FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Računarstvo usluga i analiza podataka

SEMINARSKI RAD

Prepoznavanje katastrofe iz teksta

Tomislav Kušević

Dario Lucić

Petar Nenadić

Domagoj Voćanec

Osijek, 2021.

SADRŽAJ

1. UVOD

U današnje doba, količina podataka koja se generira u svim sferama života je ogromna. S porastom količine podataka pred ljude se stavljaju nova pitanja o kojima do sada nisu morali razmišljati. Pitanja kao jesu li ti podaci bitni za mene, kako ih sažeti i drugo. Pojavom tehnologija i sustava za analizu velikih količina podataka, uspjeli smo riješiti probleme kako ih predočiti u smislenom obliku, te se sada postavlja pitanje kako iskoristiti te podatke za svoje potrebe. Jedan od načina kako se ti podaci koriste i s kojim će se baviti ovaj projekt je analiza podataka na društvenim mrežama. Zadatak ovog projekta je analizirati poruke koje korisnici postavljaju na društvenim mrežama te ih prikazati pomoću web servisa izrađenog u Azure Machine Learning Studiju. Nakon analize potrebno je korisniku tekstualno i grafički prikazati rezultat, te omogućiti i neke dodatne mogućnosti.

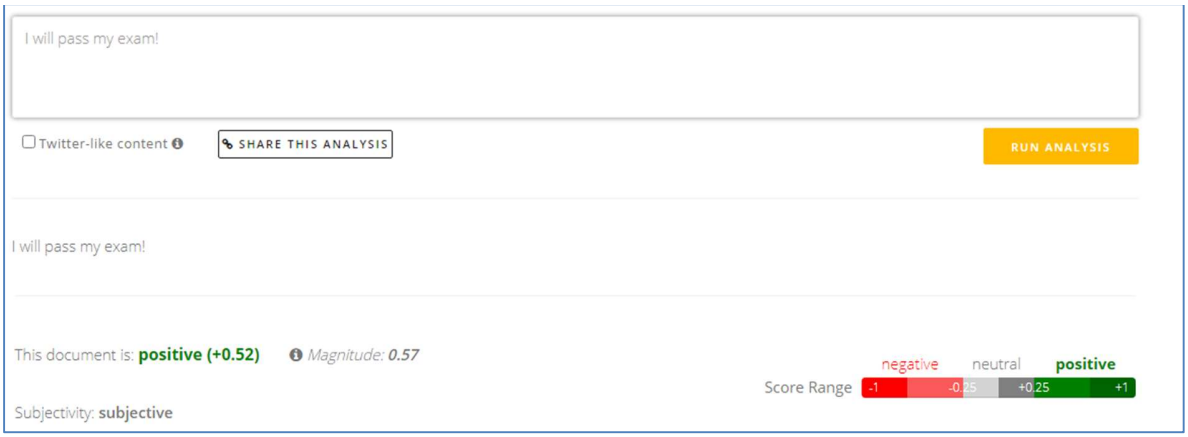
2. OPIS PROBLEMA

Tema ovog projekta je klasificirati poruke na društvenim medijima po proizvoljnom kriteriju. Potrebno je razraditi način klasifikacije, kako izraditi model i te koje algoritme strojnog učenja koristiti. Također, potrebno je osmisliti aplikaciju putem koje će korisnik komunicirati s izrađenim web servisom, dali izraditi web rješenje, mobilno rješenje ili desktop aplikaciju. Glavni zahtjev koji je trebalo ispuniti je da se korisniku omogući analiza poruke te da kao odgovor dobije u koju se kategoriju analizirana poruka klasificira. Nakon što se odluči kojim putem će se krenuti, potrebno je najprije odabrati prigodan model strojnog učenja te ga istrenirati i testirati. Kako bi rješenje ovog projekta bilo što točnije nije dovoljno isprobati samo jedan algoritam i jedne postavke, već treba obaviti više raznih treninga s raznim postavkama dok se ne dobije što precizniji model. Nakon što se provedu sva testiranja i objavi web servis, potrebno je izraditi korisničko sučelje. Treba razmisliti koje sve mogućnosti korisnik mora imati. U ovom slučaju, glavna mogućnost je unos teksta, analiza te prikaz rezultata analize.

Pri istraživanju teme pronađena su neka rješenja koja su poslužila kao inspiracija pri izradi. Sva ta rješenja su realizirana kao web stranice, dok je ovaj projekt realiziran kao mobilna aplikacija.

Ispod su navedeni neki primjeri s kratkim opisima:

1. Free sentiment analysis demo

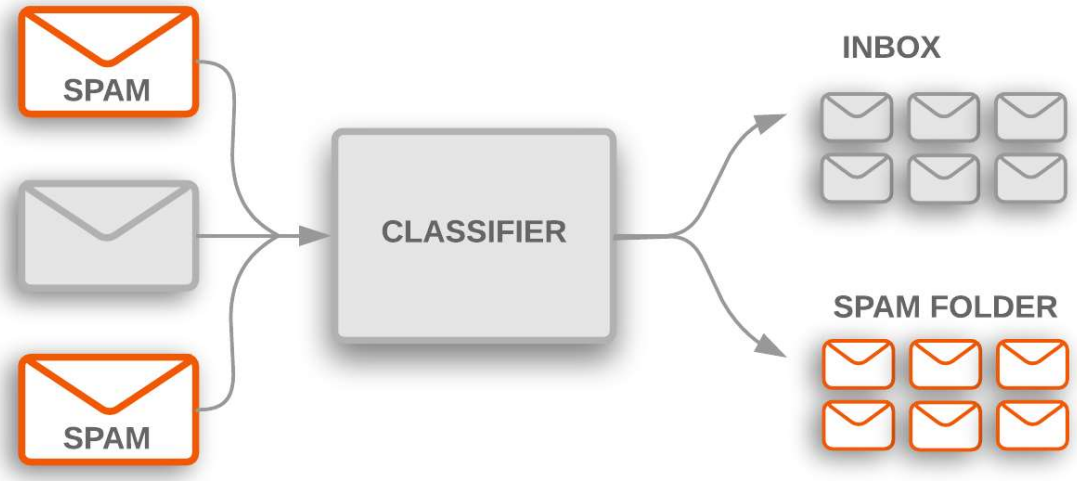


Slika 2.1. Free sentiment analysis demo

Jednostavna web stranica koja na temelju upisanog teksta radi klasifikaciju na negativno, pozitivno i neutralno. Ova stranica je bila inspiracija za odabir kriterija klasifikacije.

