Zbornik radova Fakulteta tehničkih nauka, Novi Sad

UDK: 004.85
DOI: https://doi.org/10.24867/07BE25Peric

DETEKCIJA SARKAZMA U KOMENTARIMA SA *REDDIT* STRANICE

SARCASM DETECTION IN *REDDIT* COMMENTS

Sara Perić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *Analiza sentimenta je veoma zastupljena u istraživanjima danas. Deo problema određivanja sentimenta predstavlja detekcija sarkazma, jer on često može da navede model da zaključuje suprotno od tačnog. Ovaj fenomen se javlja zbog same prirode sarkazma: upotreba pozitivnih reči u cilju izražavanja negativnih osećanja. Tema ovog rada je detekcija sarkazma u komentarima, gde se on češće javlja u odnosu na druge tekstualne sadržaje. U ovom radu prikazani su različiti pristupi u rešavanju datog problema. Predloženi su različiti klasifikacioni modeli – metod slučajnih šuma (engl. Random Forest), metod potpornih vektora (engl. Support Vector Machines - SVM), logistička regresija, kao i različite arhitekture neuronskih mreža – Yoon Kim model, konvolucioni model, rekurentni model i konvoluciono-rekurentni. Uporedo sa srodnim istraživanjima i ovim radom je pokazano da je detekcija sarkazma moguća i da se daljim unapređenjem modela tačnost može povećati i time doprineti značajnom poboljšanju analize sentimenta.*

Ključne reči: *Sarkazam, Konvolucione i Rekurentne neuronske mreže, Metod potpornih vektora, Logistička regresija, Metod slučajnih šuma*

Abstract – *Sentiment analysis is widespread in research today, and sarcasm represents one of its problems, which can often lead the model to conclude the opposite of the correct sentiment. This phenomenon occurs because of the very nature of sarcasm: the use of positive words to express negative feelings. Therefore, the topic of sarcasm detection was chosen particularly in comments, where it occurs more often than in usual textual content. This paper presents different approaches to solving a given problem. Different classification models are proposed - Random Forest, Support Vector Machines (SVM), Logistic Regression, as well as various neural network architectures - Yoon Kim model, convolution model, recurrent model, and convolution-recurrent model. Along with related research, this work has shown that sarcasm detection is possible and that by further refining of the model, accuracy can be increased and thus contribute to a significant improvement in sentiment analysis.*

Keywords: *Sarcasm, Convolutinal and Recurrent neural networks, Support Vector Machines, Logistic Regression, Random Forest*

1. UVOD

Sarkazam predstavlja stilsku figuru koja se koristi za izražavanje negativnih osećanja korišćenjem pozitivnih reči. Tumači se kao ruganje s ciljem da se ismeje onaj kome je upućena sarkastična poruka. Tokom dijaloga, ljudi se često koriste tehnikama naglašavanja reči, kao i različitim oblicima gestikulacije s ciljem naglašavanja ironije. S obzirom na to da se u tekstu ne nalazi takav vid „olakšice“ za prepoznavanje sarkazma, utoliko je teže detektovati ga, kako za ljude tako i za istrenirane modele mašinskog učenja. Performanse sentiment analize, koja predstavlja deo mnogih modernih istraživanja, mogu biti narušene zbog problema pogrešnog zaključivanja sentimenta usled prisustva sarkazma [1].

Istraživanjem rešenja koja se bave problemom detekcije sarkazma, došlo se do zaključka da je komentare pre svega potrebno pretprocesirati, a da se potom nad njima treniraju različiti modeli mašinskog učenja. Naučni radovi koji se bave sličnom temom su vršili detekciju sarkazma u postovima sa *Twitter*[1] društvene mreže [6, 7, 8, 10]. Za razliku od njih, ovaj rad koristi drugačiji skup podataka, sa drugog veb portala (*Reddit*[2]).

