

Primjena i vrednovanje metoda za detekciju sarkazma u engleskome jeziku

Pavlić, Josip

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:168:220942>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-03-14**



Repository / Repozitorij:

[FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1532

**PRIMJENA I VREDNOVANJE METODA ZA DETEKCIJU
SARKAZMA U ENGLESKOME JEZIKU**

Josip Pavlić

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1532

**PRIMJENA I VREDNOVANJE METODA ZA DETEKCIJU
SARKAZMA U ENGLESKOME JEZIKU**

Josip Pavlić

Zagreb, lipanj 2024.

ZAVRŠNI ZADATAK br. 1532

Pristupnik: **Josip Pavlić (0036543803)**
Studij: Elektrotehnika i informacijska tehnologija i Računarstvo
Modul: Računarstvo
Mentor: prof. dr. sc. Jan Šnajder

Zadatak: **Primjena i vrednovanje metoda za detekciju sarkazma u engleskome jeziku**

Opis zadatka:

Sarkazam je oblik verbalne ironije koji je karakteriziran korištenjem jezika čije je doslovno značenje suprotno namjeravanom značenju kod tumačenja teksta. Ova specifičnost sarkazma oslanja se na značajke poput intonacije, konteksta i pozadinskog znanja kako bi se ispravno prenijelo namjeravano značenje, čineći ga izazovnim problemom u domeni obrade prirodnoga jezika. Tema ovog završnog rada je primjena i vrednovanje različitih metoda dubokog učenja, s naglaskom na modele arhitekture transformatora, na problem detecije sarkazma u engleskome jeziku. Cilj rada je implementirati i usporediti različite pristupe za binarnu detekciju te klasifikaciju u podkategorije sarkazma koristeći označeni skup podataka objavljen u dijeljenom zadatku iSarcasmEval (SemEval 2022). Potrebno je proučiti kako predtrenirani model radi bez finog ugađanja parametara na podacima i nakon njega. Usporediti rezultate i komentirati razlike, a potom koristiti odgovarajuće metrike za vrednovanje točnosti modela, provesti optimizaciju hiperparametara i odgovarajuću statističku analizu rezultata. Radu je potrebno priložiti izvorni kod, rezultate i programsku dokumentaciju te citirati korištenu literaturu.

Rok za predaju rada: 14. lipnja 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1532

**PRIMJENA I VREDNOVANJE METODA ZA
DETEKCIJU SARKAZMA U ENGLESKOME
JEZIKU**

Josip Pavlič

Zagreb, lipanj 2024.

ZAVRŠNI ZADATAK br. 1532

Pristupnik: **Josip Pavlić (0036543803)**
Studij: Elektrotehnika i informacijska tehnologija i Računarstvo
Modul: Računarstvo
Mentor: prof. dr. sc. Jan Šnajder

Zadatak: **Primjena i vrednovanje metoda za detekciju sarkazma u engleskome jeziku**

Opis zadatka:

Sarkazam je oblik verbalne ironije koji je karakteriziran korištenjem jezika čije je doslovno značenje suprotno namjeravanom značenju kod tumačenja teksta. Ova specifičnost sarkazma oslanja se na značajke poput intonacije, konteksta i pozadinskog znanja kako bi se ispravno prenijelo namjeravano značenje, čineći ga izazovnim problemom u domeni obrade prirodnoga jezika. Tema ovog završnog rada je primjena i vrednovanje različitih metoda dubokog učenja, s naglaskom na modele arhitekture transformatora, na problem detecije sarkazma u engleskome jeziku. Cilj rada je implementirati i usporediti različite pristupe za binarnu detekciju te klasifikaciju u podkategorije sarkazma koristeći označeni skup podataka objavljen u dijeljenom zadatku iSarcasmEval (SemEval 2022). Potrebno je proučiti kako predtrenirani model radi bez finog ugađanja parametara na podacima i nakon njega. Usporediti rezultate i komentirati razlike, a potom koristiti odgovarajuće metrike za vrednovanje točnosti modela, provesti optimizaciju hiperparametara i odgovarajuću statističku analizu rezultata. Radu je potrebno priložiti izvorni kod, rezultate i programsku dokumentaciju te citirati korištenu literaturu.

Rok za predaju rada: 14. lipnja 2024.

Ana Barić, David Dukić, Jan Šnajder, Gašpar Haramija, Dorian Matić

Sadržaj

1. Uvod	3
2. Srodni radovi	5
2.1. Pristupi s plitkim modelima	5
2.2. Pristupi s dubokim modelima	6
2.3. Pristupi s višezadaćnim učenjem	7
3. Sarkazam u obradi prirodnog jezika	9
4. Podaci	11
5. Modeli	15
5.1. Stroj potpornih vektora	15
5.2. Transformatorski modeli	18
5.2.1. Koder	19
5.2.2. Dekoder	20
5.3. BERT	21
5.4. Fino ugađanje transformatorskih arhitektura	22
5.5. Višezadaćno učenje	22
6. Metodologija	24
6.1. Preobrada podataka	25
6.2. Stroj potpornih vektora	26
6.3. BERT	27
6.3.1. Modeli višezadaćnog učenja	28
6.4. Obrada izlaza modela BERT	29

7. Rezultati	31
7.1. Vrednovanje modela	31
7.2. Rezultati referentnih modela	33
7.3. Rezultati testiranja	36
7.3.1. Zadatak 1	37
7.3.2. Zadatak 2	39
7.3.3. Zadatak 3	40
7.3.4. Analiza	42
8. Zaključak	44
Literatura	45
Sažetak	49
Abstract	50

1. Uvod

U svakodnevnom životu evaluativni jezik igra ključnu ulogu u izražavanju stavova i emocija. Sarkazam je specifičan oblik ironije koji se pojavljuje kada postoji nepodudarnost između doslovnog i namjeravanog značenja izraza. Kroz ovu nepodudarnost, govornik izražava svoj stav prema prethodnoj tvrdnji, često u obliku prijezira ili omalovažavanja. Razvoj modela za detekciju sarkazma od ključne je važnosti kako bi se moglo bolje razumjeti izrečeno, posebno u kontekstu analize sentimenta. Automatska detekcija sarkazma može značajno poboljšati performance sustava za analizu sentimenta, detekciju lažnih vijesti i komunikacijske aplikacije koje koriste chatbotove. U ovom kontekstu obrada prirodnog jezika (engl. *natural language processing*, NLP) ima ključnu ulogu jer omogućuje modelima strojnog učenja da analiziraju, razumiju i generiraju ljudski jezik na način koji je koristan za ove zadatke.

Cilj ovog završnog rada bio je usporediti različite metode strojnog učenja (engl. *machine learning*) u svrhu detekcije različitih granulacija sarkazma u rečenicama engleskog jezika. U radu su obrađena tri različita zadatka koja uključuju (1) binarnu detekciju sarkazma, (2) višerazrednu detekciju potkategorija sarkazma te (3) prepoznavanje sarkastičnih od nesarkastičnih inačica rečenica. Metodologija je obuhvatila usporedbu stroja potpornih vektora (engl. *support vector machine*, SVM), transformatorske arhitekture (engl. *Transformer*) te naprednijih metoda treniranja transformator modela pomoću višezadačnog učenja (engl. *multi-task learning*). Sva tri zadatka preuzeta su s natjecanja SemEval iz 2022. godine [1]. Za svaki od podzadataka kao početni model koristio se stroj potpornih vektora, a potom su se primijenili transformatori zbog njihovih sposobnosti raazumijevanja konteksta i značenja teksta te jednostavne adaptacije modela na specifične zadatke pomoću postupka finog ugađanja (engl. *fine tuning*). Modeli višezadačnog učenja koristili su se kako bi se potencijalno poboljšali originalni transformatorski mo-

deli prenoseći znanje iz jednog u drugi zadatak i obratno.

U ovom radu prvo će se prikazati pregled srodne literature za detekciju sentimenta i opisati izazove te ograničenja istih u poglavlju 2. Zatim će se formalno definirati sarkazam i njegove potkategorije u poglavlju 3. Podaci korišteni u istraživanju bit će opisani u poglavlju 4., dok će se metodologija i modeli detaljno opisati u poglavljima 5. i 6. Rezultati testiranja modela i konačna analiza bit će prikazani u poglavlju 7., a zaključci rada u poglavlju 8.

2. Srodni radovi

S obzirom na brojne materijale vezane za srodne radove, ovo poglavlje je izdvojeno kako bi se prikazali različiti pristupi korišteni za detekciju sarkazma.

2.1. Pristupi s plitkim modelima

Pristupi koji koriste plitke modele poput stroja potpornih vektora često se oslanjaju na frekvenciju riječi te ne uzimaju u obzir kontekst. Ovakvi pristupi teško prepoznaju skriveno značenje u rečenicama.

- **Korištenje TF-IDF značajki s klasifikatorima nasumične šume i stroja potpornih vektora te implementacija SMOTE tehnike za balansiranje podataka [2]:** Ovaj pristup koristi metodu TF-IDF (engl. *Term Frequency-Inverse Document Frequency*) za kvantificiranje važnosti riječi u dokumentu. TF-IDF je metoda za ekstrakciju značajki koja vektorizira tekst u numeričke vrijednosti koje odražavaju važnost riječi u dokumentu tako što riječi koje se često pojavljuju u jednom dokumentu, ali ne i u cijelom korpusu, dobivaju višu težinu. Nakon toga, primjenjuju se klasifikatori nasumične šume i stroja potpornih vektora za klasifikaciju teksta. Detaljan opis stroja potpornih vektora nalazi se u (v. potpoglavljje 5.1.). Uz to, koristi se tehnika SMOTE (engl. *Synthetic Minority Over-sampling Technique*) zbog neravnoteže skupa podataka. Neravnoteža skupa podataka predstavlja problem jer klasifikatori imaju tendenciju favoriziranja većinskih klasa, što može rezultirati lošijom izvedbom za manjinske klase. SMOTE tehnika rješava ovaj problem generiranjem novih, sintetičkih primjera za manjinske klase, povećavajući tako njihov broj i pomažući klasifikatorima da bolje nauče značajke tih klasa [3]. U ovome radu obrađen je zadatak binarne detekcije sarkazma u engleskom i arapskom te zadatak višerazredne detekcije potkategorija sarkazma u engleskom je-

ziku. Za zadatak binarne detekcije sarkazma u engleskom jeziku, postignuta je vrijednost $F_1 = 0.4966$. Za zadatak višerazredne detekcije potkategorija sarkazma, postignuta je vrijednost $\text{macro-}F_1 = 0.0567$.

- **Detekcija sarkazma koristeći lagane modele [4]:** Ovaj rad istražuje korisnost laganih modela, kao što su model logističke regresije, Gaussov naivni Bayesov model, SVM, Ada Boost klasifikator i XGB klasifikator, u svrhu detekcije sarkazma. Model s najboljim performansama je XGB klasifikator (engl. *Extreme Gradient Boosting Classifier*) treniran na ručno izrađenim značajkama. U ovome radu obrađen je zadatak binarne detekcije sarkazma u engleskom jeziku. Za taj zadatak ovaj model postigao je vrijednost $F_1 = 0.2403$.

2.2. pristupi s dubokim modelima

Duboki modeli poput transformatora omogućuju bolje razumijevanje konteksta i skrivenih značenja u tekstu. Primjeri takvih radova su:

- **Korištenje modela BERT kao ekstraktora značajki u kombinaciji sa strojem potpornih vektora (SVM) i klasifikacijskom glavom iz biblioteke Wolf [5]:** Korištenje modela BERT [6] za ekstrakciju značajki uključuje inicijalizaciju modela BERT i tokenizatora, a zatim ekstrakciju značajki iz tokena $[CLS]$. Tako dobivene značajke koriste se kao ulaz u stroj potpornih vektora za klasifikaciju s kontekstualiziranim reprezentacijama. Kontekstualizirane reprezentacije odnose se na vektorske prikaze riječi unutar rečenica koji uzimaju u obzir i okolne riječi i njihov međusobni utjecaj na značenje. Detaljan opis modela BERT nalazi se u (v. potpoglavlje 5.3.). U ovome radu obrađen je zadatak binarne detekcije sarkazma u arapskom jeziku, zadatak višerazredne detekcije potkategorija sarkazma u engleskom i zadatak prepoznavanja sarkastičnih od nesarkastičnih inačica rečenica u engleskom jeziku. Za zadatak višerazredne detekcije potkategorija sarkazma, postignuta je vrijednost $\text{macro-}F_1 = 0.0431$. Za zadatak prepoznavanja sarkastičnih od nesarkastičnih inačica rečenica, postignuta je vrijednost $\text{Accuracy} = 0.765$.
- **Korištenje modela ELMo s jednim ugradbenim slojem (engl. *embedding*) i dva gusta sloja [7]:** model ELMo koristi duboke kontekstualne reprezentacije

riječi za precizno hvatanje sintakse i semantike, omogućujući modelu da bolje razumije sarkastične izraze u različitim kontekstima. Ovaj pristup je implementiran kroz CNN klasifikator konvolucijske neuralne mreže (engl. *convolutional neural network*, CNN) koji koristi ELMO ugradnje za rješavanje zadataka prepoznavanja sarkazma. CNN klasifikator koristi konvolucijske slojeve za ekstrakciju značajki iz podataka, dok ugradnje (engl. *embeddings*) predstavljaju numeričke prikaze riječi koji zadržavaju semantičke odnose između riječi te omogućuju modelu da prepozna i razumije složene jezične obrasce i kontekstualne informacije. U ovome radu obrađen je zadatak binarne detekcije sarkazma, zadatak višerazredne detekcije potkategorija sarkazma te zadatak prepoznavanja sarkastičnih od nesarkastičnih inačica rečenica. Sva tri zadatka obrađena su na engleskom jeziku. Za zadatak binarne detekcije sarkazma u engleskom jeziku postignuta je vrijednost $F_1 = 0.5101$. Za zadatak višerazredne detekcije potkategorija sarkazma, postignuta je vrijednost $\text{macro-}F_1 = 0.0387$. Za zadatak prepoznavanja sarkastičnih od nesarkastičnih inačica rečenica, postignuta je vrijednost $F_1 = 0.2794$.