Stranici je moguće pristupiti na linku: https://text2data.com/Demo.

2. Email spam classificator



Slika 2.2. Email spam classificator

Klasifikator koji se koristi u brojnim email poslužiteljima kako bi filtrirao mailove koji predstavljaju opasnost ili ih se smatra kao "spam" poruka.

2.1. KORIŠTENI PODACI

Podaci koji su korišteni u ovom projektu su podaci sa stranice *data.world*. Stranica koristi Twitter Search API oslanjajući se na ključne riječi. Prilikom skupljanja poruka, koriste se riječi koje označavaju neku katastrofu te se klasificiraju prema tome koliko ljudi na društvenim mrežama smatra da je ta informacija točna.

Detaljnije informacije mogu se pronaći na službenoj stranici na kojoj je skup podataka izrađen (https://data.world/crowdflower/disasters-on-social-media).

Na službenoj stranici se nalazi skup podataka u .CSV formatu, sa otprilike 10,000 poruka raspoređenih u 13 kategorija. Te kategorije su:

- unit_id – id poruke
- golden – boolean vrijednost da li je informacija korisna
- unit_state – trenutno stanje poruke (aktivno događanje ili zatvorena tema)
- trusted_judgments – broj korisnika koji su potvrdili poruku
- last_judgment_at – vrijeme posljednje potvrde
- choose_one – odabir između važne i nevažne informacije
- choose_one_confidence – vjerojatnost točnosti informacije
- choose_one_gold – odabir koliko je informacija važna
- keyword – ključna riječ poruke
- location – lokacija događaja
- text – tekst sa društvene mreže koji uključuje ključnu riječ
- tweetid – id tweet-a
- userid – id korisnika

socialmedia-disaster-tweets-DFE.csv

	0	1	2	3	4	5	6	7	8
0	_unit_id	_golden	_unit_state	_trusted_judgments	_last_judgment_at	choose_one	choose_one:confidence	choose_one_gold	keyword
1	778243823	TRUE	golden	156		Relevant	1	Relevant	
2	778243824	TRUE	golden	152		Relevant	1	Relevant	
3	778243825	TRUE	golden	137		Relevant	1	Relevant	
4	778243826	TRUE	golden	136		Relevant	0.9603	Relevant	
5	778243827	TRUE	golden	138		Relevant	1	Relevant	
6	778243828	TRUE	golden	140		Relevant	1	Relevant	
7	778243831	TRUE	golden	142		Relevant	1	Relevant	
8	778243832	TRUE	golden	151		Relevant	1	Relevant	
9	778243833	TRUE	golden	143		Relevant	1	Relevant	
10	778243834	TRUE	golden	136		Relevant	0.9606	Relevant Can't Decide	
11	778243835	TRUE	golden	152		Relevant	1	Relevant	
12	778243836	TRUE	golden	157		Relevant	1	Relevant	
13	778243837	TRUE	golden	143		Relevant	1	Relevant	
14	778243838	TRUE	golden	140		Relevant	0.9607	Relevant Can't Decide	
15	778243839	TRUE	golden	136		Relevant	0.9215	Relevant	
16	778243840	TRUE	golden	147		Relevant	0.7177	Relevant Can't Decide	
17	778243841	TRUE	golden	147		Relevant	0.9603	Relevant	
18	778243842	TRUE	golden	132		Relevant	1	Relevant	
19	778243843	TRUE	golden	144		Relevant	1	Relevant	
20	778243844	TRUE	golden	147		Relevant	0.6761	Relevant Can't Decide	
21	778243845	TRUE	golden	136		Relevant	1	Relevant	
22	778243846	TRUE	golden	151		Not Relevant	0.9612	Not Relevant	
23	778243847	TRUE	golden	124		Not Relevant	1	Not Relevant	
24	778243848	TRUE	golden	140		Not Relevant	1	Not Relevant	
25	778243849	TRUE	golden	141		Not Relevant	1	Not Relevant	
26	778243851	TRUE	golden	142		Not Relevant	1	Not Relevant	
27	778243852	TRUE	golden	138		Not Relevant	1	Not Relevant	

Slika 2.1.1. Neobrađeni podaci

U ovom projektu bazirat ćemo se na dva stupca:

- keyword
- text

Zbog korištenja samo dvaju stupaca bilo je potrebno obraditi podatke pomoću nekog alata koji omogućava manipulaciju datoteke koja ima .CSV format. Korišten je Modern CSV, alat otvorenog koda dostupan na https://www.moderncsv.com/.

Nakon obrade podataka dobivena je .CSV datoteka sa samo 2 stupca koji su nam potrebni za projekt.

