Ekstrakcija likova iz kratkih priča

Gorana Levačić Tomislav Horina

***Sažetak*—U ovom radu smo implementirali klasifikator likova u kratkim pričama. Koristili smo kratke priče s *Project Gutenberga*, online repozitorija besplatnih knjiga. Priče smo samostalno označavali jer nismo naišli na odgovarajući skup podataka na Internetu, a postojeća rješenja ne prepoznaju na primjer kralja kao lika. Promatrali smo ponašanje klasifikatora na četiri načina zadavanja skupa za treniranje. Vrste su:**

1) **bez ikakve promjene**
2) **samo riječi bez interpukcijskih znakova**
3) **riječi pisane malih slovima**
4) **riječi pisane malim slovima i bez interpukcijskih znakova**

Isprobali smo algoritme klasifikacije koji su preporučeni za izradu tog modela – skriveni Markovljev model(eng. *Hidden Markov Model, HMM*) i uvjetna slučajna polja(eng. *Conditional Random Fields, CRF*). Također smo istrenirali i Stanford NER model.

I. Uvod

U ovom projektu bavit ćemo se ekstrahiranjem likova iz kratkih priča, konkretno priča za djecu. Taj problem pripada problemu ekstrakcije, odnosno identifikacije entiteta u tekstu, poznatiji pod engleskim nazivom named-entity recognition (NER). NER je podvrsta zadaće crpljenja obavijesti (information extraction), u kojoj se svakom elementu teksta pridjeljuje neki atribut. U općem slučaju imamo više atributa, na primjer osoba, lokacija, vrijeme, iznos novca i drugi, te više riječi može činiti entitet kojem se pridjeljuje jedan atribut. Jasnije je iz sljedećeg primjera:

Jim bought 300 shares of Acme Corp. in 2006.

$|Jim|_{person}$ bought 300 shares of $|AcmeCorp.|_{organization}$ in $|2006|_{time}$.

U našem slučaju imamo samo jedan atributa, lik (character), koji određuje tko su sve likovi u priči.

II. Podaci

S web stranice *Project Gutenberg* smo skinuli desetak dječjih knjiga koje sadrže različite priče, bajke i mitove. Potom smo napisali skriptu koja je iz tih knjiga vadila pojedinačne priče i spremala ih u zasebne dokumente.
Sljedeći korak je bilo pisanje skripte koja je iz tih priča stvara *tab separated value (.tsv)* dokumente koji imaju dva stupca te onoliko redaka koliko riječi i interpunkcijskih znakova ima priča. U njima se u prvom stupcu nalazi oznaka *O (other)*, a u drugom stupcu se nalazi riječ, odnosno znak (token) kojem ta oznaka pripada.

Posljednji korak u pripremi podataka je bilo samostalno označavanje likova. To smo bili primorani napraviti pošto postojeća rješenja, kao što je Stanford NER, ne prepoznaju likove koji nemaju vlastito ime, kao što su *kralj, kraljica, mačak* i slično. Svakoj riječi koja predstavlja lika smo oznaku promijenili u *C (character)*.

U našim modelima gledat ćemo utječu li na njega velika početna slova osobnih imena i postojanje interpunkcijskih znakova. Zato ćemo imati četiri vrste podataka za učenje:

1) nepromjenjene riječi i interpunkcijski znakovi
2) samo nepromjenjene riječi, bez interpukcijskih znakova
3) riječi pisane malim slovima i interpunkcijski znakovi
4) samo riječi pisane malim slovima, bez interpukcijskih znakova

III. Opis korištenih meotda

Koristili smo algoritme klasifikacije koji su preporučeni za izradu tog modela :

1) Skriveni Markovljev model (eng. *Hidden Markov Models, HMM*)
2) Uvjetna slučajna polja (eng. *Conditional Random Fields, CRF*)

Dodatno smo učili Stanford NER na našim podacima.

A. Skriveni Markovljev model

Skriveni Markovljev model (HMM) prvog reda jest skup slučajnih varijabli koji se sastoji od dva podskupa, Q i O :

- $Q = Q_1, \ldots, Q_N$ – skup slučajnih varijabli koje poprimaju diskretne vrijednosti
- $O = O_1, \ldots, O_N$ – skup slučajnih varijabli koje poprimaju diskretne ili kontinuirane vrijednosti.