- **Korištenje modela BERT, ColBERTa i RoBERTa [8]:** Modeli BERT, te ColBERT i RoBERTa, koji su varijante modela BERT, koriste napredne tehnike transformatora za klasifikaciju sarkastičnih i nesarkastičnih tekstova u engleskom i arapskom jeziku. Model BERT koristi mehanizam samopozornosti za kodiranje konteksta u rečenici, dok ColBERT optimizira kasne interakcije za bolju učinkovitost, a RoBERTa poboljšava preciznost i predikcije povećanom količinom treninga. Ovi modeli su trenirani na podacima koje su osigurali organizatori natjecanja. U ovome radu obrađen je zadatak binarne detekcije sarkazma u engleskom i arapskom i zadatak višerazredne detekcije potkategorija sarkazma u engleskom. Za zadatak binarne detekcije sarkazma u engleskom jeziku, postignuta je vrijednost $F_1 = 0.2637$. Za zadatak višerazredne detekcije potkategorija sarkazma, postignuta je vrijednost $\text{macro-}F_1 = 0.0596$.

2.3. Pristupi s višezadaćnim učenjem

Pristupi s višezadaćnim učenjem omogućuju modelima da dijele znanje između različitih zadataka, što može poboljšati performance na individualnim zadacima. Osnovna

razlika modela višezadaćnog učenja od normalnih modela je ta što modele višezadaćnog učenja treniramo na više zadataka odjednom te razvijamo njihovo zajedničko znanje, dok obični modeli svoje znanje razvijaju iz samo jednog specifičnog zadatka te su time limitirani. Detaljan opis modela baziranih na višezadaćnom učenju nalazi se u (v. poglavlje 5.5.)

- **Višezadaćno učenje za detekciju sarkazma na engleskom i arapskom jeziku [9]:** U ovome radu opisuje se pristup višezadaćnom učenju utemeljen na višejezičnom učenju. U njemu se koriste unaprijed trenirani jezični modeli poput ERNIE-M i DeBERT-a za prepoznavanje sarkazma na engleskom i arapskom jeziku. Višejezično učenje omogućuje modelu dijeljenje znanja između različitih jezika, poboljšavajući performance tako što trenira model na 2 zadatka odjednom prenoseći znanje iz arapske inačice u englesku inačicu zadatka i obratno. U ovome radu obrađen je zadatak binarne detekcije sarkazma na engleskom i arapskom, zadatak višerazredne detekcije potkategorija sarkazma na engleskom te zadatak prepoznavanja sarkastičnih od nesarkastičnih inačica rečenica na engleskom i arapskom. Za zadatak binarne detekcije sarkazma u engleskom jeziku, postignuta je vrijednost $F_1 = 0.5691$. Za zadatak višerazredne detekcije potkategorija sarkazma, postignuta je vrijednost $\text{macro-}F_1 = 0.0799$. Za zadatak prepoznavanja sarkastičnih od nesarkastičnih inačica rečenica na engleskom jeziku, postignuta je vrijednost $\text{Accuracy} = 0.8700$.

Usprkos brojnim pristupima koji se bave rješavanjem problema detekcije sarkazma, ovaj zadatak zajedno s njegovim inačicama predstavlja i dalje težak problem za područje obrade prirodnog jezika. Dodatno, u svakome od opisanih pristupa kompleksnost pojedinih aspekata problematike detekcije sarkazma je ograničeno pokrivena te time negativno utječe na uspješnost rješavanja ovog zadatka. U ovome radu fokus se stavlja na usporedbu performanci modela različite kompleksnosti kako bi se pronašli najkvalitetniji modeli za pojedine podzadatke.

3. Sarkazam u obradi prirodnog jezika

Sarkazam je oblik izražavanja u kojem govornik koristi riječi i izraze u kontekstu suprotne njihovom stvarnom značenju kako bi prenio kritiku, ismijavanje ili ironiju [10].

U svakodnevnoj komunikaciji, sarkazam se najčešće raspoznaje pomoću kombinacije značajki poput intonacije govornika, konteksta teme i pozadinskog znanja slušatelja. Međutim, u domeni obrade prirodnog jezika, značajke poput intonacije govornika su rijetko kad dostupne, dok su za reprezentaciju konteksta i pozadinskog znanja potrebni složeni modeli što čini zadatak detekcije sarkazma posebno složenim. Na primjer, rečenica „Da, naravno da te volim!“ može se interpretirati na sarkastičan i nesarkastičan način, te nekad sam tekst u tako kratkom formatu nije dovoljan indikator o tome može li se ta izjava smatrati sarkastičnom ili nesarkastičnom. To pokazuje da često u pismenom formatu imamo stanoviti „šum u govornom kanalu“, koji udaljava od samog značenja onoga što je druga osoba željela reći.

Kategorije ironičnog izražavanja, kao što su sarkazam, ironija, satira, podcjenjivanje, precjenjivanje i retoričko pitanje, dodatno kompliciraju zadatak prepoznavanja sarkazma jer svaka od njih zahtijeva različite pristupe u analizi teksta. Ironija proturječi stvarnom stanju, te može, ali i ne mora biti usmjerena prema entitetu, i ako je, nije očito kritična prema tom entitetu. Satira je usmjerena prema entitetu kojeg naizgled podržava, ali u stvari izražava neslaganje, ismijavanje, prijezir ili omalovažavanje istog. Podcjenjivanje ne proturječi stvarnom stanju, ali umanjuje njegovu važnost. Precjenjivanje također ne proturječi stvarnom stanju, ali mu pridaje nerealno visoku važnost. Retoričko pitanje je pitanje s impliciranim odgovorom koji proturječi stvarnom stanju [5].

S obzirom na tu činjenicu, stvara se i potreba za automatskim prepoznavanjem sarkazma u području obrade prirodnog jezika. Dok su standardni algoritmi za analizu senti-

menta i detekciju namjere u velikoj mjeri uspješni u prepoznavanju doslovnih značenja, sarkastične izjave često ostaju neotkrivene ili pogrešno interpretirane. To može rezultirati znatnim pogreškama u analizi stavova korisnika, što je posebno kritično za poslovne subjekte koji se oslanjaju na analitiku društvenih medija za donošenje poslovnih odluka.

Razumijevanje sarkazma u tekstu može unaprijediti performance sustava za analizu sentimenta, detekciju lažnih vijesti i poboljšati korisničko iskustvo u komunikacijskim aplikacijama koje primjenjuju chatbotove.

4. Podaci

Za eksperimente u ovom radu korišten je skup podataka iSarcasmEval preuzet sa SemEval natjecanja [1]. S obzirom na to da se rješavaju tri zadatka, svaki od zadataka koristi vlastiti testni skup podataka. U prvom zadatku radi se binarna klasifikacija sarkastičnih rečenica te se za njegovo testiranje koristi testni skup podataka `task_A_En_test`, koji će se u nastavku zvati testni skup binarne detekcije sarkazma. Zadatak binarne klasifikacije sarkazma će se u nastavku teksta referencirati kao zadatak 1. U drugom zadatku radi se višerazredna binarna klasifikacija potkategorija sarkazma te se za njegovo testiranje koristi testni skup podataka `task_B_En_test`, koji će se u nastavku zvati testni skup potkategorija sarkazma. Zadatak višerazredne detekcije potkategorija sarkazma će se u nastavku teksta referencirati kao zadatak 2. U trećem zadatku radi se prepoznavanje sarkastične inačice rečenice u odnosu na njemu refraziranu varijantu binarnom oznakom koja je od njih dvije sarkastična te se za njegovo testiranje koristi testni skup podataka `task_C_En_test`, koji će se u nastavku zvati testni skup refrazirane inačice. Zadatak prepoznavanja sarkastične inačice će se u nastavku teksta referencirati kao zadatak 3.

Skup za trening sastoji se 3468 primjera i 9 oznaka. Svaki *tweet* sadrži oznaku je li sarkastičan ili nesarkastičan, koja se koristi pri klasifikaciji u zadatku 1. Također, svaka sarkastična rečenica sadrži svoju refraziranu inačicu pod oznakom "rephrase", koja se koristi pri klasifikaciji u zadatku 3. Oznake smo dobili od samih autora tih rečenica. Također, ako je neka rečenica sarkastična to znači da se u barem jednom od 6 kategorija "sarcasm", "irony", "satire", "understatement", "overstatement" i "rhetorical_question" nalazi pozitivna oznaka, time označavajući pripadnost rečenice toj kategoriji ironičnog govora. Pozitivna oznaka se može nalaziti u proizvoljnom broju tih 6 kategorija, odnosno jedan *tweet* može istovremeno pripadati u više od jedne kategorije ironičnog govora. Refrazirane rečenice napisali su također autori originalnih rečenica, dok su lingvisti odabrali

oznake za šest vrsta ironičnog govora.

Ako je neka rečenica nesarkastična, u 6 kategorija ironične rečenice sadrži negativnu oznaku, s obzirom na to da očitno ne pripada nijednoj od njih, dok u kategoriji "rephrase" ne sadrži nikakvu oznaku jer ne možemo napraviti nesarkastičnu inačicu već nesarkastične rečenice.

Prvi zadatak svodi se na klasičnu binarnu klasifikaciju rečenice u kojem za zadanu rečenicu model mora predvidjeti je li ta rečenica sarkastična ili nije. Testni skup za ovaj zadatak je testni skup binarne detekcije sarkazma.

	Skup za učenje	Skup za testiranje
<i>sarcastic</i>	867	200
<i>non-sarcastic</i>	2601	1200
Ukupno	3468	1400

Tablica 4.1. Podjela distribucije primjera za binarnu detekciju sarkazma u zadatku 1. na skup za učenje i testiranje.

Primjeri sarkastičnog i nesarkastičnog *twita* nalaze se u tablici 4.2.

Primjer	Kategorija
The only thing I got from college is a caffeine addiction	<i>sarcastic</i>
Ready to not be pregnant so I can enjoy my first espresso martini	<i>non-sarcastic</i>

Tablica 4.2. Primjeri rečenica i njihove sarkastične oznake za binarnu detekciju sarkazma

U drugome zadatku trebali smo za neku rečenicu odrediti u koje od kategorija ironičnog govora spada, ako u ijednu od njih. Ovaj zadatak je višeznačna klasifikacija s obzirom na to da jedna rečenica može spadati u bilo koji broj oznaka. Testni skup za ovaj zadatak je testni skup potkategorija sarkazma.

	Sarkazam	Ironija	Satira	Podcjenj- ivanje	Precjenj- ivanje	Retoričko pitanje
Sarkastičan	713	155	25	10	40	101
Nesarkastičan	2932	3490	3620	3635	3605	3544
Ukupno	3645	3645	3645	3645	3645	3645

Tablica 4.3. Distribucija oznaka na skupu za trening potkategorija sarkazma za 2. zadatak

	Sarkazam	Ironija	Satira	Podcjenj- ivanje	Precjenj- ivanje	Retoričko pitanje
Sarkastičan	180	20	49	1	10	11
Nesarkastičan	1220	1380	1351	1399	1390	1389
Ukupno	1400	1400	1400	1400	1400	1400

Tablica 4.4. Distribucija oznaka na testnom skupu za potkategorije sarkazma u 2. zadatku

Primjeri različitih potkategorija *tvitova* nalaze se u tablici 4.5.