socialmedia-disaster-tweets-DFE.csv * ×

	0	1
110	accident	RT @SleepJunkies: Sleeping pills double your risk of a car accident http://t.co/7s9Nm1fiCT
111	accident	'By accident' they knew what was gon happen https://t.co/Ysxun5vCeh
112	accident	@Traffic_SouthE @roadpol_east Accident on A27 near Lewes is it Kingston Roundabout rather than A283
113	accident	Traffic accident N CABRILLO HWY/MAGELLAN AV MIR (08/06/15 11:03:58)
114	accident	I-77 Mile Marker 31 to 40 South Mooresville Iredell Vehicle Accident Congestion at 8/6 1:18 PM
115	accident	the pastor was not in the scene of the accident.....who was the owner of the range rover ?
116	accident	@sakuma_en If you pretend to feel a certain way the feeling can become genuine all by accident. -Hei (Darker than Black) #manga #anime
117	accident	For Legal and Medical Referral Service @1800_Injured Call us at: 1-800-465-87332 #accident #slipandfall #dogbite
118	accident	me: 'why is that?'...
119	accident	I was in a horrible car accident this past Sunday. I'm finally able to get around. Thank you GOD??
120	accident	Can wait to see how pissed Donnie is when I tell him I was in ANOTHER accident??
121	accident	#TruckCrash Overturns On #FortWorth Interstate http://t.co/Rs22LJ4qFp Click here if you've been in a crash>http://t.co/Ld0unIYw4k
122	accident	Accident in #Ashville on US 23 SB before SR 752 #traffic http://t.co/hylMo0WgFI
123	accident	There's a construction guy working on the Disney store and he has huge gauges in his ears ?? ...that is a bloody accident waiting to happen
124	accident	@RobynJillian @WISDOMTEETHS I feel like I'm going to do it on accident. Teesha is gonna come out??
125	accident	On the #M42 northbound between junctions J3 and J3A there are currently delays of 10 mins due to an accident c... http://t.co/LwI3prBa31
126	accident	@DaveOshry @Soembie So if I say that I met her by accident this week- would you be super jelly Dave? :p
127	accident	Carolina accident: Motorcyclist Dies in I-540 Crash With Car That Crossed Median: A motorcycle rider traveling... http://t.co/p18lzRlmy6
128	accident	ACCIDENT - HIT AND RUN - COLD at 500 BLOCK OF SE VISTA TER GRESHAM OR [Gresham Police #PG15000044357] 10:35 #pdx911
129	accident	FYI CAD:FYI: ;ACCIDENT PROPERTY DAMAGE;NHS;999 PINER RD/HORNDALE DR
130	accident	RT nAAYf: First accident in years. Turning onto Chandanee Magu from near MMA. Taxi rammed into me while I was halfway turned. Everyone conf��_
131	accident	Accident left lane blocked in #Manchester on Rt 293 NB before Eddy Rd stop and go traffic back to NH-3A delay of 4 mins #traffic
132	accident	;ACCIDENT PROPERTY DAMAGE; PINER RD/HORNDALE DR
133	accident	???? it was an accident http://t.co/Oia5fxi4gM
134	accident	FYI CAD:FYI: ;ACCIDENT PROPERTY DAMAGE;WPD;1600 S 17TH ST
135	accident	8/6/2015@2:09 PM: TRAFFIC ACCIDENT NO INJURY at 2781 WILLIS FOREMAN RD http://t.co/VCkIT6EDEv
136	accident	Aashiqui Actress Anu Aggarwal On Her Near-Fatal Accident http://t.co/6Otfp31LqW
137	accident	Suffield Alberta Accident https://t.co/bPTmlF4P10

Slika 2.1.2. Obrađeni podaci

2.2. KORIŠTENI POSTUPCI STROJNOG UČENJA

Kao što je navedeno u prethodnom tekstu, ovaj projekt se temelji na klasifikaciji. Osim podataka koji su prethodno navedeni, također je potrebno koristiti odabrani algoritam strojnog učenja, metode za odvajanje podataka na trening i test, uspoređivanje stvarnih i predviđenih vrijednosti i ocjenjivanje modela. Podaci se odvajaju na trening i test kako bi model, na osnovu zadanih i obrađenih trening podataka, mogao klasificirati nepoznate, odnosno nove varijable u odgovarajuće kategorije.

Odabir algoritma je jako bitan kod izrade modela iz razloga što ne rade svi algoritmi jednako dobro sa svim tipovima podataka, odnosno neki rade bolje sa tekstom, a neki s brojevima. Na primjer, kod projekata gdje se moraju predvidjeti neke vrijednosti, ne mogu se koristiti klasifikacijski algoritmi, već se moraju koristiti algoritmi za regresiju. U ovom slučaju,

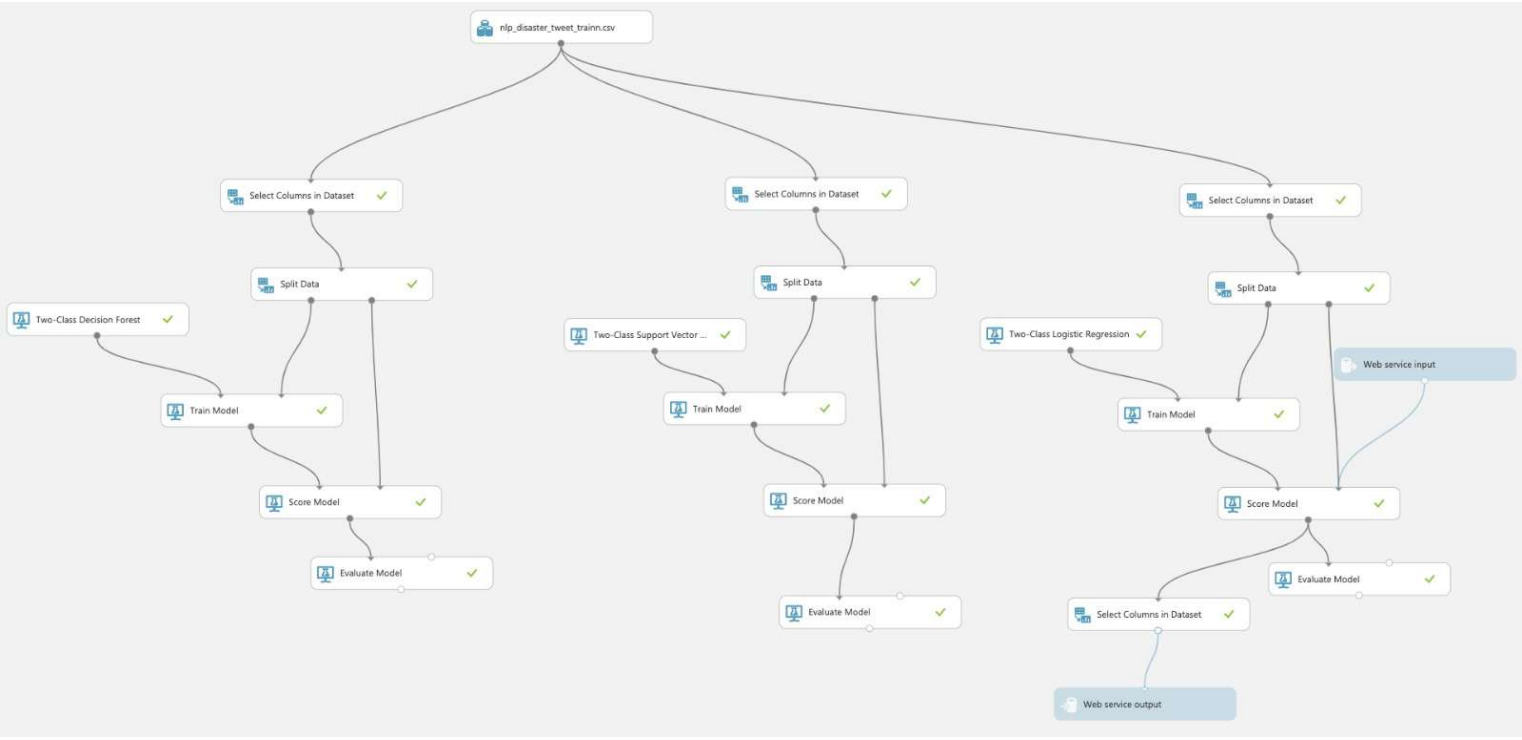
radi se sa samo dvije varijable, tako da je najoptimalnije koristiti neki od algoritama za klasifikaciju dviju klasa, poput *„Two-Class Support Vector Machine“*, *„Two-Class Neural Network“* ili neki drugi. Prednost ovakvih algoritama je što klasifikacijski modeli koji rade samo s dvije varijable imaju manji mogući broj izlaznih vrijednosti te zbog toga imaju jako veliku učinkovitost.