Te varijable zadovoljajvaju sljedeće uvjete:

$$\mathbb{P}(Q_t|Q_{t-1}, O_{t-1}, \ldots, Q_1, O_1) = \mathbb{P}(Q_t|Q_{t-1}) \tag{1}$$

$$\mathbb{P}(O_t|Q_T, O_T, \ldots, Q_{t+1}, O_{t+1}, Q_t, Q_{t-1}, O_{t-1}, \ldots, Q_1, O_1) = \mathbb{P}(O_t|Q_t) \tag{2}$$

Relacija (1) kaže da je vjerojatnost da se, za $t \in \{1, 2, \ldots, N\}$, nalazimo u stanju Q_t uz uvjet da su se dogodila prethodna stanja $Q_1, \ldots, Q_{t-1}$ i da su emitirani simboli $O_1, \ldots, O_{t-1}$ jednaka tranzicijskoj vjerojatnosti iz stanja Q_{t-1} u stanje Q_t.

Relacija (2) povlači da realizacija nekog opažanja u sadašnjem stanju ovisi samo o tom stanju. Vjerojatnosti iz relacije (2) nazivamo emisijske vjerojatnosti i kažemo da stanje Q_t emitira simbol O_t.

Skriveni Markovljev model zadan je sljedećim parametrima:

- N – broj stanja u kojima se proces može nalaziti
  $S = \{1, \ldots, N\}$ – skup svih stanja procesa

- M – broj mogućih opažanja
  $B = \{b_1, \ldots, b_M\}$ – skup svih opaženih vrijednosti
- L – duljina opaženog niza
  $X = (x_1, \ldots, x_L)$ – opaženi niz
- $A = \{a_{ij}\}$ – matrica tranzicijskih vjerojatnosti, pri čemu je $a_{ij} = \mathbb{P}(Q_{t+1} = j | Q_t = i), 1 \leq i, j \leq N$
- $E = \{e_j(k)\}$ – matrica emisijskih vjerojatnosti, pri čemu je $e_j(k) = \mathbb{P}(O_t = b_k | Q_t = j), 1 \leq j \leq N, 1 \leq k \leq M$

B. Uvjetna slučajna polja

Uvjetna slučajna polja (CRF) je diskriminativni vjerojatnosni model strojnog učenja za struktuiranu predikciju koja je temeljena na modelima neusmjerenih grafova.

Neka je dan vektor $x = \{x_1, x_2, \ldots, x_T\}$, gdje je T dužina sekvence, a svaki x_i predstavlja vektor karakteristika podataka na poziciji i. Za svaki od tih vektora potrebno je odrediti oznaku (tag) y_i. U slučaju određivanja vrste riječi, T bi bio broj riječi u tekstu, y_i vrsta riječi na poziciji i , a za svaki x_i bi bio vektor informacija o i-toj riječi kojima se opisuju sama riječ, informacije o prefiksima, sufiksima i veličini slova i drugo.

Neka su x i Y slučajne varijable, $w = \{w_k\}$ realni vektor parametara i $F = \{f_k(y_t y_{t-q}, x_t)\}_{k=1}$ skup realnih karakterističnih funkcija. Linearna uvjetna slučajna polja definiraju se uvjetnom raspodjelom

$$P(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{T} exp \left[\sum_{k=1}^{K} w_k f_k(y_t, y_{t-q}, x_t) \right]$$

pri čemu je

$$Z(x) = \sum_{y} \prod_{t=1}^{T} exp \left[\sum_{k=1}^{K} w_k f_k(y_t, y_{t-q}, x_t) \right]$$

Karakteristična funkcija se može promatrati proizvoljne karakteristike sekvence, tako da se vektor x_t, zapravo može zamijeniti cjelokupnom sekvencom opservacija x. Linearna uvjetna slučajna polja omogućavaju promatranje samo dvije uzastopne oznake, zbog čega se model može promatrati kao lanac. Linarna uvjetna slučajna polja mogu se definirati i preko grafa nad skupom čvorova $U = X \cup Y$ kao

$$P(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{T} \Psi_t(y_t, y_{t-1}, x_t)$$

$$\Psi_t(y_t, y_{t-1}, x_t) = exp \left[\sum_{k=1}^{K} w_k f_k(y_t, y_{t-q}, x_t) \right]$$

Definirajmo opći oblik uvjetnih slučajnih polja.