Dati skup podataka se sastoji od komentara varijabilne dužine (od po koje reči do nekoliko rečenica), tekst je manje formalan, te i ispravnost gramatike varira od komentara do komentara. Dok su tvitovi celina za sebe, komentari sa *Reddit* stranice mogu biti uslovljeni samim sadržajem koji se komentariše, tekstom roditeljskog komentara te i viših komentara u hijerarhiji vezanih za originalni post.

Ono što je u ovom radu slično prethodnim rešenjima jesu metode koje su bile primenjene za rešavanje datog problema, a to su: metod slučajne šume (*Random Forest*) [2], metod potpornih vektora (*Support Vector Machines*, SVM) [3], logistička regresija (*Logistic Regression*) [4], kao i *Yoon Kim* model [5], konvolucioni model i konvoluciono-rekurentni model.

U narednom poglavlju su prikazani srodni radovi koji se bave problemom detekcije sarkazma u tekstu, modeli koji one koriste i rezultati koji su postignuti.

U poglavlju 3 opisan je skup podataka, komentara sa *Reddit* stranice koji je korišćen u radu, te primenjene metodologije.

U poglavlju 4 prikazani su postignuti rezultati. Poglavlje 5 sadrži sumarizaciju rada.

NAPOMENA:
Ovaj rad proistekao je iz master rada čiji mentor je bila dr Jelena Slivka, docent.

[1] https://www.twitter.com
[2] https://www.reddit.com

2. SRODNA ISTRAŽIVANJA

Zadatak rada [6] jeste prepoznavanje sarkazma u postovima na *Twitter*-u, a rešenje je bazirano na prethodnom istraživanju [7]. Skup podataka je nastao tokom nekoliko meseci skupljanja tvitova 2014. godine. Sadrži 25.278 sarkastičnih i 117.842 nesarkastičnih postova. Sakupljeni postovi su nasumice podeljeni na dva skupa, 70% podataka za trening i 30% za test skup. Osnovni model korišten u radu [6] je SVM, na način na koji je implementiran u *LinearSVC*[3] funkciji iz biblioteke *scikitlearn*[4] u *Python*-u. Drugi pristup isproban u radu [6] je primena Naivnog Bajesovog klasifikatora[5], dok je treći bio takođe SVM, ali samo sa jednom klasom koju su činili nesarkastični postovi. Pre obuke modela mašinskog učenja je izvršeno pretprocesiranje podataka, pri čemu su izbačeni haštagovi, karakteri koji nisu ASCII i http linkovi. Za evaluaciju rešenja su korišteni F1 mera i tačnost (eng. *accuracy*).

Tema rada [8] je prepoznavanje sarkazma u tekstualnom sadržaju, na osnovu grupe ključnih reči izdvojenih iz *Twitter* postova. Skup podataka je generisan tako što su izdvojeni postovi koji imaju u sebi #*sarcasm* ili #*sarcastic* i oni su dati ljudima na prevođenje, s ciljem pronalaska adekvatne reči u bukvalnom značenju (za razliku od sarkazma prisutnog u inicijalnoj reči). Za ovo je korišćena *Amazon Mechanical Turk* platforma[6]. Na ovaj način je za svaki sarkastičan tvit dobijeno više nesarkastičnih prevoda, nakon čega je tehnikama nenadgledanog učenja detektovana ključna reč iz sarkastičnog tvita, traženjem reči sa suprotnim značenjem.

Nakon što je kreiran skup ključnih reči, skup sarkastičnih izjava je dobijen tako što su iz skupa sarkastičnih tvitova (*#sarcasam* ili *#sarcastic*) izdvojeni oni u kojima se ključne reči pojavljuju, a skup nesarkastičnih izjava izdvajanjem tvitova bez pomenutih tagova koji sadrže ključne reči. Izdvojen je i treći skup, koji predstavlja skup nesarkastičnih izjava sa sentimentom, tako što su traženi tvitovi sa tagovima poput *#sad*, *#happy* koji sadrže neku ključnu reč. Za 70 ključnih reči ukupno je prikupljeno 2.542.249 tvitova. Ukupno 80% je korišteno za trening, 10% za test i 10% za razvoj. Primenjena su dva pristupa, distribucioni i klasifikacioni. Distribucioni semantički model koristi kontekstne vektore za reprezentaciju podataka i kosinusnu sličnost kao meru udaljenosti.