U ovom projektu je napravljena usporedba između 3 različita klasifikacijska algoritma, te je odabran najoptimalniji, a taj postupak će biti detaljnije opisan u poglavlju *„Model strojnog učenja“*.

3. OPIS PROGRAMSKOG RJEŠENJA

3.1. MODEL STROJNOG UČENJA

Na slici 3.1.1. prikazan je kompletan model strojnog učenja za prepoznavanje katastrofe iz teksta.



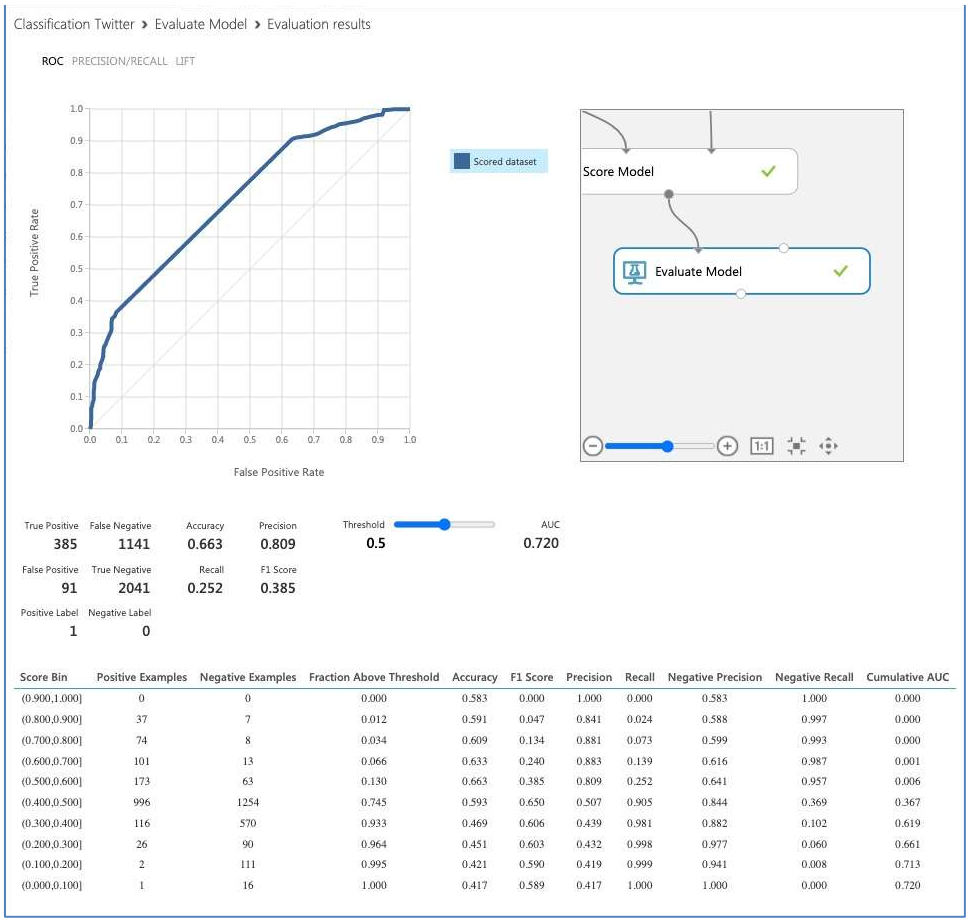
Slika 3.1.1. Kompletan model strojnog učenja

Prije odabira algoritma potrebno je podatke podijeliti u određenom omjeru u dvije skupine. Prva skupina predstavlja podatke za treniranje/učenje, dok druga skupina podataka služi za testiranje naučenog algoritma. Podjela je vršena uz pomoć djelitelja podataka (eng. *Split Data*) u Azure Studiju.