Prvo definiramo **faktor graf** – bipartitni graf koji predstavlja faktorizaciju funkcije. Neka je $g(X_1, \ldots, X_n)$ funkcija čija faktorizacija je $\prod_{j=1}^{m} f_j(S_j)$, pri čemu je $S_j \subseteq \{X_1, \ldots, X_n\}$. Faktor graf funkcije G je graf $G = (X, F, E)$, pri čemu su $X = \{X_1, \ldots, X_n\}$ vrhovi varijable, $F = \{f_1, \ldots, f_m\}$ vrhovi faktori, te bridovi E određeni s:

$$e \text{ brid između } f_j \text{ i } X_k \iff X_k \in S_j$$

Konačno definiramo opći oblik uvjetnih slučajnih polja.

Neka je G faktor graf nad slučajnim varijablama X i y. (X, Y) je **uvjetno slučajno polje** ako se za neke vrijednosti x varijabli X, uvjetna vjerojatnost $P(y|x)$ modelira prema faktoru G. Svako uvjetno slučajno polje se modelira prema nekom faktoru grafa G. Dati faktor graf G i njegov skup faktora $\{\Psi_a\}$ definiraju raspodjelu

$$P(y|x) = \frac{1}{Z(x)} \prod_{i=1}^{T} \Psi_a(x_a, y_a)$$

gdje je

$$\Psi_a(x_a, y_a) = exp \left[\sum_{k=1}^{K} w_{ak} f_{ak}(x_a, y_a) \right]$$

U indeksiranju karakterističnih funkcija i odgovarajućih parametara sudjeluje i oznaka faktora a jer svaki faktor može imati svoj skup parametara. Faktor Ψ_a, na primjer, ovisi o uniji nekih podskupova $X_a \subseteq X$ i $Y_a \subseteq Y$. U slučaju linearnih uvjetnih slučajnih polja, skup Y_a je mogao sadržavati samo dvije susjedne oznake.

Skup svih parametara može se podjeliti na klase $C = \{C_1, C_2, \ldots, C_p\}$ gdje svaka klasa C_p koristi isti skup karakterističnih funkcija $\{f_{pk}(x_c, y_c)\}_{k=1,\ldots,K(p)}$ i vektor parametara w_{kp} veličine $K(p)$. Raspodjela vjerojatnosti se može napisati kao

$$P(y|x) = \frac{1}{Z(x)} \prod_{i=1}^{T} \Psi_a(x_a, y_a)$$

$$\Psi_c(x_c, y_c; w_p) = exp \left[\sum_{k=1}^{K} w_{pk} f_{pk}(x_c, y_c) \right]$$

C. Stanford Ner

Stanford NER je NER sustav kojeg je izradio *The Natural Language Processing Group* Sveučilišta u Stanfordu. *The NLP Group* je tim znanstvenika, predavača, programera i studenata koji zajednički rade na različitim algoritmima koji omogućuju računalima obradu i razumijevanje ljudskog jezika. Njihov rad je rezultirao nekim od najuspješnijih modela strojnog učenja, pogotovo za analizu teksta.

Stanford NER je još poznat i kao *CRFClassifier*. U pozadini koristi uvjetna slučajna polja te različite optimizacije i dodatna svojstva. Izvorni Stanford NER je treniran nad velikim količinama podataka te prepoznaje tri klase, *PERSON, ORGANIZATION, LOCATION*. Veoma je uspješan u prepoznavanju imenovanih entiteta, ali nije primjenjiv za naš problem u izvornom obliku jer prepoznaje samo imenovane likove, kao što su Alladin ili Arthur, a ne prepoznaje *kralja* ili *kraljicu*.