Primjer	Potkategorija sarkazma
Today my pop-pop told me I was not 'forced' to go to college... okay sure sureeee.	<i>Sarkazam</i>
Good morning September... Hello tonsillitis :(<i>Ironija</i>
Well, at least the covid situation seems to be improving in time for christmas...	<i>Sarkazam i Satira</i>
Bit odd for someone to resign over a party that didn't happen *eye roll*	<i>Sarkazam, Pocjenjivanje</i>
Having the worst time on holiday	<i>Sarkazam, Precjenjivanje</i>
When someone puts on too much perfume I would say nice perfume, how long did you let it marinate???	<i>Sarkazam, Retoričko Pitanje</i>

Tablica 4.5. Primjeri rečenica i njihove oznake za zadatak 2.

U trećem zadatku za par rečenica, od kojih je jedna sarkastična a druga njezina ne-sarkastična inačica, trebamo odrediti koja od njih je sarkastična, odnosno na kojoj od te dvije pozicije se nalazi sarkastična rečenica. Radi se o binarnoj klasifikaciji s obzirom na to da su samo dvije rečenice u pitanju. Testni skup za ovaj zadatak je Testni skup C.

	Trening skup	Testni skup
Pozicija 0	434	107
Pozicija 1	433	93
Ukupno	867	200

Tablica 4.6. Prikaz pozicija sarkastičnih rečenica na trening i testnom skupu za zadatak prepoznavanja sarkastične inačice

Primjeri različitih *tweetova* i njihovih refraziranih varijanti nalaze se u tablici 4.7.

Nulta rečenica	Prva rečenica	Pozicija sarkazma
I see that your team played well today!	I'm sorry that your team didn't win yesterday.	<i>Nulta rečenica</i>
The weather is gloomy, just raining and dull.	What a glorious weather today.	<i>Prva rečenica</i>

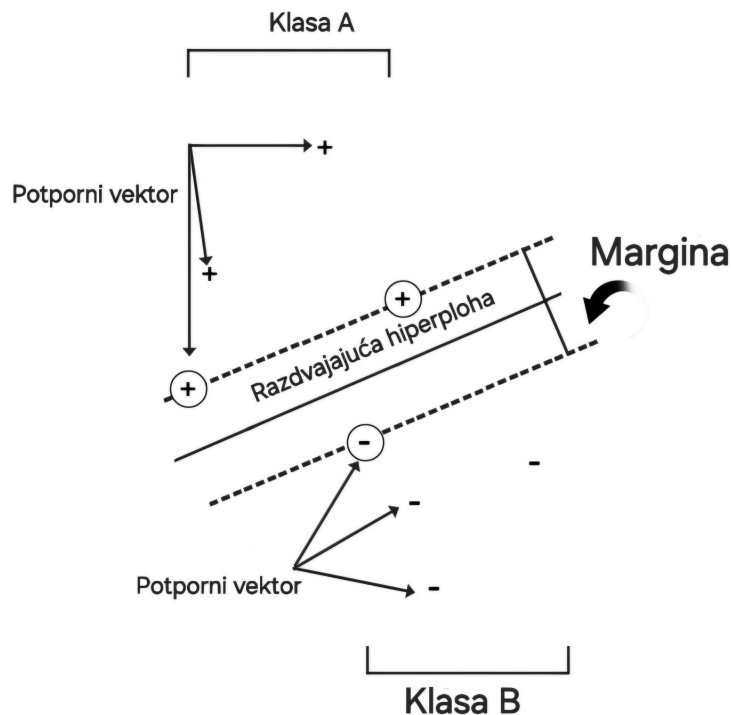
Tablica 4.7. Primjeri rečenica i pozicije sarkazma u zadatku prepoznavanja sarkastične inačice.

5. Modeli

Za rješavanje navedenih zadataka koristile su se metode koje će u nastavku biti predstavljene sa svojim ključnim komponentama. Posebna pažnja posvećena je metodama potpornih vektora (SVM) i transformatorima, uključujući model BERT. Konačno, objasnit će se i pristup višezadaćnom učenju koji je primijenjen u cilju poboljšanja performanci na svim zadacima.

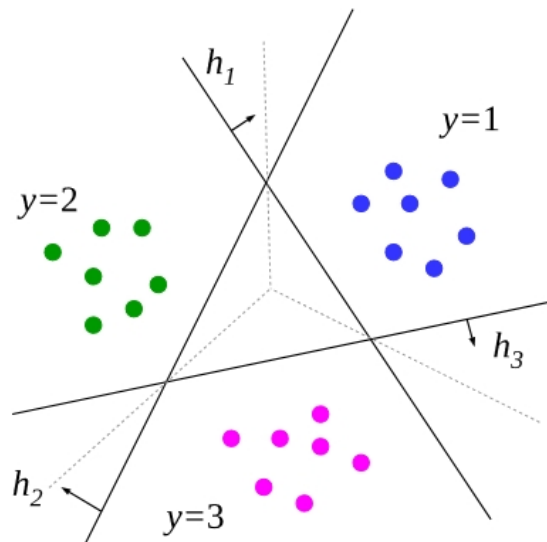
5.1. Stroj potpornih vektora

Stroj potpornih vektora [11] je linearan model strojnog učenja koji služi razdvajanju linearno odvojivih klasa u nekom vektorskom prostoru. Zasniva se na ideji odabira najbolje hiperravnine. Hiperravnina je razdvojna ploha koja služi kao klasifikator između dviju grupa informacija u vektorskom prostoru. Pojednostavljeno, u dvodimenzionalnom vektorskom prostoru to je linija koja odvaja dvije klase. Hiperravnina ima beskonačno mnogo, no optimizacijom se traži onu koja maksimalno povećava margine (udaljenosti najbližih točaka svake od klasa). Na slici 5.1. može se vidjeti primjer postavljene hiperplohe koja razdvaja klase.



Slika 5.1. Primjer klasifikacije stroja potpornih vektora [11]

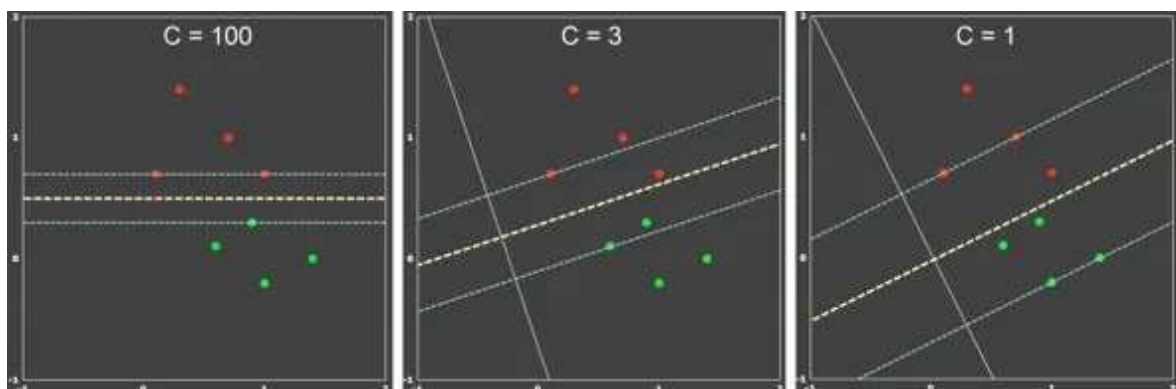
Algoritmi metode potpornih vektora primarno su dizajnirani za binarnu klasifikaciju, te se izvorno s pomoću njih ne može napraviti višeznačna klasifikacija zbog same prirode klasifikatora. Jedna hiperploha nužno ne mora moći odvojiti više od dvije grupe informacija te osim toga, čak i kada bi to bilo moguće, jedna točka ne može se nalaziti u dva potprostora odjednom, što onemogućuje višeznačnu klasifikaciju. No, zahvaljujući činjenici da se može odvojiti svaka od klasa koje klasificiramo od unije ostalih klasa u binarnu klasifikaciju, razdvajanjem podzadatka na šest različitih problema (s obzirom na to da imamo šest vrsta ironičnog govora), može se riješiti problem višeznačne klasifikacije s pomoću funkcionalnosti „MultiOutputClassifier“. Ta se metoda naziva jedan protiv ostalih (engl. *one versus rest*, OVR) [12]. Vizualni primjer klasifikacije pomoću metode OVR vidi se na slici 5.2.



Slika 5.2. Prikaz postavljanja hiperploha u metodi OVR [12]

U modelima su se optimizirala dva hiperparametra stroja potpornih vektora ne bismo li dobili model s najboljom metrikom za klasifikaciju. Ti hiperparametri C i γ određuju kako se stvara granica između klasa [13].

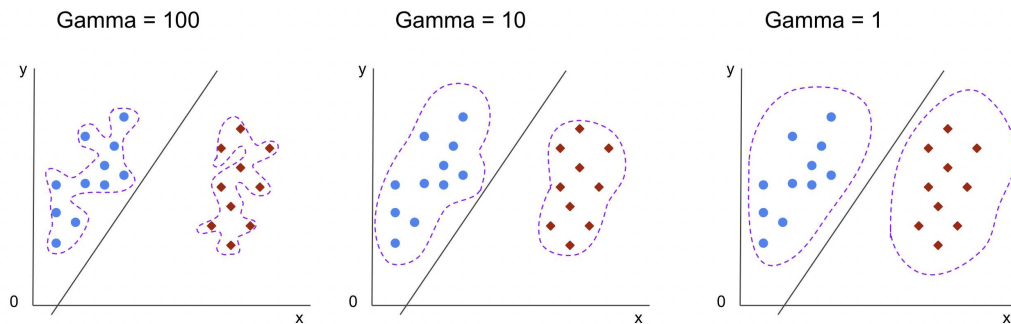
Hiperparametar C određuje koliko će model za vrijeme treniranja penalizirati pogrešku nad podacima. Visoka vrijednost hiperparametra C modelu implicira da napravi što manje pogrešaka te ima strožu granicu između klasa, unatoč mogućnosti za pretreniranje i lošijoj generalizaciji znanja. S druge strane, niska vrijednost hiperparametra C implicira modelu da napravi što veću moguću marginu između klasa, unatoč tome što onda ima više pogrešaka nad samim podacima. Prikaz kako hiperparametar C određuje margine prikazan je na slici 5.3.



Slika 5.3. Prikaz kako hiperparametar C određuje margine između klasa [13]

Hiperparametar γ određuje kompleksnost granice odlučivanja, koja označava vektor-

ski potprostor neke klase. Za visoke vrijednosti hiperparametra γ , model dopušta jako kompleksne granice prostora i pokušava ih približiti samim točkama što bliže, dok za niske vrijednosti granica postaje puno „opuštenija“. Niska vrijednost hiperparametra γ implicira jednostavniji model, no postoji mogućnost premale složenosti, dok velika vrijednost hiperparametra γ ima rizik od pretreniranja te gubljenja generalizacije znanja. Na slici 5.4. vidi se kako parametar γ utječe na granicu odluke između klasa.



Slika 5.4. Prikaz kako hiperparametar γ određuje granice odlučivanja za klase [13]

5.2. Transformatorski modeli

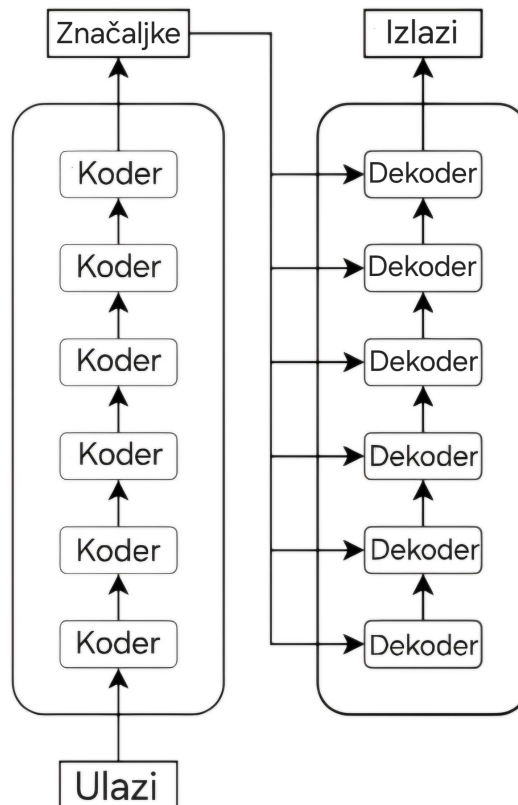
Transformatori su vrsta dubokih neuronskih mreža nastalih 2017. godine te su ubrzo postali ključni u području obrade prirodnog jezika, ali našli su primjenu i u širokom spektru zadataka unutar strojnog učenja i umjetne inteligencije [14].