Testirana su tri algoritma te je odabran najučinkovitiji, a to su:

- Two-Class Decision Forrest – Ovaj algoritam je cjelovita metoda učenja namijenjena klasifikacijskim zadacima. Metode sastavljanja temelje se na općem načelu da, umjesto da se oslanjate na jedan model, možete postići bolje rezultate i općenitiji model stvaranjem više povezanih modela i njihovim kombiniranjem na neki način. Općenito, modeli ansambla pružaju bolju pokrivenost i točnost od stabala pojedinačnih odluka. Postoji mnogo načina za stvaranje pojedinačnih modela i njihovo kombiniranje u cjelinu.

Ova posebna provedba šuma odlučivanja radi izgradnjom više stabala odlučivanja, a zatim glasovanjem o najpopularnijoj izlaznoj klasi. Glasanje je jedna od poznatijih metoda za generiranje rezultata u modelu ansambla.

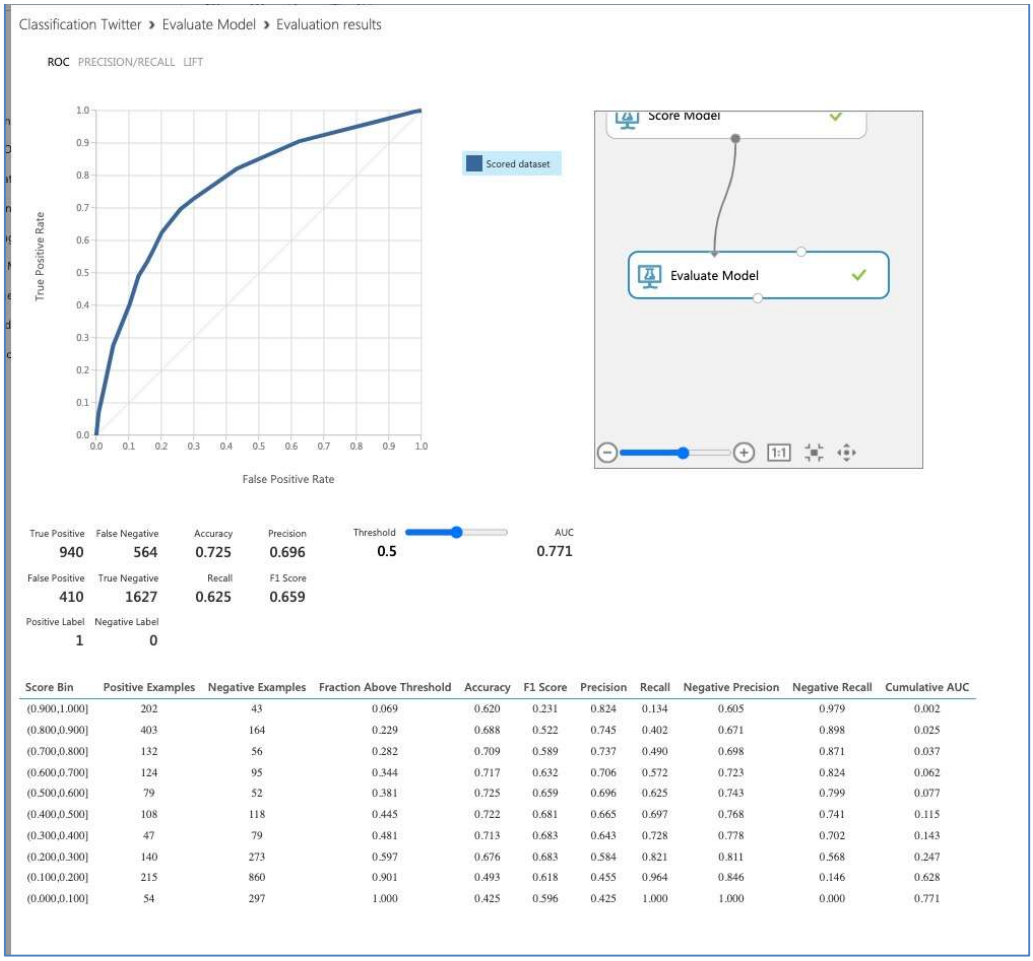


Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	0	0	0.000	0.583	0.000	1.000	0.000	0.583	1.000	0.000
(0.800,0.900]	37	7	0.012	0.591	0.047	0.841	0.024	0.588	0.997	0.000
(0.700,0.800]	74	8	0.034	0.609	0.134	0.881	0.073	0.599	0.993	0.000
(0.600,0.700]	101	13	0.066	0.633	0.240	0.883	0.139	0.616	0.987	0.001
(0.500,0.600]	173	63	0.130	0.663	0.385	0.809	0.252	0.641	0.957	0.006
(0.400,0.500]	996	1254	0.745	0.593	0.650	0.507	0.905	0.844	0.369	0.367
(0.300,0.400]	116	570	0.933	0.469	0.606	0.439	0.981	0.882	0.102	0.619
(0.200,0.300]	26	90	0.964	0.451	0.603	0.432	0.998	0.977	0.060	0.661
(0.100,0.200]	2	111	0.995	0.421	0.590	0.419	0.999	0.941	0.008	0.713
(0.000,0.100]	1	16	1.000	0.417	0.589	0.417	1.000	1.000	0.000	0.720

Slika 3.1.2. Rezultati Two-Class Decision Forrest algoritma

- Two-Class Support Vector Machine - jedan su od najranijih algoritama strojnog učenja, a SVM modeli korišteni su u mnogim aplikacijama, od dohvaćanja informacija do klasifikacije teksta i slike. SVM se mogu koristiti i za klasifikacijske i za regresijske zadatke. Ovaj SVM model je nadzirani model učenja koji zahtijeva označene podatke. U procesu obuke, algoritam analizira ulazne podatke i prepoznaje obrasce u višedimenzionalnom prostoru obilježja koji se naziva hiperravnina. Svi primjeri unosa su predstavljeni kao točke u ovom prostoru i mapirani su u izlazne kategorije na takav način da su kategorije podijeljene sa što širim i čistim prazninama. Za predviđanje, SVM

