nisu računalno puno kompleksnije, ali nisu toliko intuitivne. Njihov opis izlazi izvan okvira ovog rada.

3.1.1. Aditivno zaglađivanje

Ideja aditivnog zaglađivanja je da se svakom mogućem *n*-gramu pridjeli neka mala vjerojatnost, makar se taj *n*-gram nije pojavio u korpusu za treniranje. U izrazu vjerojatnosti mora se voditi računa da suma vjerojatnosti svih mogućih *n*-grama bude 1. Uzevši to u obzir dobivamo sljedeće.

$$P(w_i|w_{i-(n-1)}, ..., w_{i-1}) = \frac{C(w_{i-(n-1)}, ..., w_{i-1}, w_i) + \alpha}{C(w_{i-(n-1)}, ..., w_{i-1}) + \alpha d}$$

Pri tome je α proizvoljan broj, tipično 1 ili manje, a d broj svih mogućih *n*-grama. Koristeći ovakvu formulaciju izbjegava se da vjerojatnost bilo kojeg *n*-grama bude 0. Nedostatak ovog pristupa je to što na umjetni način svim *n*-gramima povećava broj pojavljivanja za isti broj α, bez obzira na njihovu relativnu smislenost. Modificiranjem vjerojatnosti kvalitetnijim pretpostavkama drugi oblici zaglađivanja u pravilu postižu bolje rezultate.

[1] Oxford English Dictionary, http://www.oed.com

3.2. Implementacija

Praktična izvedba prebrojavanja *n*-grama je konceptualno jednostavna. Tekst treba pretvoriti u *n*-grame i za svaki *n*-gram ustanoviti broj pojavljivanja u korpusu. Kompleksnije metode zaglađivanja mogu zahtjevati dodatne strukture podataka, ali one nisu kompliciranije za izvedbu. Jedini problem prebrojavanja *n*-grama je njihova brojnost.

Već je spomenuto kako je za engleski jezik broj mogućih trigrama okvirno $27*10^{15}$. Ako bismo za svaki od tih trigrama pohranili broj pojavljivanja u samo dva okteta računalne memorije, bilo bi nam potrebno oko 9 milijuna *terrabyte*-a memorije. Očigledno je ovo sa današnjim računalima neizvedivo. Problem je potrebno rješiti drukčije. Moramo uzeti u obzir da je većina tih trigrama iznimno malo vjerojatno i doslovce besmisleno. Oni se nikada ne javljaju u tekstu te je nepotrebno trošiti memoriju na njih. Stoga se prebrojavanje *n*-grama u pravilu izvodi korištenjem rijetkih (engl. *sparse*) zapisa, pamte se brojevi pojavljivanja samo onih *n*-grama koji su se barem jednom pojavili. Ovaj pristup uspješnije rješava problem, ali je kompliciraniji. Rijetke strukture podataka teže se implementiraju, a memorijske i brzinske performanse im ovise o brojnim čimbenicima. Kvalitetno upoznavanje takvih struktura preporučeno je pri pokušaju implementacije jezičnih modela baziranih na prebrojavanju *n-grama*.

4. Neuronske mreže

5. Ograničeni Boltzmannov stroj

5.1. Učenje

5.2. Implementacija

6. Log-bilinearni model

6.1. Učenje

6.2. Implementacija

7. Evaluacija

7.1. Korišteni podatci

7.2. Način evaluacije

7.3. Rezultati

8. Zaključak

LITERATURA

9. Sažetak