Stvari koje ih čine posebnima su:

- Pozicijsko kodiranje (engl. *Positional encoding*): Umjesto da se oslanjaju na redoslijed riječi u rečenici, transformatori svakoj riječi pridodaju jedinstvene numeričke vrijednosti koje pružaju informacije o poziciji svakog tokena u sekvenciji kako bi znao točno o kojem se mjestu radi.
- Samopozornost (engl. *self-attention*): Mehanizam samopozornosti izračunava težine za svaku riječ u rečenici u odnosu na sve ostale riječi, čime si omogućava predviđanje sljedeće riječi u rečenici ovisno o statističkim vjerojatnostima.

Nakon transformacije rečenice u numeričku reprezentaciju i dodavanja pozicijskog kodiranja, te informacije šalju se u slojeve kodera, te potom u jednak broj dekodera. Izlaz kodera postavlja se na ulaz dekodera. Transformacijom zadnjeg izlaza zadnjeg dekodera

i korištenjem funkcije softmax dobivamo generiranje riječi te taj izlaz šaljemo kao ulaz dekoderu za generiranje iduće riječi. Funkcija softmax pretvara sirove izlazne vrijednosti (logite) u vjerojatnosti, tako da svaki izlazni element dobiva eksponencijalnu vrijednost, koja se zatim normalizira dijeljenjem s ukupnom sumom svih eksponencijalnih vrijednosti, osiguravajući da zbroj vjerojatnosti iznosi 1 [15]. Na slici 5.5. je prikazana arhitektura transformatora.



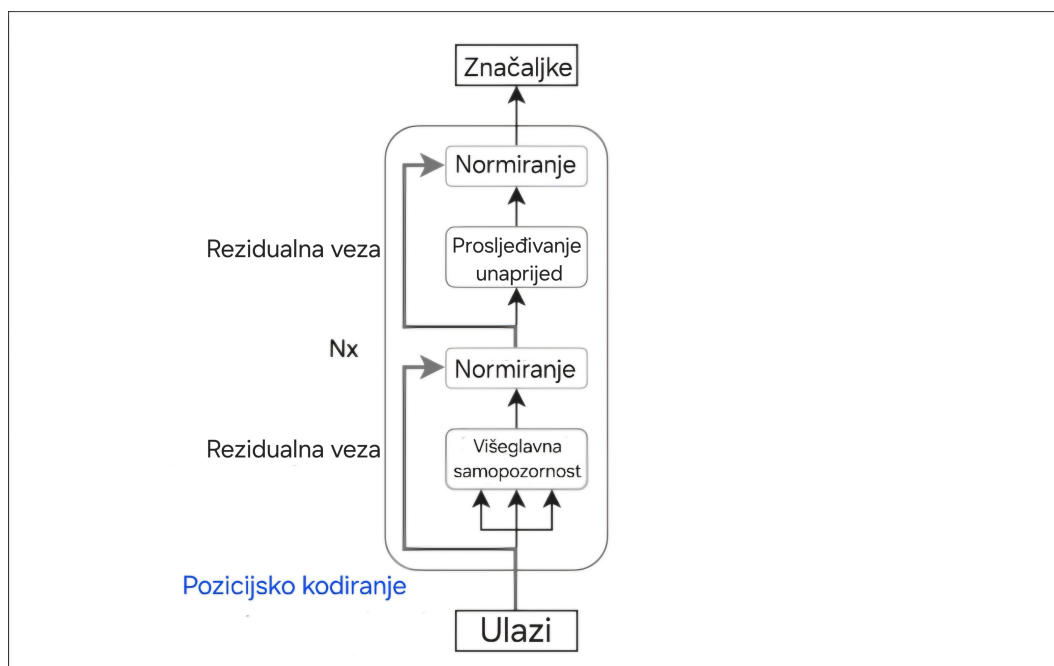
Slika 5.5. Prikaz prolaska informacija kroz arhitekturu transformatora [16]

5.2.1. Koder

Koderi [17] su ključna komponenta u transformatorima, koja služi za ekstrakciju informacija iz teksta i njihovo prenošenje u brojčani oblik. Koderski modeli kao što je BERT [6], koji smo koristili, mogu se koristiti za klasifikaciju teksta, analizu sentimenta, detekciju spam pošte, itd.

Sam koder, nakon obrade ulaznih podataka i dodavanja pozicijskog kodiranja, koristi mehanizam samopozornosti u sloju višeglavne pozornosti za obogaćivanje svakog tokena kontekstualnim informacijama iz rečenice. Ukratko, izračunava se matrica re-

zultata koja određuje naglasak svake riječi prema ostalima te se ti rezultati skaliraju. Primjenjuje se funkcija softmax, te se kombiniraju s vrijednostima kako bi se dobio izlazni vektor. Proces se ponavlja za svaku od glava sloja, a dobiveni izlazi prolaze kroz završni linearni sloj. Nakon normalizacije izlaza, on se predaje neuralnoj mreži, te se njezin izlaz ponovno normalizira i stavlja na početak te mreže za onoliko koliko kodera imamo. Na kraju zadnjeg kodera dobiva se skup vektora koji čini bogatu vektorsku reprezentaciju originalne rečenice. Taj se izlaz stavlja na ulaz dekodera ili se koristi u neke druge svrhe, kao što će se kasnije vidjeti. Na slici 5.6. vidi se struktura kodera.



Slika 5.6. Prikaz prolaska informacija kroz arhitekturu kodera [16]

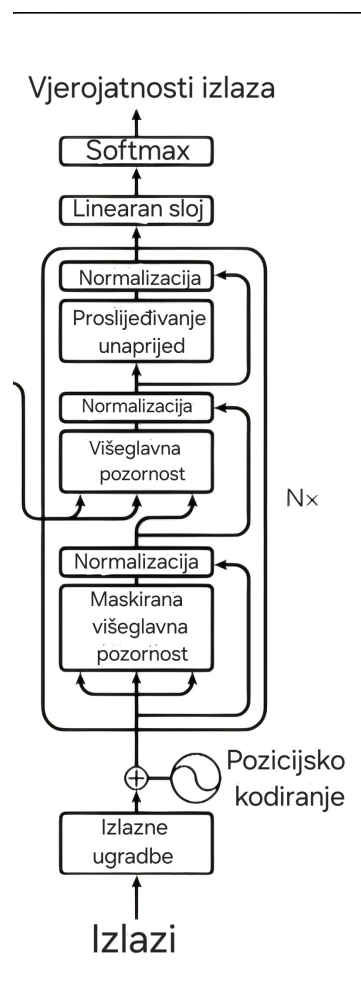
5.2.2. Dekoder

Zadatak dekodera [17] u transformatoru je generiranje tekstne sekvence. Slično koderu, dekodeer također sadrži podslojeve: dva sloja višeglavne pozornosti, unaprijedni sloj (engl. *feed-forward layer*), te rezidualne veze i normalizaciju sloja nakon svakog podsloja. Na početku se ulaz kodira te prolazi kroz maskirani mehanizam samopozornosti koji je sličan mehanizmu samopozornosti u koderskom sloju, osim što sprječava pažnju tokena na buduće tokene, osiguravajući time da svaki token nije pod utjecajem onih tokena u nastavku sekvence.

U idućem sloju višeglavne pozornosti uvode se izlazi iz kodera te se preko ovog sloja u

kombinaciji s dekoderovim izlazima interno identificiraju najrelevantniji dijelovi ulaza. Potom ti podaci prolaze kroz propagirajuću neuralnu mrežu i šalju se ponovno na ulaz idućeg dekodera.

Izlaz konačnog sloja dekodera transformira se u predviđenu sekvencu, obično kroz linearni sloj uz kombinaciju sa softmaxom za generiranje vjerojatnosti nad vokabularom kako bi odredio točne riječi. Ovaj se postupak ponavlja dok model ne predvidi poseban token koji se koristi za završetak niza. Na slici 5.7. vidi se struktura dekodera.



Slika 5.7. Arhitektura dekodera [18]

5.3. BERT

BERT je Googleov jezični model temeljen na transformatorovom sloju kodera. On je već treniran na njihovom velikom korpusu (engl. *pretrained*). Kod njegova ulaz cijeli se niz riječi unosi odjednom te se koristi obosmjernim treniranjem koristeći kontekst s obje

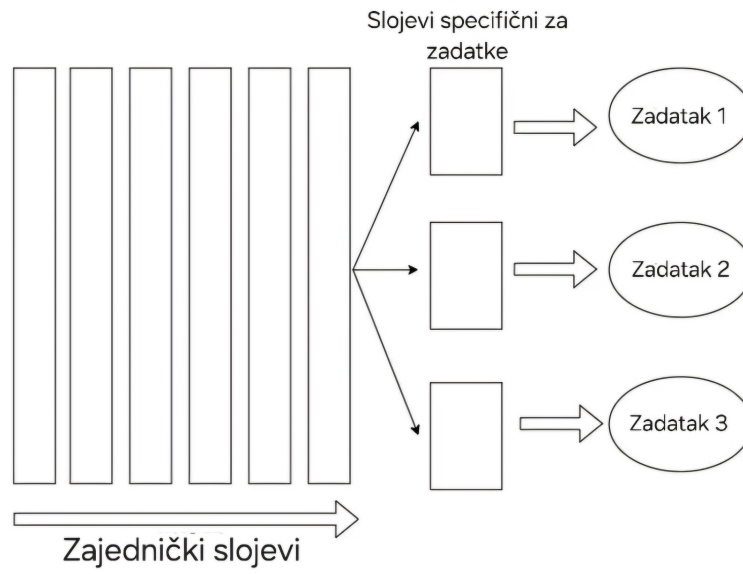
strane riječi, što pripomaže u dubljem shvaćanju riječi [6]. Pri samom procesiranju ulaznih podataka prije ulaska u model BERT rečenicama se nakon tokenizacije pridodaju još 2 vrste tokena, token *[CLS]* i token *[SEP]*. Token *[CLS]* [19] uvijek dolazi na prvo mjesto u tokeniziranoj rečenici, odnosno lijevo od prve riječi u ulazu. On enkapsulira cijelo značenje ulaza te se propagira kroz model i omogućava modelu znanje o kontekstu i odnosima riječi u ulazu te mu omogućava klasifikaciju. Token *[SEP]* [20] postavlja se između više rečenica jednog ulaza te na kraj zadnje rečenice u ulazu. Njegova svrha je označiti bitnu informaciju modelu o kraju rečenice i početku nove kako bi model bolje mogao shvatiti kontekstualne informacije ulaza.

5.4. Fino ugađanje transformatorskih arhitektura

Fino ugađanje (engl. *fine-tuning*) omogućava da se na pretreniranom modelu kao što je BERT nadoda klasifikacijska glava koja se sastoji od linearnog sloja čije se interne težine mijenjaju tako što se model trenira na određenom zadatku. Time BERT postaje sposobniji za rješavanje konkretnog problema za koji je treniran [6]. Upravo to primijenilo se na sva tri podzadatka.

5.5. Višezadačno učenje

Višezadačno učenje (engl. *multi-task learning*) je vrsta strojnog učenja u kojoj se model trenira za obavljanje više zadataka istovremeno, koristeći zajedničke težine za poboljšanje performanci na svim zadacima [21]. Umjesto treniranja zasebnih modela za svaki zadatak, model višezadačnog učenja je dizajniran za dijeljenje informacija između zadataka kako bi došao do poboljšanih performanci za individualne zadatke, prikupljajući veće generalno znanje i poboljšavajući same klasifikatore. Na kraj modela postavlja se onoliko klasifikacijskih glava koliko je samih zadataka na kojima se trenira, te se testni podaci, ovisno o njihovoj vrsti, šalju na zadanu klasifikacijsku glavu kao što je prikazano na slici 5.8.



Višezadačno učenje

Slika 5.8. Arhitektura transformatora za višezadačno učenje [22]

6. Metodologija

U ovom poglavlju bit će predstavljene metode korištene za rješavanje zadataka zajedno s njihovim ključnim komponentama. Metodologija se sastoji od nekoliko faza, uključujući preobradu podataka, te individualne opise konkretne uporabe različitih modela za klasifikaciju podzadataka.

Prvo, opisat će se preobrada podataka uz opis razdvajanja testnih skupova. Nakon toga, detaljno će se objasniti algoritmi stroja potpornih vektora (SVM) koji su korišteni kao referentni (engl. *baseline*) modeli za sve podzadatke. Za implementaciju modela stroja potpornih vektora korištena je biblioteka Scikit-learn [23]. Ta biblioteka odabrana je zbog svoje popularnosti i jednostavnosti korištenja. Korištenje Scikit-learn omogućuje lako podešavanje hiperparametara samog stroja potpornih vektora te omogućava brzo konfiguriranje istog.