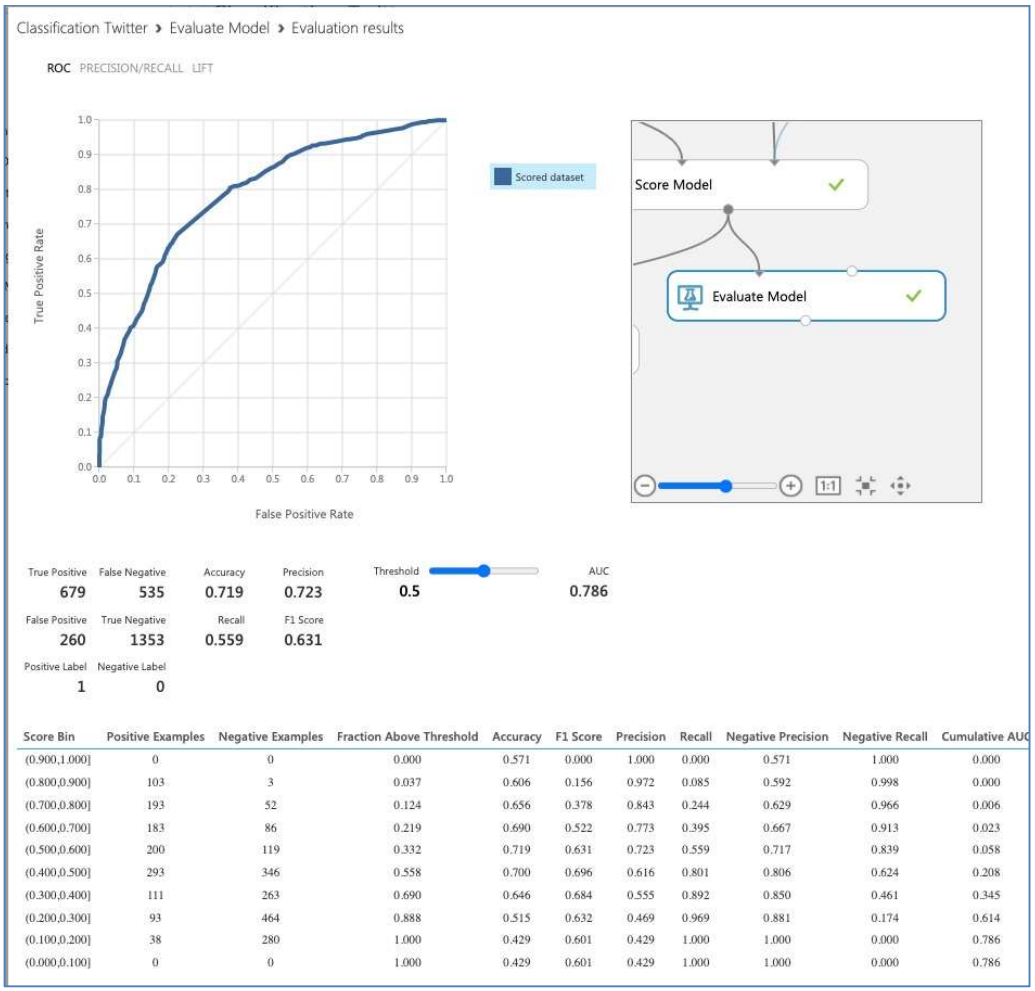
algoritam dodjeljuje nove primjere u jednu ili drugu kategoriju, preslikavajući ih u taj isti prostor.



Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	202	43	0.069	0.620	0.231	0.824	0.134	0.605	0.979	0.002
(0.800,0.900]	403	164	0.229	0.688	0.522	0.745	0.402	0.671	0.898	0.025
(0.700,0.800]	132	56	0.282	0.709	0.589	0.737	0.490	0.698	0.871	0.037
(0.600,0.700]	124	95	0.344	0.717	0.632	0.706	0.572	0.723	0.824	0.062
(0.500,0.600]	79	52	0.381	0.725	0.659	0.696	0.625	0.743	0.799	0.077
(0.400,0.500]	108	118	0.445	0.722	0.681	0.665	0.697	0.768	0.741	0.115
(0.300,0.400]	47	79	0.481	0.713	0.683	0.643	0.728	0.778	0.702	0.143
(0.200,0.300]	140	273	0.597	0.676	0.683	0.584	0.821	0.811	0.568	0.247
(0.100,0.200]	215	860	0.901	0.493	0.618	0.455	0.964	0.846	0.146	0.628
(0.000,0.100]	54	297	1.000	0.425	0.596	0.425	1.000	1.000	0.000	0.771

Slika 3.1.3. Rezultati Two-Class Vector Machine algoritma

- Two-Class Logistic Regression.- poznata je metoda u statistici koja se koristi za predviđanje vjerojatnosti ishoda, a osobito je popularna za klasifikacijske zadatke. Algoritam predviđa vjerojatnost nastanka događaja uklapanjem podataka u logističku funkciju. U ovom modulu klasifikacijski algoritam optimiziran je za binarne varijable. Ako je potrebno klasificirati više ishoda, upotrijebite modul Višeklasična logistička regresija.



Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	0	0	0.000	0.571	0.000	1.000	0.000	0.571	1.000	0.000
(0.800,0.900]	103	3	0.037	0.606	0.156	0.972	0.085	0.592	0.998	0.000
(0.700,0.800]	193	52	0.124	0.656	0.378	0.843	0.244	0.629	0.966	0.006
(0.600,0.700]	183	86	0.219	0.690	0.522	0.773	0.395	0.667	0.913	0.023
(0.500,0.600]	200	119	0.332	0.719	0.631	0.723	0.559	0.717	0.839	0.058
(0.400,0.500]	293	346	0.558	0.700	0.696	0.616	0.801	0.806	0.624	0.208
(0.300,0.400]	111	263	0.690	0.646	0.684	0.555	0.892	0.850	0.461	0.345
(0.200,0.300]	93	464	0.888	0.515	0.632	0.469	0.969	0.881	0.174	0.614
(0.100,0.200]	38	280	1.000	0.429	0.601	0.429	1.000	1.000	0.000	0.786
(0.000,0.100]	0	0	1.000	0.429	0.601	0.429	1.000	1.000	0.000	0.786

Slika 3.1.4. Rezultati Two-Class Logistic Regression algoritma

Na tri prethodne slike prikazani su rezultati navedenih algoritama, koji su dobiveni opcijom „Evaluate Model“ gdje pišu informacije o preciznosti i točnosti koristeći zadane podatke. Za konačni model odabran je algoritam Two-Class Logistic Regression jer je on davao najtočnije rezultate.