Za klasifikaciju je korišten SVM sa modifikovanim kernelom i različitim *word embedding* modelima [9]. Tvitovi su pretprocesirani tako da su sve reči konvertovane u mala slova, izbačeni su hashtagovi, svi brojevi su konvertovani u generički token "22". Ono što se razlikuje od rada [6], je *word2vec* koji je usvojen i iskoristićen i u modelu prikazanom u ovom radu, kao deo pripreme podataka, što je kasnije objašnjeno u četvrtom poglavlju.

Zadatak rada [10] jeste prepoznavanje sarkazma u tekstualnom sadržaju tvitova. Metodologija za rešavanje problema se sastojala u korišćenju dubokih konvolucionih mreža sa ciljem uočavanja konteksta samog tekstualnog sadržaja. Upotrebljeno je više nezavisno istreniranih modela konvolucionih neuronskih mreža namenjenih za predikciju sentimenta (skup podataka je sadržao rečenice od kojih su 5895 pozitivnog sentimenta, 3131 negativnog i 471 neutralnog), emocije teksta (sa skupom od 5205 rečenica) i ličnosti (sa skupom podataka od 2400 eseja labeliranih tipovima ličnosti). Krajnje rešenje koristi dva modela: „čistu" konvolucionu neuronsku mrežu i obeležja izdvojena iz konvolucione mreže koja su potom prosleđena kao ulaz u SVM model. Prilikom pripreme podataka, za reprezentaciju reči korišten je *Google*-ov *word2vec* [9]. Ono što se može zaključiti iz ovog rada jeste da se za rešavanje problema detekcija sarkazma mogu koristiti konvolucione neuronske mreže, te su po uzoru na njega primenjene i u ovom radu. Za računanje tačnosti korištena je F1 mera, koja je iznosila 93,30%.

Po uzoru na rad [6], u ovom radu je primenjen SVM, sa različitim parametrima, o čemu je reč u poglavlju 3. Naivni Bajes nije davao veću tačnost od SVM-a, te nije korišćen u ovom radu. Obrada podataka iz rada [6] je uzeta kao osnovna ideja, koja je potom nadograđena. Ideja za uptrebu *word2vec* modela proistekla je iz rada [8] i [10]. Rad [10] je pokazao da se za rešavanje problema detekcije sarkazma mogu koristiti konvolucione neuronske mreže, te su po tom uzoru primenjene i u ovom radu.

3. METOD

U narednim poglavljima izloženi su skup podataka, arhitektura i trening modela.

3.1. Skup podataka

Korišćeni skup podataka je javno dostupan resurs sa *Kaggle*[7] sajta, koji se odnosi na komentare na *Reddit*-u[8]. Čini ga potpuno balansirani odnos broja sarkastičnih komentara prema nesarkastičnim, odnosno, 505.413 komentara svakog tipa. Ciljni atribut predstavlja labela kao indikator prisutnosti sarkazma u komentaru, dok ostale atribute predstavljaju: roditeljski komentar, tekst komentara, *subreddit* (označava kom podforumu pripada komentar), *ups* i *downs* (numerički atributi koji pokazuju koliko osoba je dalo pozitivnu, odnosno, negativnu ocenu posmatranom komentaru), s*core* (numerički atribut koji predstavlja ukupnu ocenu komentara), autor komentara, vreme i datum kreiranja komentara.