Potom će se predstaviti transformator modeli, s posebnim naglaskom na model BERT koji je korišten za sve zadatke. Objasniti će se proces treniranja modela BERT, uključujući fino ugadanje i mehanizam ranog zaustavljanja kako bi se spriječilo prenaučavanje. Za implementaciju modela BERT, korišten je radni okvir Hugging Face Transformers [24]. Hugging Face Transformers je odabran zbog svoje sveobuhvatne podrške za različite modele transformatora, jednostavne integracije i aktivne zajednice koja doprinosi poboljšanjima i održavanju koda. Ovaj radni okvir pruža unaprijed trenirane modele i alate za fino podešavanje modela za specifične zadatke, što olakšava prilagodbu BERT-a na zadatke detekcije sarkazma i njegovih granulata.

Na kraju, detaljno će se opisati pristup višezadačnom učenju koji je primijenjen u cilju poboljšanja performanci modela na svim zadacima. Ovaj pristup uključuje treniranje modela za obavljanje više zadataka istovremeno, koristeći zajedničke težine za poboljša-

nje performanci na svim zadacima.

6.1. Preobrada podataka

U koraku predobrade podataka za svaki zadatak maknuti su dijelovi teksta koji počinju s “http”, “@” i “www” s obzirom na to da se smatra da njihova tokenizacija ne pripomaže ni treniranju ni evaluaciji te se smatraju redundantnim informacijama.

Nakon prvog koraka obrade podataka testni je skup razdvojen na testni skup i evaluacijski skup tako da je evaluacijski skup 10 posto veličine originalnog testnog skupa. Također, omjer oznaka je konzistentan kroz nove skupove, na način da se u testnom i evaluacijskom skupu za binarnu detekciju sarkazma omjer sarkastičnih rečenica ostane isti, dok su se u zadatku klasifikacije potkategorija sarkazma i zadatku prepoznavanja sarkastične inačice testne skupovi razdvajali tako da se održi omjer rečenica koje spadaju pod sarkazam istim, te omjer rečenica koje su na prvome mjestu sarkastične isti.

	Skup za učenje	Skup za validaciju	Skup za testiranje
Zadatak 1	3120	347	1400
Zadatak 2	3280	365	1400
Zadatak 3	780	87	200

Tablica 6.1. Završne veličine skupova

U sva tri referentna modela stroja potpornih vektora modela korišten je *word_tokenize* tokenizator te *Word2Vec* vektorizator kako bi se tokenizirane rečenice pretvorile u vektore brojeva koje stroj potpornih vektora može prihvatiti kao ulaze.

Za sva tri individualna modela BERT korišten je *BertTokenizer* koji automatski obradom rečenice napravi format koji BERT može prihvatiti i dodaje im *[CLS]* i *[SEP]* tokene, koji u sebi sadrže informacije o samom ulazu te oznaku kraja rečenice. Proces prije tokenizacije bio je identičan kao i u korištenju stroja potpornih vektora.

U svim modelima višezadačnog učenja korišten je *MultitaskBertForSequenceClassification* uz *Autotokenizer* za tokenizaciju.

6.2. Stroj potpornih vektora

Binarna klasifikacija sarkazma

Nakon jednostavne tokenizacije i vektorizacije rečenica, provedeno je pretraživanje po rešetci (engl. *grid search*) za vrijednosti hiperparametara γ i C kako bi se odredile najbolje vrijednosti na skupu za treniranje binarne klasifikacije.

Pretraživanje po rešetci je metoda optimizacije koja sustavno pretražuje unaprijed definirani prostor hiperparametara kako bi pronašla optimalne vrijednosti za određeni model. U ovom slučaju, početne vrijednosti za γ i C su postavljene na 0.0625, te se svaka od njih množila s 2 u svakoj idućoj iteraciji. Ukupno je provedeno sedam iteracija, što je rezultiralo s 49 različitih kombinacija hiperparametara.

Za svaku kombinaciju hiperparametara model je evaluiran na evaluacijskom skupu, te je izračunata vrijednost F_1 . Tako je dobiveno 49 vrijednosti F_1 na evaluacijskom skupu, po jedna za svaku kombinaciju hiperparametara.

Hiperparametri koji su dali najbolju vrijednost F_1 na evaluacijskom skupu su zabilježeni, te je model s tim hiperparametrima korišten kao referentni model.

Originalna rečenica	Tokenizirana rečenica
I would date any of these men	[i, would, date, any, of, these, men]

Tablica 6.2. Primjer originalne i tokenizirane rečenice

Višeoznačna klasifikacija potkategorija sarkazma

Za referentni model drugog zadatka također su tokenizirane i vektorizirane rečenice, no prije toga su rečenice koje imaju više od jedne oznake razdvojene u onoliko primjera koliko pozitivnih oznaka sadrže tako da je primjerice rečenica koja je sarkastična i ironična 2 puta dodana u skup podataka za trening, jednom kao sarkastična, a jednom kao ironična rečenica. Svaka rečenica u skupu za treniranje koja je bila označena kao ne-sarkastična korištena je tako što je za svih šest oznaka postavljena negativna vrijednost. Potom je ponovno napravljeno pretraživanje po rešetci u 7 iteracija za jednake početne vrijednosti hiperparametara i najbolji je model zadržan kao referentni model.

Raspoznavanje sarkastičnih i nesarkastičnih inačica

Za treći podzadatak tokenizirane su originalna rečenica i njena nesarkastična inačica, te je uzeta sarkastična rečenica i na njezin kraj je dodan *[SEP]* token, koji je služio za razdvajanje rečenice od njezine parafrazirane verzije. Nakon toga na taj niz znakova dodana je parafrazirana rečenica, stvarajući spojenu rečenicu s oznakom "nulta rečenica", jer se sarkastična rečenica nalazila lijevo od tokena *[SEP]*. Zatim je napravljena jednaka stvar, ali obrnutim redom: uzeta je parafrazirana rečenica, nadodan je token *[SEP]* na njezin kraj i potom dodana originalna sarkastična rečenica. Ova „obrnuto spojena“ rečenica dobila je oznaku 1, jer se sarkastična rečenica nalazila s desne strane tokena.

Originalna rečenica	Refrazirana rečenica	Rečenica spojena sa <i>[SEP]</i>
[see, brexit, is, going, well]	[brexit, really, isn, ', t, going, to, plan]	[see, brexit, is, going, well, [SEP], brexit, really, isn, ', t, going, to, plan]

Tablica 6.3. Primjer tokeniziranog tvita i njegove refrazirane inačice prije i nakon spajanja sa *[SEP]* tokenom

Kako je svaki od 867 potencijalnih primjera imao dvije reprezentacije za treniranje (jednu „spojenu“ i jednu „obrnuto spojena“), naizmjenično se birala jedna od njih za treniranje, osiguravajući da se svaka rečenica pojavljuje jednom i samo jednom u skupu podataka. Na taj način oznaka položaja sarkastične rečenice bila je ravnomjerno raspoređena između "nulte rečenice" i "prve rečenice". Tek se nakon toga skup razdvojio na evaluacijski i skup za trening. Zatim su se s pomoću pretraživanja po rešetci od 15 iteracija dobile najbolje vrijednosti hiperparametara C i γ te se taj model i koristio na skupu za testiranje.

6.3. BERT

Za prvi i treći zadatak korišten je model BertForSequenceClassification, dok se u drugom podzadatku koristio model MultiLabelBertClassifier. Implementacije ovih modela su preuzete iz radnog okvira Hugging Face Transformers [24]. U sva tri zadatka tokenizacija se provela na isti način kao i za stroj potpornih vektora, ali korištenjem BertTokenizer. Nakon tokenizacije, u sva tri podzadatka, za vrijeme finog ugađanja hiperparametara unutar treninga, primijenio se mehanizam za rano zaustavljanje treni-

ranja u slučaju da dođe do gubitka na evaluacijskom skupu podataka.

Rano zaustavljanje tehnika je koja se primjenjuje u treniranju neuronskih mreža kako bi se spriječilo prenaučavanje (engl. *overfitting*) [25]. Prenaučavanje se događa kada model počne učiti statistički šum u skupu podataka za treniranje, što rezultira smanjenom mogućnosti modela da ispravno predviđa na novim podacima te slabijom generalizacijom znanja. Tijekom treniranja modela, nakon svakog odsječka treniranja evaluira se model na podacima za evaluaciju. U trenutku kad pogreška na evaluacijskom skupu naraste, zaustavlja se treniranje modela i može se početi testirati. Grafički primjer ranog zaustavljanja vidi se na slici 6.1.



Slika 6.1. Prikaz trenutka ranog zaustavljanja za vrijeme treniranja [26]

6.3.1. Modeli višezadačnog učenja

Zadaci koji su obrađivani su srodni, pa je primijenjena metoda višezadačnog učenja kako bi se ispitalo koliko signali iz jednog zadatka mogu pomoći klasifikaciji drugoga. Napravili smo tri modela višezadačnog učenja, gdje je svaki model bio treniran na dva zadatka

istovremeno. Modeli i njihovi zadaci su prikazani u tablici 6.4.

Model	Prvi zadatak	Drugi zadatak
MT12	Zadatak 1	Zadatak 2
MT13	Zadatak 1	Zadatak 3
MT23	Zadatak 2	Zadatak 3

Tablica 6.4. Kombinacije zadataka za modele višezadačnog učenja

Za tokenizaciju se koristio `AutoTokenizer` te je proces tokenizacije proveden jednako kao i u prethodnim modelima. Koristio se model `MultitaskBertForSequenceClassification`, koji je baziran na arhitekturi BERT.

Primijenjen je mehanizam ranog zaustavljanja treniranja, gdje se treniranje zaustavlja ako bilo koji od dva validacijska gubitka (jer se rješavaju dva podzadatka odjednom) naraste u bilo kojem trenutku.

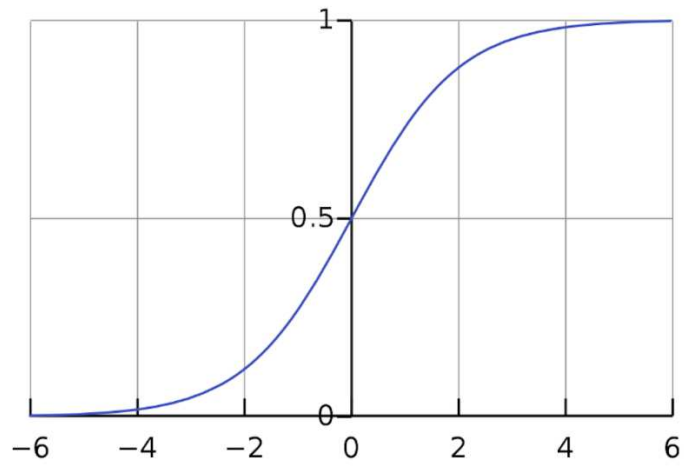
6.4. Obrada izlaza modela BERT

Pri evaluaciji modela BERT na njegovu izlazu dobiveni su logiti, brojevi koji su prosljeđeni u aktivacijsku sigmoidnu funkciju. Sigmoidna logistička funkcija je poseban oblik logističke funkcije koja sve realne brojeve pretvara u vrijednost između 0 i 1. Formula sigmoidne funkcije je:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

gdje je $\sigma(x)$ izlaz funkcije, x je ulazni logit, a e je Eulerov broj.

Ova funkcija pretvara logite u vjerojatnosti koje se koriste za klasifikaciju. Graf sigmoidne funkcije vidljiv je na slici 6.2.



Slika 6.2. Graf sigmoidne funkcije [27]

Koristila se granična vrijednost od 0.5. To znači da je svaki logit, nakon što je prošao kroz sigmoidnu funkciju, ako je veći od 0.5, postao pozitivni razred, dok su svi ostali postali negativni razred.

7. Rezultati

7.1. Vrednovanje modela

Evaluacijska metrika po kojoj su se vrednovali modeli je vrijednost $\text{macro-}F_1$ [28]. Vrijednost $\text{macro-}F_1$ globalna je mjera koja izračunava vrijednost F_1 za svaku klasu individualno, te kao rezultat vraća njihov prosjek. Time se postiže podjednak utjecaj svake klase u klasifikatoru na samu vrijednost, kako bi se spriječilo da neka zastupljenija klasa ne bi previše utjecala na percepciju uspješnosti klasifikacije. Za binarnu klasifikaciju s jednom klasom vrijednost $\text{macro-}F_1$ računa se jednako kao i obična F_1 -vrijednost za jednu klasu. Vrijednost F_1 je mjera vrednovanja koja kombinira preciznost i odziv pomoću njihove harmonijske sredine.