3.2. NAČIN KORIŠTENJA API-JA

API korišten u izradi ovog projekta služi za komunikaciju između modela i korisničkog sučelja aplikacije. Potrebno je napraviti web servis i preuzeti REST API (engl. *Representational State Transfer Application Programming Interface*) ključ za korištenje modela u aplikaciji. API je skup pravila i funkcija koji omogućuju već navedenu komunikaciju koristeći REST arhitekturu. REST je zasnovan na HTTP (engl. *Hypertext Transfer Protocol*) protokolu koji koristi svoje metode za iniciranje željene akcije. Neke on najkorištenijih metoda su: GET, PUT, POST, DELETE itd. Zahtjev (engl. *request*) se šalje putem HTTP POST metode te ukoliko je on uspješan dobije se odgovor (engl. *response*) u obliku JSON (engl. *JavaScript Object Notation*) objekta koji sadrži sve potrebne podatke za provedbu željene akcije.

```
@Headers(
        "Accept: application/json",
        "Authorization: Bearer
472ATlkLfWUee0yaTp9EwCg1t+Gh1xyAODMIV6i6/c/P2f7j4h2R6xPRqk8VwyOFTCjRO
COdIYiKM1yk0HK2Pw==",
        "Content-Type: application/json"
    )

@POST("/workspaces/504782373e424b018d37767150b89756/services/c4a171b1
fc3841e5882522711c39ccbf/execute?api-version=2.0&details=true")
    fun getSummaryResponse(@Body body: RequestReq): Call<ResponseRes>
```

Kod 3.2.1. Ruta koja se poziva sa zaglavljima (engl. *header*)

U prethodnom dijelu koda prikaza je ruta od API-ja koja se poziva, te vidimo da se dobiva odgovor (engl. *response*) u JSON formatu. U Prikazu koda 2.2.2. prikazana je BASE_URL konstanta koja sadrži link od Azure servisa, a ispod konstante vidimo direktan poziv iz koda.

```
const val BASE_URL = https://ussouthcentral.services.azureml.net/
.
.
.
private fun getResults(text: String, key: String) {
        progress.visible()
        interactor.getResults(text, key, getSummaryCallback())
}
```

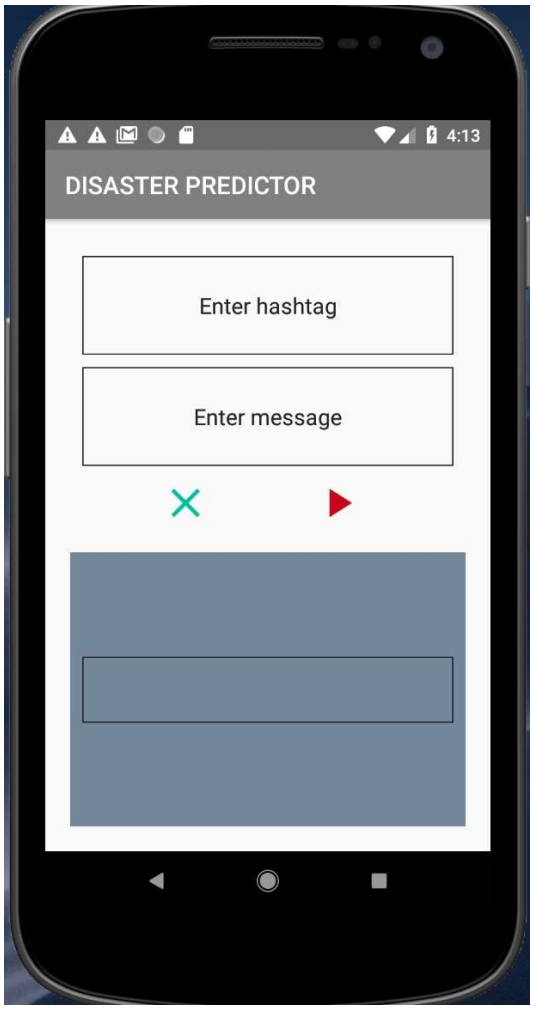
Kod 3.2.2. BASE_URL konstanta i poziv iz koda

Kod 3.2.3. Primjer odgovora (engl. *response*)

U prethodnom dijelu koda vidimo primjer odgovora od API-ja, koji je u JSON formatu. Svaki odgovor izgleda ovako, te sadrži predviđenu vrijednost koja nam je vraćena te se koristi u sučelju aplikacije prilikom unosa teksta.

3.3. KLIJENTSKA APLIKACIJA

Klijentska aplikacija je napravljena preko Android Studio programskog alata. U ovom poglavlju bit će objašnjeno izravno korištenje aplikacije, kako ona funkcionira i što se vraća prilikom unosa. Na sljedećoj slici prikazano je početno sučelje koje korisnik vidi prilikom otvaranja aplikacije.