3.2. Obrada podataka

Inicijalna faza razvoja bazirana je na pretprocesiranju skupa podataka i obradi teksta, te obuhvata:

1. Izbacivanje nepoželjnih karaktera (linkova, tagova, ne alfa-numeričkih karaktera, zamena pojave karaktera @ sa "at"),
2. Prebacivanje teksta u mala slova
3. Tokenzaciju - pretvaranje komentara i roditeljskog komentara u dva vektora stringova; odvajanje skraćenica, većine znakova interpunkcije, zareza i pojedinačnih navodnika, kada ih prati razmak, kao i tačaka koje se nalaze na kraju
4. Izbacivanje stop reči - reči koje se često javljaju u tekstu, a ne doprinose značenju teksta (zamenice, predlozi i prilozi kao što su, za dati skup podataka na

[3] https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html
[4] https://scikit-learn.org/stable/index.html
[5] https://scikit-learn.org/stabsle/modules/naive_bayes.html
[6] https://www.mturk.com/
[7] https://www.kaggle.com/
[8] https://www.kaggle.com/danofer/sarcasm

engleskom jeziku „*the*", „*a*", „*an*", „*in*", „*between*", „*yourself*", „*but*", „*again*", itd.); stop reči su preuzete iz *nltk*[9] biblioteke, iz kategorije reči za engleski jezik.

3.3. Metodologija

Metodologija ovog rešenja može se podeliti u dve celine: obučavanje i upotreba različitih vektorskih reprezentacija teksta (u ovom slučaju: komentara i roditeljskog komentara) i korišćenje odgovarajuće klasifikacione metode u cilju pronalaženja one koja za dati problem daje najtačnije rezultate. Isprobane vektorske reprezentacije su *word2vec* [9] i *GloVe* [11].

Za potrebe klasifikacije je izvučeno 100.000 nasumičnih primeraka iz dobijenog modela, kako bi računska zahtevnost procesa klasifikacije bila manje kompleksna. Trening i test skupovi su dobijeni deljenjem originalnog skupa podataka, u razmeri 7:3 pri čemu je očuvana distribucija labela.

Word2vec model je istreniran nad celokupnim skupom podataka (komentari i roditeljski komentari). Postavljeni parametri istreniranog modela su sledeći:

- Dimenzionalnost vektora: 50
- Minimalni broj pojavljivanja reči u skupu: 1
- Maksimalna distanca između trenutne i prediktovane reči u rečenici: 10
- Vrednost praga iznad kog reči sa većom frekvancijom se nasumično smanjuju: $1e^{-3}$.

Ostali parametri imaju predefinisane vrednosti. Razlog ovakvih podešavanja se ogleda u optimalnim rezultatima do kojih se došlo empirijskom analizom. Model je prethodno treniran i sa drugačijim parametrima, od kojih je bitno istaknuti dimenzionalnost vektora, čija vrednost je prvobitno bila postavljena na 300, 150, 100. No, kako na taj način istrenirani model nije doprinosio tačnosti rešenja, odabrana je vrednost 50, što je uzrokovalo ubrzanjem treniranja i predikcije rešenja. Pored ručno treniranog modela, isproban je i pretrenirani model *Google News*, dimenzionalnosti 300 [12].

GloVe nudi nekoliko pretreniranih modela od kojih su isprobana dva trenirana nad *Twitter* sadržajima, dimenzionalnosti vektora 50 i 100 [11].

Sledi pregled klasifikacionih metoda.

3.3.1 Yoon Kim model – NN model 1

U ovom projektu korištena je modifikovana verzija *Yoon Kim* modela. Implementiran je korišćenjem *Keras* programske biblioteke[10]. Sastoji se od:

1. Ulaznog sloja (*Embedding*)
2. Konvolucionog sloja (veličina filtera: 128, veličina kernela: 3, aktivaciona funkcija: *Relu*)
3. *MaxPooling* sloja
4. Konvolucionog sloja (veličina filtera: 128, veličina kernela: 4, aktivaciona funkcija: *Relu*)
5. *MaxPooling* sloja
6. *Droput* sloja
7. Potpuno povezanog sloja (veličina: 128)
8. *Dropout* sloja
9. Potpuno povezanog sloja (aktivaciona funkcija: *softmax*, optiomizaciona funkcija: Adam)

Korišćeni atributi: labela, komentar, roditeljski komentar. Kako upotreba ostalih atributa nije uticala na povećanje tačnosti, pri daljem treniranju modela nisu korišteni. Isto važi i za ostale NN (*Neural Network*) modele.