Definirana je kao: [29]

$$F1 = 2 \cdot \frac{P \cdot R}{P + R}$$

gdje je P preciznost, a R odziv. Preciznost i odziv se mogu izračunati iz elemenata matrice zabune (konfuzijske matrice) na sljedeći način:

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

gdje su: - TP (engl. *True Positives*) stvarno pozitivni primjeri, - FP (engl. *False Positives*) lažno pozitivni primjeri, - FN (engl. *False Negatives*) lažno negativni primjeri.

Vrijednost F_1 je korisna mjera kada želimo uzeti u obzir i preciznost i odziv modela, a posebno je prikladna u situacijama gdje postoji neuravnoteženost klasa.

Pretpostavimo da imamo sljedeću konfuzijsku matricu:

	Stvarno pozitivni	Stvarno negativni
Predviđeno pozitivni	TP = 50	FP = 10
Predviđeno negativni	FN = 5	TN = 35

Tablica 7.1. Konfuzijska matrica

Iz ove konfuzijske matrice možemo izračunati preciznost i odziv:

$$P = \frac{TP}{TP + FP} = \frac{50}{50 + 10} = 0.833$$

$$R = \frac{TP}{TP + FN} = \frac{50}{50 + 5} = 0.909$$

Sada možemo izračunati vrijednost F_1 :

$$F_1 = 2 \cdot \frac{P \cdot R}{P + R} = 2 \cdot \frac{0.833 \cdot 0.909}{0.833 + 0.909} = 2 \cdot \frac{0.758}{1.742} = 0.870$$

Nakon računanja vrijednosti F_1 za svaku od klasa višeklasne klasifikacije, vrijednost Macro- F_1 se računa kao prosjek tih vrijednosti:

$$\text{Macro } F_1 = \frac{1}{K} \sum_{i=1}^K F_{1_i}$$

Modeli bazirani na modelu BERT trenirani su 5 puta. Za svaku od pet iteracija treniranja korišteni su isti podaci, ali je redoslijed ulaska u trening randomiziran pomoću vrijednosti sjemena (engl. *seed*). Vrijednost seed dodijeljena je funkciji `RandomSampler`, koja određuje nasumični redoslijed grupa u treniranju. Tih pet vrijednosti daje, za isti skup podataka za treniranje i iste hiperparametre, pet vrijednosti macro- F_1 koje se mogu koristiti u t-testu za određivanje postoji li značajna razlika u vrijednostima Macro- F_1 te zaključivanje ima li višezadačno učenje značajne pozitivne ili negativne posljedice na uspješnost klasifikacije podzadataka [30].

T-test je statistička metoda koja se koristi za utvrđivanje postoji li značajna razlika između srednjih vrijednosti dviju skupina. Za provođenje t-testa nad nekim skupovima uzoraka potrebno je osigurati da ti uzorci dolaze iz populacija koje su normalno distribuirane i čije varijance nisu poznate. Za sve t-testove korištena je granična vrijednost 0.05. P-vrijednost opisuje kolika je vjerojatnost dobivanja rezultata pod uvjetom da nul-hipoteza eksperimenta vrijedi. Provedeno je šest t-testova za usporedbu vrijednosti individualnih modela sa svakim od uparenih modela koji sadrže podzadatak klasificiran u individualnom zadatku.

Prije provođenja t-testa, potrebno je odrediti dolaze li podaci iz normalne distribucije. Za provjeru normalnosti podataka proveden je Shapiro-Wilk test za skup dobivenih macro- F_1 vrijednosti svakog zadatka. Nul-hipoteza Shapiro-Wilk testa je da podaci proizlaze iz normalno distribuirane populacije. Kao granična vrijednost korištena je p-vrijednost od 0.05. Uz ovu graničnu vrijednost, niti u jednom slučaju nije odbačena nul-hipoteza, čime je pretpostavka o normalnosti zadovoljena.

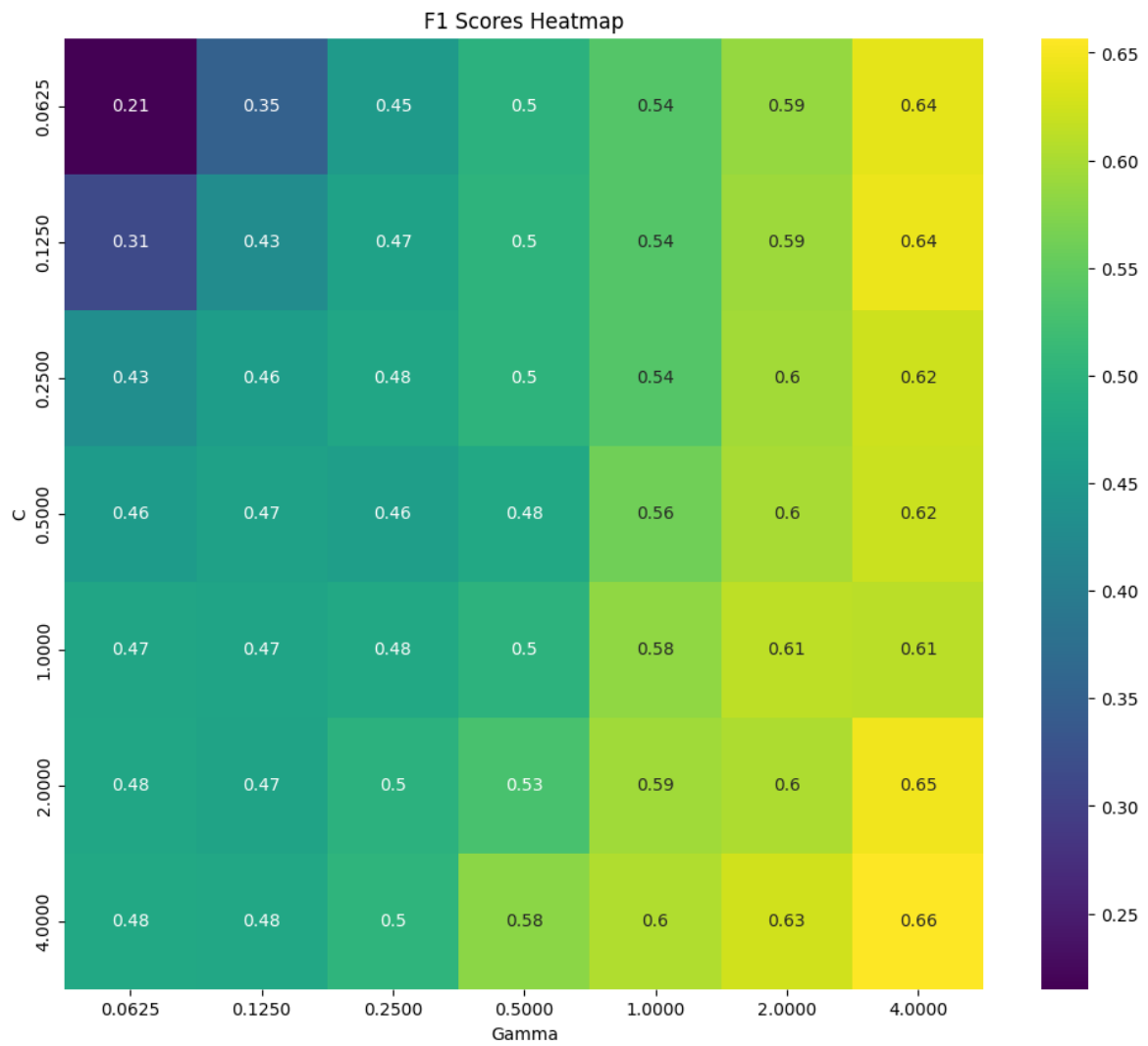
7.2. Rezultati referentnih modela

Rezultati početnih modela napravljenih sa strojem potpornih vektora te matrice vrijednosti F_1 na skupu za testiranje su:

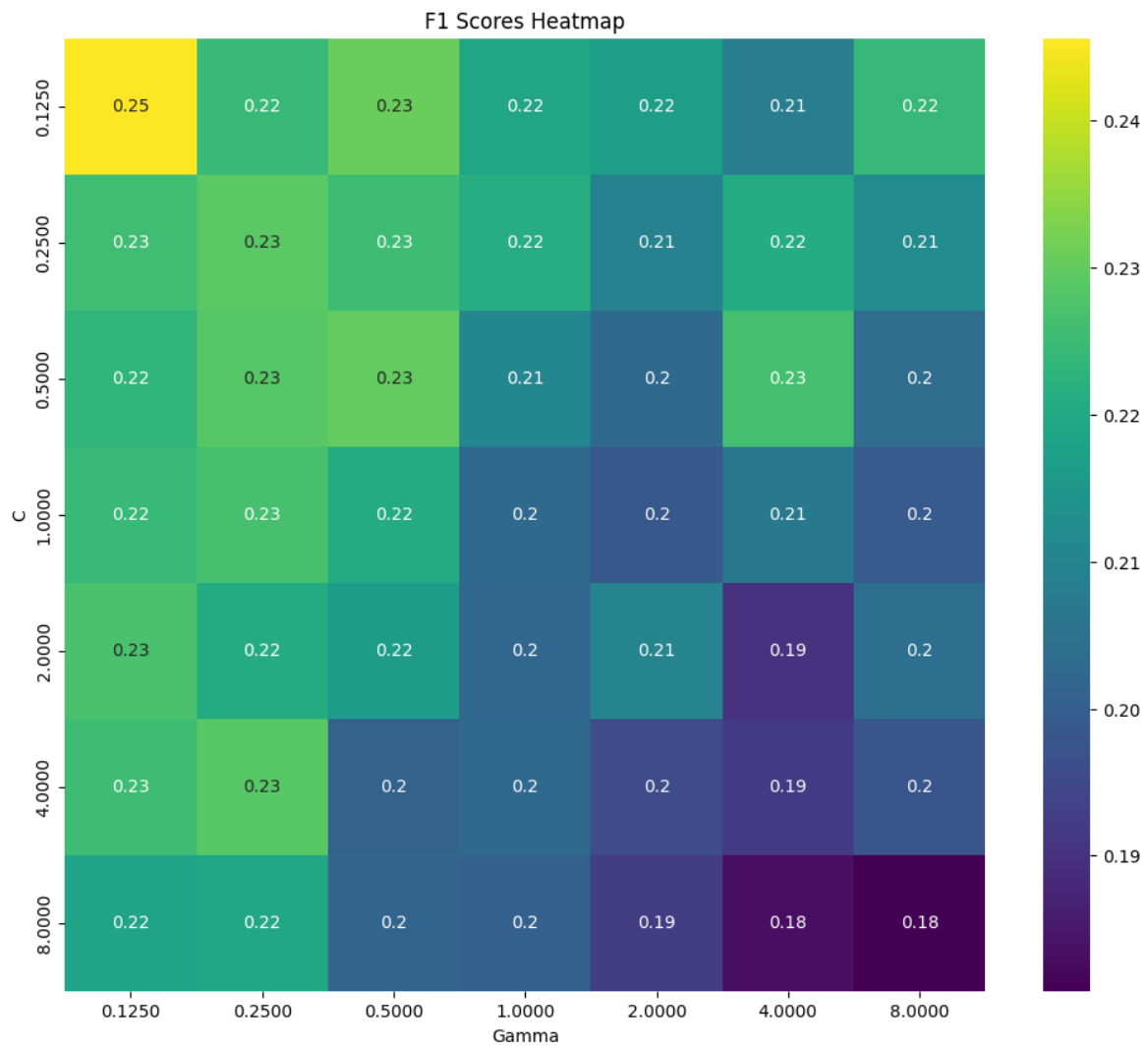
Zadatak	C	γ	Vrijednost Macro F_1
1	4	4	0.49
2	0.25	0.25	0.06
3	128	2	0.46