Unatoč tome, uspješno smo iskoristili Stanford NER za rješavanje našeg problema. Naime, moguće je i zadati skup podataka za treniranje vlastitog Stanford NER modela. Također je moguće zadati i parametre treniranja, kao što je duljina N-grama (niza od N riječi, odnosno konteksta) koje model trenutno promatra. Stanford NER pri treniranju koristi podatke u jednakom obliku kao i CRF i HMM modeli.

IV. Rezultati

A. Metode evaluacije

Kao mjeru za ocjenjivanje uspješnosti modela smo koristi F_2-mjeru, izračunate preko matrice konfuzije. **Matricu konfuzije** definiramo preko vrijednosti:

- **TP (true positive)** – riječi koje je model prepoznao kao likove, a koje doista označavaju likove
- **FP (false positive)** – riječi koje je model prepoznao kao likove, a koje ne označavaju likove
- **FN (false negative)** – riječi koje model nije prepoznao kao likove, odnosno označio ih je s *O (other)*, a koje zapravo označavaju likove
- **TN (true negative)** – riječi koje je model uspješno prepoznao da nisu likovi, odnosno označio ih s *O*

Matrica konfuzije je matrica oblika:

$$M = \begin{bmatrix} TP & FN \\ FP & TN \end{bmatrix}$$

Nadalje definiramo:

$$\text{preciznost (eng. } \textit{precision}) = P = \frac{TP}{TP + FP}$$

$$\text{osjetljivost (eng. } \textit{recall}) = R = \frac{TP}{TP + FN}$$

Konačno definiramo F_β mjeru kao:

$$F_\beta = (1 + \beta^2)\frac{\text{preciznost} \cdot \text{osjetljivost}}{\beta^2 \cdot \text{preciznost} + \text{osjetljivost}}$$

Parametar β određuje koliku važnost pridajemo osjetljivosti, odnosno preciznosti.

- Za $\beta = 0.5$ veću važnost u određivanju F_β vrijednosti ima preciznost.
- Za $\beta = 1$ jednaku važnost u određivanju F_β vrijednosti imaju i preciznost i osjetljivost.
- Za $\beta = 2$ veću važnost u određivanju F_β vrijednosti ima osjetljivost.

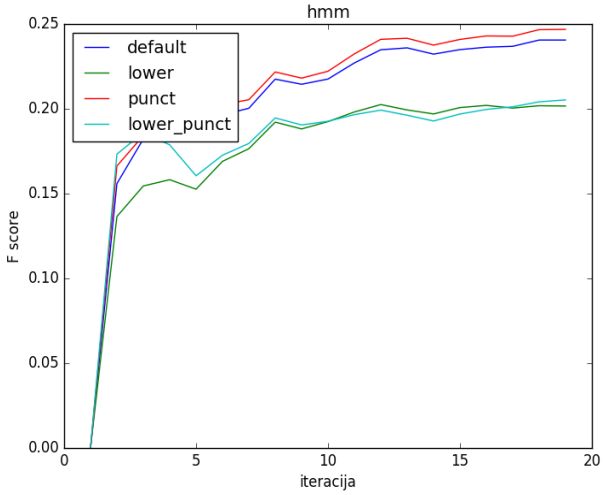
Mi smo odabrali $\beta = 2$, s obzirom da smo očekivali relativno loše rezultate, pa nam je važnije da su metode uspješno prepoznale neke od likove, nego da su doista označene riječi likovi.