3.3.2 Konvolucioni model – NN model 2

Kreiran po uzoru na metod koji je korišćen u radu [10]. Sastoji se od:

1. Ulaznog sloja (*Embedding*)
2. Konvolucionog sloja (veličina filtera: 128, veličina kernela: 4, aktivaciona funkcija: *Relu*)
3. *MaxPooling* sloja
4. Konvolucionog sloja (veličina filtera: 128, veličina kernela: 3, aktivaciona funkcija: *Relu*)
5. *MaxPooling* sloja
6. Potpuno povezanog sloja (veličina: 128)
7. Potpuno povezanog sloja (veličina: 2, aktivaciona funkcija: *softmax*, optimizaciona funkcija: *Adadelta*)

3.3.3 Konvoluciono-rekurentni model – NN model 3

Sastoji se od:

1. Ulaznog sloja (*Embedding*)
2. Konvolucionog sloja (veličina filtera: 64, veličina kernela: 5, aktivaciona funkcija: *Relu*)
3. *MaxPooling* sloja
4. LSTM (*Long shot-term memory*) sloja (veličina: 100)
5. Potpuno povezanog sloja (aktivaciona funkcija: *softmax*, optimizaciona funkcija: *Adagrad*

3.3.4 Rekurentni model – NN model 4

Isporban metod biderekcionih LSTM ćelija:

1. Ulazni sloj (*Embedding*)
2. Bidirekcioni LSTM sloja (veličine:100)
3. Potpuno povezanog sloja (aktivaciona funkcija: *softmax*, optimizaciona funkcija: rmsprop).

3.3.5 Ostale klasifikacione metode

Pored navedenih modela neuronskih mreža, isprobane su i druge tehnike mašinskog učenja po ugledu na [6, 8, 10]. Korišteni atributi: labela, komentar, roditeljski komentar, *score*, *ups*, *downs*. Do vrednosti navedenih parametara u zagradama došlo se empirijski, korišćenjem validacionog skupa, dok ostali parametri zadržavaju podrazumevane vrednosti.

1. *Random Forest* klasifikator (*n_estimators*: 1000, *min_samples_split*: 16)
2. SVM klasifikator (LinearSvc, c: 0.01)
3. Logistička regresija (*multiclass*: "ovr", *solver*: "liblinear").

4. REZULTATI I DISKUSIJA

Treniranje modela neuronskih mreža je zaustavljeno *Keras* mehanizmom za rano zaustavljanje (engl. *EarlyStopping*) posmatranjem funkcije gubitka (engl. *loss function*), pri promeni manjoj od 0,001 i faktorom strpljenja 6.

Najbolje se pokazao NN model 1, korišćenjem *word2vec* vektorske reprezentacije veličine 50, treniranog nad celokupnim skupom podataka. Tokom 12 epoha se trenirala

[9] https://www.nltk.org/
[10] https://keras.io/

mreža dostignuvši tačnost od 66% nad test skupom podataka, a 72% nad trening skupom podataka.

Tokom 9 epoha trenirala se mreža modela NN 2 dostignuvši tačnost od 63% nad test skupom podataka, a 80% nad trening skupom podataka, koršćenjem *word2vec* vektorske reprezentacije veličine 50, trenirane nad celokupnim skupom podataka.

Najvišu tačnost model NN 3 je dostigao koršćenjem *GloVe* vektorske reprezentacije, veličine 100, prethodno istreniran nad *Twitter* sadržajima. Mreža je trenirana tokom 9 epoha, dostignuvši tačnost od 63% nad test skupom podataka, a 76% nad trening skupom podataka.

NN Model 4 se najbolje pokazao koršćenjem *GloVe* vektorske reprezentacije veličine 100, prethodno istreniran nad *Twitter* sadržajima. Mreža je trenirana tokom 15 epoha, dostignuvši tačnost od 64% nad test skupom podataka, a 78% nad trening skupom podataka.