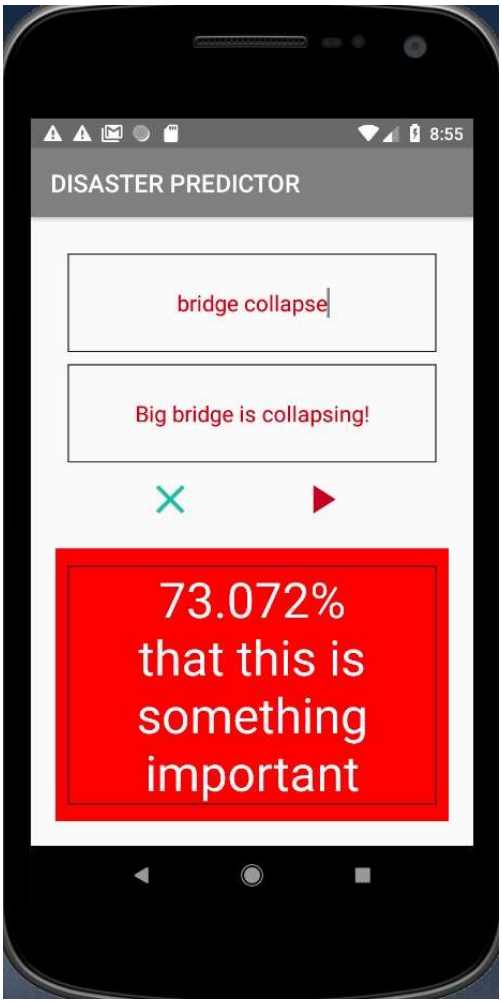
Slika 3.3.1. Prikaz korisničkog sučelja na početku

Aplikacija funkcionira na sljedeći način: postoje dva polja za tekst (engl. *textbox*) koja predstavljaju nekakvu virtualnu Twitter poruku. Prvo polje predstavlja ključnu riječ odnosno hashtag, dok drugo polje predstavlja objavu s Twittera. Aplikacija nam onda šalje povratnu vrijednost dali kombinacija oba polja sadrži neku od ključnih riječi koje predstavljaju katastrofu. Nakon unosa teksta u oba polja, dobijemo povratnu informaciju, odnosno postotak kolika je

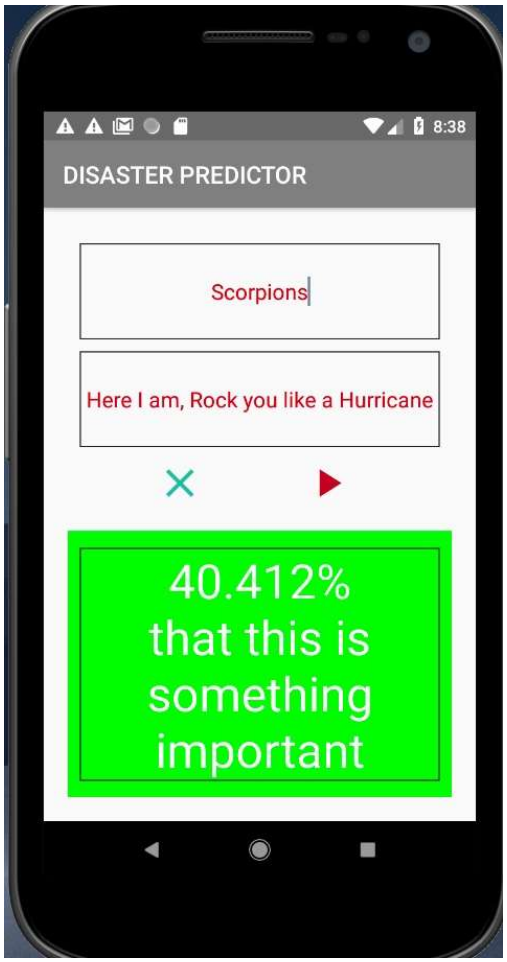
šansa da se radi o katastrofi. Ukoliko je šansa za opasnost preko 50%, dobiva se poruka označena crvenom bojom, dok je u suprotnom poruka označena zelenom. Postotak je moguće dobiti na osnovu istreniranih podataka te modela putem Azure-a koji nam šalje povratnu informaciju.

Na sljedeće dvije slike vidimo izgled aplikacije u dva slučaja, ovisno u unesenom tekstu:

- Kada je velika šansa da se radi o katastrofi
- Kada je veća šansa da se ipak o normalnoj Twitter objavi.



Slika 3.3.2. Izgled aplikacija kada je riječ o katastrofi



Slika 3.3.3. Izgled aplikacija kada je riječ o normalnoj objavi

4. ZAKLJUČAK

Zadani zadatak je uspješno obavljen i pomoću njega je moguće vrlo precizno klasificirati unesene ili dohvaćene poruke. Pri izradi ovog projekta proširena je i nadograđena podloga koja je stečena na laboratorijskim vježbama iz kolegija *Računarstvo usluga i analiza podataka*. Uz znanje koje je stečeno o izradi modela, prošireno je modelima koji nisu korišteni na laboratorijskim vježbama, te neke nove elemente koji su morali biti korišteni kako bi web servis bio što precizniji, kao što je izrada mobilne aplikacije, korištenje biblioteka za pristup i komunikaciju s društvenim mrežama i druge elemente.

Uspješno je izvedena sva funkcionalnost koja je zadana u zadatku, ali uvijek postoji mjesta za nadogradnju. Neke od mogućih nadogradnji bi bile, na primjer, mogućnost pretrage poruke po osobi ili lokaciji. Ta nadogradnja je i sada bila izvediva, ali bi se pojavila prepreka u aplikaciji jer je model istreniran na engleski jezik, odnosno, kada bi se upisala neka riječ, lokacija ili ime osobe čije poruke nisu na engleskom jeziku dobila bi se procjena koja ne bi bila točna.

Na kraju, kroz izradu modela uviđeno je da su analiza podataka i strojno učenje (engl. *Machine learning*) veoma korisne tehnologije i jasnije je zašto su toliko rasprostranjene u današnje vrijeme.

5. POVEZNICE I LITERATURA

Programskom je rješenju moguće pristupiti preko:

Programsko rješenje na GitHubu
ML model

1. https://www.redhat.com/en/topics/api/what-is-a-rest-api
2. https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/two-class-decision-forest
3. https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/two-class-support-vector-machine
4. https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/two-class-logistic-regression
5. https://text2data.com/Demo
6. https://data.world/crowdflower/disasters-on-social-media