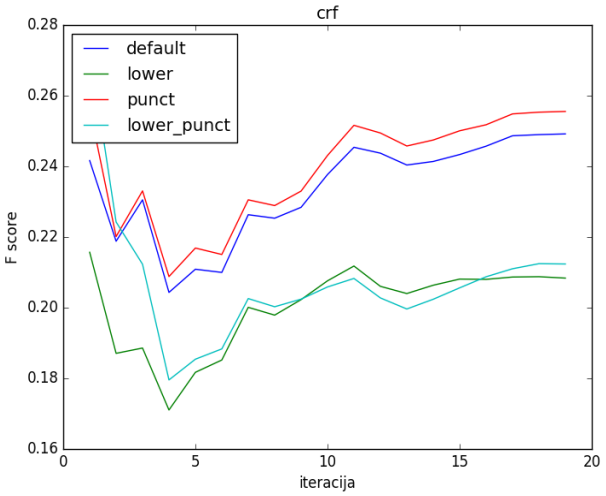
B. k-unakrsna validacija

Pri treniranju modela HMM i CRF smo koristili unakrsnu validaciju. Veličina podataka za učenje je 59 priča, a veličina skupa za unakrsnu validaciju je 3, što nam daje ukupno $k = 19$ modela. Odabrali smo skup za unakrsnu validaciju veličine 3 jer smo htjeli sačuvati što više podataka za učenje, dok s druge strane nismo htjeli da kvaliteta modela ovisi o jednoj priči koja slučajno najbolje odgovara modelu. Na primjer, model uspješno prepoznaje *queen, king* kao likove, ali lošije prepoznaje likove-životinje, pa bi se moglo dogoditi da priča o kralju i kraljici dobije jako dobre rezultate.

Pri treniranju Stanford NER modela, s obzirom da ga je potrebno trenirati preko konzole, manualno, nismo koristili unakrsnu validaciju.

Donji grafovi opisuju F_2 mjeru pri pojedinim iteracijama unakrsne validacije za HMM i CRF modele, za sve četiri varijante:





Objasnimo značenje svake od naznaka modela:

- Naznaka *default* označava da se u modelu ne koriste interpunkcijski znakovi te da se koristi izvoran oblik riječi (ne pisan malim slovima).
- Naznaka *punct* označava da se u modelu koriste interpunkcijski znakovi te da se koristi izvoran oblik riječi
- Naznaka *lower* označava da se u modelu koriste isključivo riječi pisane malim slovima te da nema interpunkcijskih znakova.
- Naznaka *lower punct* označava da se u modelu koriste isključivo riječi pisane malim slovima te da se koriste interpunkcijski znakovi.

Kao što vidimo, najuspješniji modeli pri unakrsnoj validaciji su bili oni koji su pri treniranju koristili i riječi i interpunkcijske znakove, te koji nisu koristili tekst pisan samo malim slovima. Modeli koji su koristili riječi pisane malim slovima su postigli znatno lošiji rezultat. Također uočavamo i da su CRF i HMM modeli postigli približno jednake rezultate nad skupom za unakrsnu validaciju.

C. Testiranje

Pri treniranju i unakrsnoj validaciji smo uočili da povećanje skupa za treniranje za samo nekoliko (5-7) priča rezultira primjetno boljim modelima. Stoga je skup priča za testiranje relativno malen te sadrži ukupno sedam nasumično odabranih priča.

Slijede matrice konfuzije za svaki od modela.

Prvi modeli koje smo testirali su četiri HMM modela. Iako su pokazali relativno dobre rezultate nad skupom za unakrsnu validaciju, nad testnim podacima su dali posve negativne rezultate – niti jednu riječ nisu označili kao lika. Donja tablica daje matricu konfuzije za sve četiri varijante. Naravno, pripadni F_2 score je nula.

HMM svi modeli		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	0	86
	Nisu likovi	0	5366

Sljedeći po redu su CRF modeli. Oni su se pokazali prihvatljivima nad testnim skupom, ali nisu dali dovoljno dobre rezultate. Modeli koji u obzir uzima interpunkcije te riječi u izvornom obliku se pokazao najboljim. Najlošiji model ne uzima interpunkcije niti velika slova u obzir.

CRF default		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	32	54
	Nisu likovi	52	5314

$$F_2 = 0.3738$$

CRF punct		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	33	53
	Nisu likovi	54	5312

$$F_2 = 0.3828$$

CRF lower		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	25	61
	Nisu likovi	47	5319

$$F_2 = 0.3005$$

CRF lower punct		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	27	59
	Nisu likovi	36	5330

$$F_2 = 0.3317$$

Posljednji na redu su Stanford NER modeli. S obzirom da je Stanford NER razvijan godinama od strane vrhunskih stručnjaka i znanstvenika, očekivali smo da će dati najbolje rezultate. Naša očekivanja su se ostvarila, što je vidljivo iz donjih tablica. Najbolji rezultati su ostvareni uz riječi u izvornom obliku i uklanjanje interpunkcijskih znakova iz teksta. Najlošiji rezultati su ostvareni u suprotnom slučaju, uz riječi pisane malim slovima i interpunkcijske znakove.