Random Forest se najbolje pokazao ne koristivši se atributom roditeljskog komentara sa tačnošću 65,31%. SVM daje gore rezulate: 62,02%, dok mu je vrlo bliska Logistička regresija sa dostignutih 62,03%.

Tačnost koja je postignuta u ovom radu je u opsegu 60% do 80%, zavisno od modela koji je upotrebljen, nasuport najvišoj tačnosti od 97% u srodnim radovima.

Mimoilaženja u rezultatima ogledaju se delom u veličini skupa podataka, a delom i u samom kvalitetu podataka. Komentari su, poput postova, nestruktuirani tip podataka, ali ono što dodatno otežava detekciju sarkazma u njima jeste što se, za dati skup podataka, sarkazam može odnositi na sadržaj samog komentara, na roditeljski komentar ili čak komentar koji je izvan domena postojećeg skupa podataka. Istrenirani modeli korišteni u radu [10] dosta su kompleksniji od modela do kojih se došlo u ovom radu. Pored toga, sam skup podataka nad kojim je obučena mreža dosta se razlikuje, s obzirom da je pored tvitova korišten veliki broj eseja, dubokog teksta, što samom modelu „olakšava“ zaključivanje konteksta.

Kao jednu od mogućnosti unapređenja, potrebno je naglasiti važnost pronalaženja novih specifičnih obeležja sarkastičnog teksta, pomoću kojih bi detektovanje postalo preciznije. Potrebno je i naći veće i pouzdanije skupove podataka, koji zasigurno sadrže sarkastične tekstove, za razliku od oslanjanja na *#sarcasm*, *#sarcastic* i slična labeliranja od strane samih korisnika društvenih mreža.

5. ZAKLJUČAK

U ovom radu predstavljeno je nekoliko metoda mašinskog učenja koje bi se mogle primeniti za rešavanje problema detekcije sarkazma kao i kako se svaki od njih ponaša na zadatom skupu komentara sa *Reddit* veb stranice. Metode kombinuju ideje iz prethodnih, srodnih istraživanja: SVM [6, 8], konvolutivne mreže [10] kao inicijativno isprobane: Logistička regresija, *Random Forest* i rekurentne neuronske mreže.

Rezultati pokazuju da je detekcija sarkazma u tekstu moguća, ali da je ipak potrebno refiniranje modela kako bi njihova primena u kombinaciji sa modelima za analizu sentimenta doprinela sveopštem cilju. Predloženi su mogući dalji koraci razvoja u ovom smeru, koji bi omogućili povećanje tačnosti modela za detekciju sarkazma.

6. LITERATURA

[1] Sarcastic sentiment detection in tweets streamed in real time: a big data approach, K.Bharti, B.Vachha, R.K.Pradhan, K.S.Babu, S.K.Jena

[2] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[3] https://scikit-learn.org/stable/modules/svm.html

[4] https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

[5] Convolutional neural networks for sentence, Kim, Yoon, 2014

[6] Detecting Sarcasm in Text, Peng, Chun-Che; Lakis, Mohammad, Pan, Jan Wei.

[7] http://www.thesarcasmdetector.com/

[8] Sarcastic or not: Word embeddings to predict the literal or sarcastic meaning of words, Ghosh, Debanjan; Guo, Weiwei; Muresan, Smaranda.

[9] https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa

[10] A deeper look into sarcastic tweets using deep convolutional neural networks, Poria, Soujanya

[11] https://nlp.stanford.edu/projects/glove/

[12] https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/edit

Kratka biografija:

Sara Perić rođena je 1996. godine u Loznici. Osnovne akademske studije završila je 2018. godine na Fakultetu tehničkih nauka, na kom brani i master rad 2019. godine iz oblasti Elektrotehnike i računarstva – Softversko inženjerstvo i informacione tehnologije.
kontakt: sara.peric013@gmail.com