Tablica 7.2. vrijednosti C , γ i Macro F_1 za referentne modele

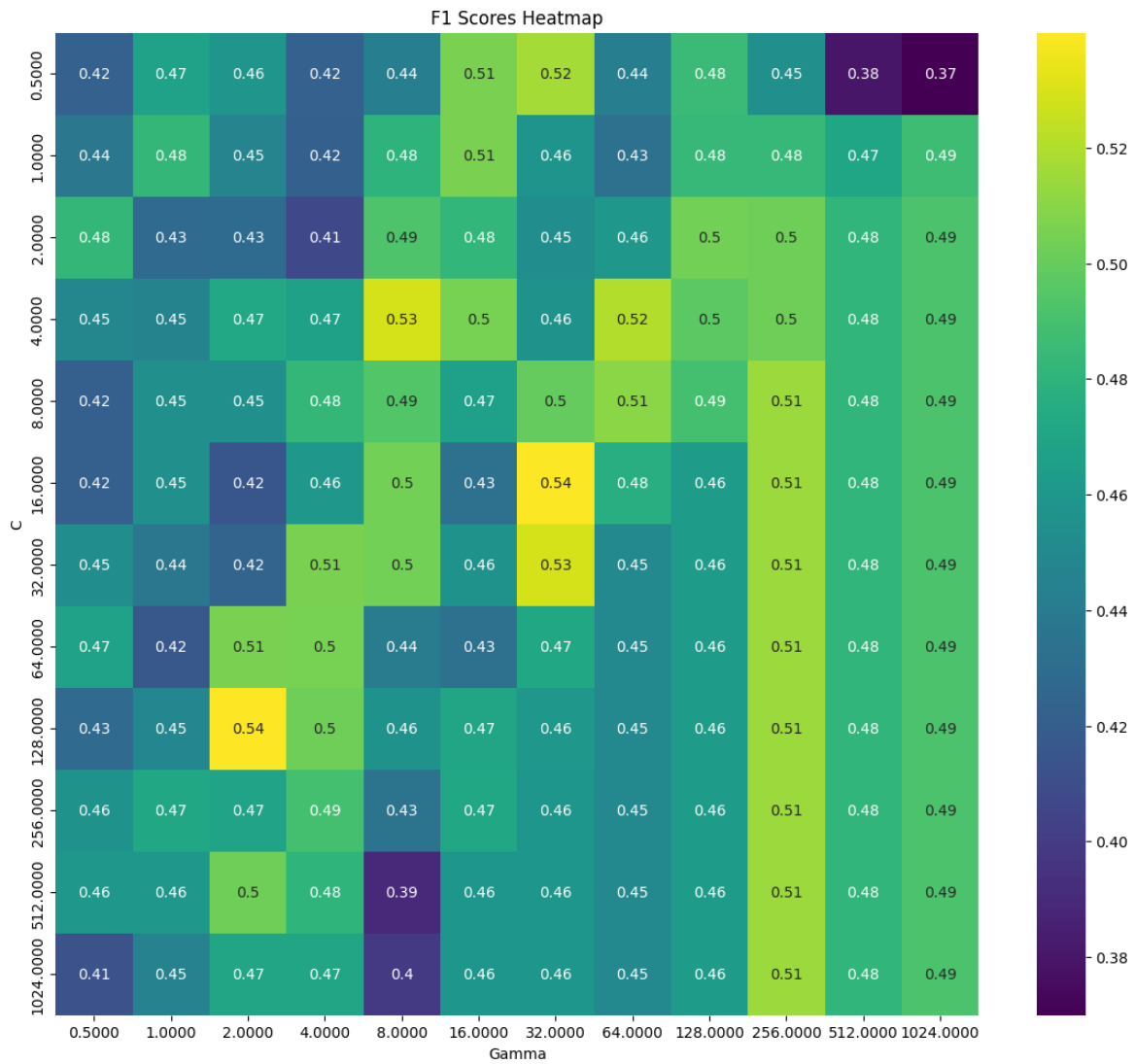
Na slikama 7.1., 7.2. i 7.3. vide se rezultati pretraživanja po rešetci u svrhu traženja najviše F_1 vrijednosti na evaluacijskom skupu.



Slika 7.1. Traženje najbolje vrijednosti F_1 za prvi referentni model na evaluacijskom skupu podataka pomoću pretraživanja po rešetci



Slika 7.2. Traženje najbolje vrijednosti F_1 za drugi referentni model na evaluacijskom skupu podataka pomoću pretraživanja po rešetci



Slika 7.3. Traženje najbolje vrijednosti F_1 za treći referentni model na evaluacijskom skupu podataka pomoću pretraživanja po rešetci

7.3. Rezultati testiranja

U ovome potpoglavlju bit će prikazani rezultati različitih modela baziranih na transformatorima te će se usporediti njihovi rezultati.

Hiperparametar	Vrijednost
Broj epoha	4
Veličina grupe	32
Stopa učenja	2e-5

Tablica 7.3. Hiperparametri korišteni u modelima BERT

Korištenje zvjezdica u rezultatima

Zvjezdica (*) se koristi za označavanje statistički značajnih rezultata. U ovom kontekstu, rezultat se smatra statistički značajnim ako je p-vrijednost manja od 0.05. Zvjezdica će biti dodana uz prosječne vrijednosti u tablicama gdje je t-test pokazao značajnu razliku.

7.3.1. Zadatak 1

Seed	Task1	MT12-1	MT13-1
1	0.58	0.6	0.46
25	0.6	0.56	0.55
37	0.61	0.57	0.65
56	0.57	0.61	0.56
73	0.61	0.59	0.46
Prosjek	0.59	0.59	0.54
Shapiro p vrijednost	0.25	0.75	0.38

Tablica 7.4. Vrijednosti za Task1, MT12-1 i MT13-1

S obzirom na to da su p-vrijednosti Shapiro-Wilk testa za sve skupove veće od 0.05, može se pretpostaviti da podaci dolaze iz normalno distribuiranih populacija. Stoga su svi kandidati za t-test. Provedeni su t-testovi između Task1 i MT12-1 te između Task1 i MT13-1.

Polazna hipoteza za t-test je da ne postoji značajna razlika između srednjih vrijednosti dviju skupina (Task1 i MT12-1, odnosno Task1 i MT13-1).

Za prvi t-test hipoteze su:

- H_0 : Srednje vrijednosti Macro- F_1 su iste.

$$H_0 : \mu_{\text{Task1}} = \mu_{\text{MT12}}$$

- H_1 : Srednje vrijednosti Macro- F_1 nisu iste.

$$H_1 : \mu_{\text{Task1}} \neq \mu_{\text{MT12}}$$

Rezultati t-testa:

- **Task1 i MT12-1:** $t(4) = 0.6489, p = 0.5346$

S obzirom na to da je p-vrijednost veća od 0.05, ne odbacuje se početna hipoteza te se zaključuje da prijenosno učenje modela s prvim i drugim podzadatom u isto vrijeme nema značajan utjecaj na kvalitetu klasifikacije za prvi podzadatak.

Za drugi t-test hipoteze su:

- H_0 : Srednje vrijednosti Macro- F_1 su iste.

$$H_0 : \mu_{\text{Task1}} = \mu_{\text{MT13}}$$

- H_1 : Srednja vrijednost Macro- F_1 Task1 modela je veća nego srednja vrijednost Macro- F_1 MT13 modela na podzadatu 1.

$$H_1 : \mu_{\text{Task1}} > \mu_{\text{MT13}}$$

Rezultati t-testa:

- **Task1 i MT13-1:** $t(4) = 1.5892, p = 0.1507$

S obzirom na to da je p-vrijednost veća od 0.05, ne odbacuje se početna hipoteza te se zaključuje da prijenosno učenje modela s prvim i trećim podzadatom u isto vrijeme nema značajan utjecaj na kvalitetu klasifikacije za prvi podzadatak.

7.3.2. Zadatak 2

Seed	Task2	MT12-2	MT23-2
1	0.03	0.02	0
25	0.05	0	0
37	0.04	0.01	0
56	0.02	0.02	0
73	0	0	0
Prosjek	0.03	0.01	0*
Shapiro p vrijednost	0.93	0.12	1

Tablica 7.5. Vrijednosti za Task2, MT12-2 i MT23-2

S obzirom na to da su p-vrijednosti Shapiro-Wilk testa za sve skupove veće od 0.05, može se pretpostaviti da podaci dolaze iz normalno distribuiranih populacija. Stoga su svi kandidati za t-test. Provedeni su t-testovi između Task2 i MT12-2 te između Task2 i MT23-2.

Polazna hipoteza za t-test je da ne postoji značajna razlika između srednjih vrijednosti dviju skupina (Task2 i MT12-2, odnosno Task2 i MT23-2).

Za prvi t-test hipoteze su:

- H_0 : Srednje Macro-F1 vrijednosti su iste.

$$H_0 : \mu_{\text{Task2}} = \mu_{\text{MT12}}$$

- H_1 : Srednja Macro-F1 vrijednost Task2 modela je veća nego srednja Macro-F1 vrijednost MT12 modela na podzadatku 2.

$$H_1 : \mu_{\text{Task2}} > \mu_{\text{MT12}}$$

Rezultati t-testa:

- **Task2 i MT12-2:** $t(4) = 1.8566, p = 0.1005$

S obzirom na to da je p-vrijednost veća od 0.05, ne odbacuje se početna hipoteza te

se zaključuje da prijenosno učenje modela s prvim i drugim podzadatom u isto vrijeme nema značajan utjecaj na kvalitetu klasifikacije za drugi podzadatak.

Za drugi t-test hipoteze su:

- H_0 : Srednje Macro-F1 vrijednosti su iste.

$$H_0 : \mu_{\text{Task2}} = \mu_{\text{MT23}}$$

- H_1 : Srednja Macro-F1 vrijednost Task2 modela je veća nego srednja Macro-F1 vrijednost MT23 modela na podzadatumu 2.

$$H_1 : \mu_{\text{Task2}} > \mu_{\text{MT23}}$$

Rezultati t-testa:

- **Task2 i MT23-2:** $t(4) = 3.2549, p = 0.0116$

S obzirom na to da je p-vrijednost manja od 0.05, odbacuje se početna hipoteza te se zaključuje da prijenosno učenje modela s drugim i trećim podzadatom u isto vrijeme ima značajan negativan utjecaj na kvalitetu klasifikacije za drugi podzadatak.

7.3.3. Zadatak 3

Seed	Task3	MT13-3	MT23-3
1	0.68	0.50	0.56
25	0.72	0.61	0.73
37	0.73	0.54	0.56
56	0.75	0.70	0.72
73	0.70	0.36	0.59
Prosjek	0.72	0.54*	0.63
Shapiro p vrijednost	0.98	0.98	0.05

Tablica 7.6. Vrijednosti za Task3, MT13-3 i MT23-3

S obzirom na to da su p-vrijednosti Shapiro-Wilk testa za sve skupove veće od 0.05, može se pretpostaviti da podaci dolaze iz normalno distribuiranih populacija. Stoga su svi kandidati za t-test. Provedeni su t-testovi između Task3 i MT13-3 te između Task3 i MT23-3.

Polazna hipoteza za t-test je da ne postoji značajna razlika između srednjih vrijednosti dviju skupina (Task3 i MT13-3, odnosno Task3 i MT23-3).

Za prvi t-test hipoteze su:

- H_0 : Srednje Macro-F1 vrijednosti su iste.

$$H_0 : \mu_{\text{Task3}} = \mu_{\text{MT13}}$$

- H_1 : Srednje Macro-F1 vrijednosti nisu iste.

$$H_1 : \mu_{\text{Task3}} \neq \mu_{\text{MT13}}$$

Rezultati t-testa:

- **Task3 i MT13-3:** $t(4) = 2.9973, p = 0.0171$

S obzirom na to da je p-vrijednost manja od 0.05, odbacuje se početna hipoteza te se zaključuje da prijenosno učenje modela s prvim i trećim podzadatom u isto vrijeme ima značajan negativan utjecaj na kvalitetu klasifikacije za treći podzadatak.

Za drugi t-test hipoteze su:

- H_0 : Srednje Macro-F1 vrijednosti su iste.

$$H_0 : \mu_{\text{Task3}} = \mu_{\text{MT23}}$$

- H_1 : Srednja Macro-F1 vrijednost Task3 modela je veća nego srednja Macro-F1 vrijednost MT23 modela na podzadatumu 3.

$$H_1 : \mu_{\text{Task3}} > \mu_{\text{MT23}}$$

Rezultati t-testa:

- **Task3 i MT23-3:** $t(4) = 2.0870, p = 0.0703$

S obzirom na to da je p-vrijednost veća od 0.05, ne odbacuje se početna hipoteza te se zaključuje da prijenosno učenje modela s drugim i trećim podzadatkom u isto vrijeme nema značajan utjecaj na kvalitetu klasifikacije za treći podzadatak.

7.3.4. Analiza

Kako bi se detaljno razumjele razlike u performansama, analizirani su rezultati za svaki individualni zadatak zadatak. Fokus je stavljen na razlike između pristupa individualnog i višezadaćnog učenja te koji pristup pomaže ili odmaže kojem zadatku.