Stanford NER default		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	55	55
	Nisu likovi	31	5311

$$F_2 = 0.6057$$

NER punct		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	48	38
	Nisu likovi	30	5326

$$F_2 = 0.5556$$

NER lower		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	41	45
	Nisu likovi	66	5300

$$F_2 = 0.4545$$

NER lower punct		**Oznake modela**	
		Likovi	**Nisu likovi**
Točne oznake	**Likovi**	30	56
	Nisu likovi	36	5300

$$F_2 = 0.3659$$

V. Mogući budući nastavak istraživanja

Tijekom izrade ovog rada susreli smo se s problemom nedostupnosti podataka. Najsličniji i najčešći problem ove vrste u obzir uzima isključivo imenovane entiteta, odnosno imena osoba i nazive lokacija. Za taj problem postoje javno dostupni veliki skupovi označenih podataka, poput novinskih tekstova, ali takvi podaci nama nisu odgovarali. Stoga smo bili primorani ručno označavati podatke, što je relativno spor proces. Pretpostavljamo da su loši rezultati, pogotovo na skrivenim Markovljevim modelima, posljedica malog skupa podataka za učenje. Zato kao prvi korak za nastavak istraživanja preporučujemo povećanje skupa na treniranje na barem stotinu priča. Time bismo mogli povećati i veličinu skupa za testiranje, te dobiti vjerodostojnije podatke.

Kao sljedeći korak preporučujemo primjenu ansambla poput *bagginga*, odnosno *boostinga*, u ovisnosti o pristranosti, odnosno eventualnoj varijanci.

Kao posljednji korak spominjemo moguću upotrebu neuralnih mreža. Konvolucijske neuralne mreže su dobar odabir za probleme u analizi teksta, s obzirom da pri učenju u

obzir uzimaju okolinu (kontekst) svake riječi. Ipak, dosadašnja istraživanja pokazuju kako su metode poput skrivenih Markovljevih modela i uvjetnih nasumičnih polja jednako dobar ili bolji izbor. Također, veliki nedostatak neuralnih mreža je taj da zahtijevaju velik skup podataka za učenje. S obzirom da smo već naveli da je podatke potrebno ručno označavati, to bi moglo predstavljati veliki problem.

VI. Zaključak

Problem prepoznavanja imenovanih entiteta (NER) u tekstu se proučava već duže vrijeme, od sedamdesetih godina prošlog stoljeća. Primarno je cilj bio otkriti koje riječi predstavljaju ime osobe, naziv lokacije ili naziv organizacije u tekstu, najčešće novinskim člancima. Danas uspješna rješenja ovog problema donose financijsku korist. Naime, pretežno se koriste u oglašavanju, na primjer uz ime lokacije u članku nekog web portala možemo vezati oglas za hotel. Druga primjena je u označavanju i istraživanju biljnih, ljudskih i životinjskih gena i genoma.

Naš pristup je jako rijedak, pa ne postoje javno dostupni podaci za učenje u obliku anotiranih priča. Unatoč tome, uspjeli smo istrenirati relativno dobar model za prepoznavanje likova koristeći Stanford NER, izuzetno uspješan i godinama razvijan NER sustav. Očekujemo da bismo dobili bolje rezultate uz veći skup podataka za učenje.

Literatura

[1] M. Rudman, *Kompleksnost skrivenih Markovljevih modela*, (diplomski rad). Prirodoslovno-matematički fakultet, 2014.

[2] I. Medic, *Uslovna slučajna polja*, (master rad). Univerzitet u Beogradu, Matematički fakultet, 2013

[3] *Stanford Named Entity Recognizer*, The Stanford Natural Language Processing Group, http://nlp.stanford.edu/software/CRF-NER.shtml