Zadatak 1

Za zadatak binarne klasifikacije sarkazma (Task1), višezadaćno učenje modela s ovim i s bilo kojim od drugih zadataka nije pokazalo signifikantno poboljšanje performanci za rezultate prvog zadatka. Pretpostavlja se da je razlog tome nedovoljna sličnost između zadataka i činjenica da je skup podataka za binarnu klasifikaciju sarkazma relativno jednostavan i već dobro optimiziran za ovaj specifičan zadatak. Dodavanje drugih zadataka može uvesti dodatnu složenost koja ometa fokus modela na osnovni zadatak. Na primjer, kombiniranje binarne klasifikacije s prepoznavanjem sarkastične inačice rečenice (Task3) može unijeti suvišne informacije koje nisu korisne za razlikovanje sarkazma od nesarkazma.

Zadatak 2

Zadatak višeznačne klasifikacije potkategorija sarkazma (Task2) pokazao je značajan negativan utjecaj kada se u modelu višezadaćnog učenja kombinirao s trećim podzadatkom (MT23-2). Ovo može ukazivati na to da je zadatak prepoznavanja sarkastičnih inačica rečenica previše različit od zadatka višezadaćne klasifikacije potkategorija sarkazma. Ta razlika može dovesti do ometanja učenja modela, čime se smanjuje njegova sposobnost točne klasifikacije potkategorija sarkazma. S druge strane, kombinacija prvog i drugog podzadatka (MT12-2) nije pokazala značajnu razliku u performansama. Ovaj rezultat može sugerirati da binarna klasifikacija sarkazma i višezadaćna klasifi-

kacija potkategorija dijele određene zajedničke karakteristike koje omogućuju uspješno zajedničko učenje bez negativnog utjecaja na performance modela.

Zadatak 3

Za zadatak prepoznavanja sarkastične inačice rečenice (Task3), višezadačno učenje s prvim podzadatom (MT13-3) pokazalo je značajan negativan utjecaj na performance modela. Ovo može sugerirati da kombinacija ovog zadatka s binarnom klasifikacijom sarkazma nije korisna te da različite prirode ovih zadataka ometaju uspješno zajedničko učenje. Kombinacija drugog i trećeg podzadatka (MT23-3) također nije rezultirala poboljšanjem performanci, što može ukazivati na to da su zadaci prepoznavanja sarkastične inačice rečenice i višezadačne klasifikacije potkategorija previše različiti za učinkovito višezadačno učenje. Ovi rezultati sugeriraju da, unatoč potencijalnim prednostima višezadačnog učenja, određeni zadaci zahtijevaju treniranje sa zadacima slične prirode kako bi njihov zajednički trening rezultirao poboljšanjem u samim performansama na individualnim zadacima.

Ukupno gledano, višezadačno učenje nije donijelo značajna poboljšanja i u nekoliko slučajeva je čak pogoršalo rezultate. Razlog tome može biti nedovoljna sličnost zadataka, što implicira da zajedničko učenje nije uvijek korisno i može dovesti do negativnog prenošenja znanja iz zadatka u zadatak. Preporučuje se daljnje istraživanje s različitim kombinacijama zadataka i većim skupovima podataka kako bi se bolje razumjeli uvjeti pod kojima višezadačno učenje može biti korisno.

8. Zaključak

Razumijevanje sarkazma u tekstu predstavlja velik izazov za modele strojnog učenja i prirodne obrade jezika. Sarkazam je oblik izražavanja gdje se stvarno značenje rečenice razlikuje od doslovnog značenja, što otežava automatsko prepoznavanje sarkazma čak i za ljude. U eri digitalnog sadržaja i društvenih mreža, automatsko detektiranje sarkazma može poboljšati performance aplikacija poput analiza sentimenta, sustava za preporuke i chatbotova.

Ovaj rad istražuje sposobnost različitih modela strojnog učenja u klasifikaciji sarkazma kroz tri zadatka: binarnu klasifikaciju sarkastičnih rečenica, višezadaćnu klasifikaciju potkategorija sarkazma i prepoznavanje sarkastične inačice rečenice. Korišteni su modeli temeljeni na stroju potpornih vektora, modeli BERT te višezadaćno učenje na modelu BERT. Rezultati su pokazali da višezadaćno učenje nije značajno poboljšalo performance modela, te je u nekim slučajevima je čak pogoršalo rezultate, što može biti posljedica razlika u prirodi zadataka i ograničenog skupa za treniranje, što implicira da zadaci koji su u svojoj srži previše različiti mogu loše međusobno prenositi znanja pri treniranju.

Preporučuje se korištenje većih i raznolikijih skupova za treniranje te isprobavanje drugih arhitektura modela. Iako višezadaćno učenje nije donijelo očekivana poboljšanja, istraživanje je pružilo vrijedne uvide u izazove i mogućnosti automatske detekcije sarkazma, što može unaprijediti sposobnost modela za analizu digitalnog sadržaja i društvenih mreža.

Literatura

- [1] SemEval, “Semeval 2022 task 6: isarcasmeval”, accessed: 2024-06-05. [Mrežno]. Adresa: <https://sites.google.com/view/semEval2022-isarcasmeval>
- [2] A. S. P. B. S. K. Aparna K Ajayan, Krishna Mohanan, “Semeval-2022 task 6: isarcasmeval”, u *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022. [Mrežno]. Adresa: <https://aclanthology.org/2022.semeval-1.115/>
- [3] J. Brownlee, “Smote for imbalanced classification with python”, 2021., accessed: 2021-03-17. [Mrežno]. Adresa: <https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/>
- [4] S. Huang, E. A. Chi, i N. A. Chi, “Isd at semeval-2022 task 6: Sarcasm detection using lightweight models”, u *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics, 2022., str. 919–922, accessed: 2024-06-05. [Mrežno]. Adresa: <https://aclanthology.org/2022.semeval-1.129.pdf>
- [5] S. W. W. M. Ibrahim Abu Farha, Silviu Vlad Oprea, “Semeval-2022 task 6: isarcasmeval”, u *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022. [Mrežno]. Adresa: <https://aclanthology.org/2022.semeval-1.111/>
- [6] J. Devlin, M.-W. Chang, K. Lee, i K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding”, *arXiv preprint arXiv:1810.04805*, 2018. [Mrežno]. Adresa: <https://arxiv.org/pdf/1810.04805>

- [7] T. D. Diksha Krishnan, Jerin Mahibha C, “Semeval-2022 task 6: isarcasmeval”, u *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022. [Mrežno]. Adresa: <https://aclanthology.org/2022.semeval-1.114/>
- [8] A. S. S. M. R. M. T. T. Ramdhanush V, Rajalakshmi Sivanaiah, “Semeval-2022 task 6: isarcasmeval”, u *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022. [Mrežno]. Adresa: <https://aclanthology.org/2022.semeval-1.118/>
- [9] Y. Han, Y. Chai, S. Wang, Y. Sun, H. Huang, G. Chen, Y. Xu, i Y. Yang, “X-pudu at semeval-2022 task 6: Multilingual learning for english and arabic sarcasm detection”, u *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Kyiv, Ukraine (Virtual): Association for Computational Linguistics, July 2022., str. 999–1004. [Mrežno]. Adresa: <https://aclanthology.org/2022.semeval-1.140.pdf>
- [10] C. Dictionary, “Sarcasm”, <https://dictionary.cambridge.org/dictionary/english/sarcasm>, accessed: 2024-06-05.
- [11] Spiceworks, “What is support vector machine?” <https://www.spiceworks.com/tech/big-data/articles/what-is-support-vector-machine/>, accessed: 2024-06-05.
- [12] J. Šnajder, “Linearni diskriminativni modeli”, https://www.fer.unizg.hr/_download/repository/SU1-2022-P05-LinearniDiskriminativniModeli.pdf, 2022., accessed: 2024-06-05.
- [13] S. Abuse, “Understanding svm hyperparameters”, <https://stackabuse.com/understanding-svm-hyperparameters/>, accessed: 2024-06-06.
- [14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, i I. Polosukhin, “Attention is all you need”, *arXiv preprint arXiv:1706.03762*, 2017., provided proper attribution is provided, Google hereby grants permission to reproduce the tables and figures in this paper solely for use in journalistic or scholarly works. [Mrežno]. Adresa: <https://arxiv.org/pdf/1706.03762>

- [15] P. Belagatti, “Understanding the softmax activation function: A comprehensive guide”, 2024., accessed: 2024-06-05. [Mrežno]. Adresa: <https://www.singlestore.com/blog/a-guide-to-softmax-activation-function/>
- [16] B. Trevett, “Transformers encoder-decoder”, <https://kikaben.com/transformers-encoder-decoder/>, accessed: 2024-06-06.
- [17] DataCamp, “How transformers work”, <https://www.datacamp.com/tutorial/how-transformers-work>, accessed: 2024-06-06.
- [18] M. Hoque, “A comprehensive overview of transformer-based models: Encoders, decoders, and more”, <https://medium.com/@minh.hoque/a-comprehensive-overview-of-transformer-based-models-encoders-decoders-and-more-e9bc0644a4e5>, accessed: 2024-06-06.
- [19] H2O.ai, “Classify token”, <https://h2o.ai/wiki/classify-token/>, accessed: 2024-06-06.
- [20] —, “Separate token”, <https://h2o.ai/wiki/separate-token/>, accessed: 2024-06-06.
- [21] GeeksforGeeks, “Introduction to multi-task learning(mtl) for deep learning”, <https://www.geeksforgeeks.org/introduction-to-multi-task-learningmtl-for-deep-learning/>, accessed: 2024-06-05.
- [22] S. Wakchoure, “Multi-task learning and transfer learning”, <https://medium.com/@wakchouresarthak/multi-task-learning-and-transfer-learning-8da2bc859363>, accessed: 2024-06-05.
- [23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, i E. Duchesnay, “Scikit-learn: Machine learning in python”, 2011., accessed: 2024-06-05. [Mrežno]. Adresa: <https://scikit-learn.org/stable/>
- [24] H. Face, “Transformers: State-of-the-art natural language processing”, 2024., accessed: 2024-06-05. [Mrežno]. Adresa: <https://huggingface.co/docs/transformers/en/index>

- [25] J. Brownlee, “Early stopping to avoid overtraining neural network models”, <https://machinelearningmastery.com/early-stopping-to-avoid-overtraining-neural-network-models/>, accessed: 2024-06-05.
- [26] GeeksforGeeks, “Regularization by early stopping”, <https://www.geeksforgeeks.org/regularization-by-early-stopping/>, accessed: 2024-06-05.
- [27] BuiltIn, “Sigmoid activation function”, <https://builtin.com/machine-learning/sigmoid-activation-function>, accessed: 2024-06-05.
- [28] Masoud, “Understanding micro, macro, and weighted averages for scikit-learn metrics in multi-class classification with example”, <http://iamirmasoud.com/2022/06/19/understanding-micro-macro-and-weighted-averages-for-scikit-learn-metrics-in-multi-class-classification-with-example/>, accessed: 2024-06-05.
- [29] J. Šnajder, *Vrednovanje Modela*, 2022., strojno učenje 1, UNIZG FER, ak. god. 2022./2023., dostupno na: https://www.fer.unizg.hr/_download/repository/SU1-2022-P21-VrednovanjeModela.pdf.
- [30] Investopedia, “T test”, <https://www.investopedia.com/terms/t/t-test.asp>, 2023., accessed: 2023-06-05.

Sažetak

Primjena i vrednovanje metoda za detekciju sarkazma u engleskome jeziku

Josip Pavlić

Ovaj završni rad bavi se primjenom i vrednovanjem metoda za detekciju sarkazma u engleskom jeziku. Cilj je bio ispitati učinkovitost različitih modela strojnog učenja, uključujući stroj potpornih vektora (SVM), BERT i višezadaćno učenje, na tri podzadatka: identifikacija sarkazma, određivanje podkategorija sarkastičnih tekstova i prepoznavanje sarkazma u rečenicama i njihovim nesarkastičnim inačicama. Rezultati pokazuju da modeli BERT daju bolje rezultate od SVM-a, dok višezadaćno učenje nije pokazalo poboljšanja zbog velike razlike u prirodi podzadataka i ograničenosti skupa podataka.

Ključne riječi: obrada prirodnog jezika; detekcija sarkazma; transformatori; višezadaćno učenje; strojno učenje

Abstract

Application and evaluation of methods for detecting sarcasm in the English language

Josip Pavlić

This thesis explores the application and evaluation of methods for detecting sarcasm in the English language. The aim was to examine the effectiveness of various machine learning models, including Support Vector Machines (SVM), BERT, and multitask learning, on three subtasks: identifying sarcasm, determining subcategories of sarcastic texts, and recognizing sarcasm in sentences and their non-sarcastic paraphrases. The results show that BERT models perform better than SVM, while multitask learning did not show improvements due to the significant difference in the nature of the subtasks and the limitations of the dataset.

Keywords: natural language processing; sarcasm detection; transformers; multitask learning; machine learning