

**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA**

Klasifikacija filmskog žanra u odnosu na opis

Kolegij: Raspoznavanje uzoraka i strojno učenje

Leon Šumanovac

Osijek, 2023.

SADRŽAJ

1. UVOD.....	3
2. ISTRAŽIVAČKA ANALIZA PODATAKA.....	4
2.1. Korištene biblioteke.....	5
2.2. Priprema podataka.....	6
2.3. Predikcija podataka.....	6
3. KORIŠTENI ALGORITMI – TEORIJSKI OPIS.....	7
2.1. Logistička regresija.....	7
2.2. LinearSVC.....	7
3.1.1. Matrica zabune.....	8
3. APLIKACIJA ZA PRIKAZ REZULTATA – STREAMLIT.....	9
4. ZAKLJUČAK.....	14

1. UVOD

Ovaj projekt će se baviti problemom klasifikacije filmskog žanra iz dobivenog opisa filma. Klasifikacija će biti odrađena pomoću dva algoritma, a to su linearna regresija i LinearSVC (Linear Support Vector Classifier).

Projekt nudi mogućnost unošenja vlastitog opisa filma, te aplikacija pomoću Streamlit-a koristi zadnji korišteni algoritam, te kroz njega „provlači“ uneseni opis i ispisuje rezultat.

Za ovaj projekt će se koristiti Python 3.9.

2. ISTRAŽIVAČKA ANALIZA PODATAKA

Istraživačka analiza podataka odnosi se na kritičan proces izvođenja početnih istraživanja podataka kako bi se otkrili obrasci, uočile anomalije, testirale hipoteze i provjerile pretpostavke uz pomoć zbirne statistike i grafičkih prikaza.

Dataset koji je korišten može se pronaći na ovome linku:

<https://www.kaggle.com/datasets/hijest/genre-classification-dataset-imdb>

Unutar njega zabilježeni su sljedeći podaci o filmovima: id, naslov (s godinom izdanja), žanr i opis.

Podaci su spremljeni u .txt file sa separatorom '!!!'.

```
1 ::: Oscar et la dame rose (2009) ::: drama ::: Listening in to a conver
2 ::: Cupid (1997) ::: thriller ::: A brother and sister with a past ince
3 ::: Young, Wild and Wonderful (1980) ::: adult ::: As the bus empties t
4 ::: The Secret Sin (1915) ::: drama ::: To help their unemployed father
5 ::: The Unrecovered (2007) ::: drama ::: The film's title refers not on
6 ::: Quality Control (2011) ::: documentary ::: Quality Control consists
7 ::: "Pink Slip" (2009) ::: comedy ::: In tough economic times Max and J
8 ::: One Step Away (1985) ::: crime ::: Ron Petrie (Keanu Reeves) is a t
9 ::: "Desperate Hours" (2016) ::: reality-tv ::: A sudden calamitous eve
10 ::: Spirits (2014/I) ::: horror ::: Four high school students embark o
11 ::: The Spirit World: Ghana (2016) ::: documentary ::: Tom Beacham exp
12 ::: In the Gloaming (1997) ::: drama ::: Danny, dying of Aids, returns
13 ::: Pink Ribbons: One Small Step (2009) ::: documentary ::: A sister's
14 ::: Interrabang (1969) ::: thriller ::: A photographer is sailing with
15 ::: The Glass Menagerie (1973) ::: drama ::: Amanda Wingfield dominate
16 ::: Night Call (2016) ::: drama ::: Simon's world is turned upside dow
17 ::: Babylon Vista (2001) ::: comedy ::: Frankie Reno was a child star
18 ::: "Wo Grafen schlafen - Eine Schlösser-Reise" (2014) ::: documentary
19 ::: "Roller Warriors" (2009) ::: sport ::: Modern roller derby began i
20 ::: Bird Idol (2010) ::: animation ::: The story revolves around a bir
21 ::: O Signo das Tetas (2016) ::: drama ::: The Road of Milk narrates i
22 ::: Söderpojkar (1941) ::: comedy ::: A gang of unemployed itinerant m
23 ::: Tunnel Vision (1976) ::: comedy ::: A committee investigating TV's
24 ::: Wedded Bliss? (2002) ::: drama ::: Wedded Bliss? explores the perc
25 ::: Cheongchun highway (1973) ::: action ::: Dong-woo is released from
26 ::: The Sandman (????/I) ::: fantasy ::: A wizard attempting to captur
```

Slika 2.1 Prikaz parametara dataseta

2.1. Korištene biblioteke

Unutar rada korištene su biblioteke:

1. Pandas
2. Numpy
3. Streamlit
4. Matplotlib
5. Scikit-learn
6. Pickle
7. Os

```
import pandas as pd
import pickle
import os
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix, accuracy_score
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.svm import LinearSVC
import streamlit as st
import matplotlib.pyplot as plt
```

Slika 2.1.1 Prikaz korištenih biblioteka

Nakon što su biblioteke uvezene u program, podatci se učitavaju iz dataseta train_data.txt koji se nalazi u folderu data. U varijablu se spremaju stupci pod nazivima Title, Genre i Description razdvojeni sa ':' kao što je to rečeno u prethodnom poglavlju. Podatci su spremljeni u varijablu tipa DataFrame.

Nakon uspješnog učitavanja podataka, brišemo godinu izdavanja iz naslova filma.

```
moviesData = pd.read_csv('data/train_data.txt', sep=':::', engine='python', names=['Title', 'Genre', 'Description'])
# moviesData = pd.read_csv('data/test_data_solution.txt', sep=':::', engine='python', names=['Title', 'Genre', 'Description'])

moviesData['Title'] = moviesData['Title'].replace({' \(.?\)\': ''}, regex=True)
```

Slika 2.1.2 Prikaz učitavanja podataka i brisanja godine iz naslova

2.2. Priprema podataka

Kako bi koristili podatke iz dataseta potrebno ih je obraditi i pripremiti kako bi ih model mogao koristiti zajedno.

Kako bi to postignuli kreirana je funkcija preprocessData koja vraća features i target. I features i target su tipa list.

```
def preprocessData():  
    target = list(moviesData['Genre'])  
    features = list(moviesData['Description'])  
  
    return features, target
```

Slika 2.2.1 Prikaz pripreme podataka za korištenje u modelu

2.3. Predikcija podataka

Nakon što su podatci pripremljeni, definirana je funkcija predict, u kojoj se poziva priprema podataka i obrada samih podataka.

Funkcija na početku pomoću biblioteke os briše stari dump ako on postoji od prethodnog poziva funkcije predict. Nakon toga poziva pripremu podataka i dijeli ih na trening podatke i na test podatke u omjeru 0.8 – 0.2.

```
def predict(model):  
    try:  
        os.remove('last_dump.pkl')  
    except:  
        print('Nothing to delete')  
  
    features, target = preprocessData()  
    features_train, features_test, target_train, target_test = train_test_split(features, target, test_size = 0.2)  
    vectorizer = CountVectorizer()  
    features_train_vectorized = vectorizer.fit_transform(features_train)  
  
    if model == 1:  
        classifier = LinearSVC()  
    if model == 2:  
        classifier = LogisticRegression()  
  
    classifier.fit(features_train_vectorized, target_train)  
    X_test_vectorized = vectorizer.transform(features_test)  
    y_pred = classifier.predict(X_test_vectorized)  
  
    with open('last_dump.pkl', 'wb') as f:  
        pickle.dump((classifier, vectorizer), f)  
  
    unique_targets = np.unique(target_test)  
  
    cm = confusion_matrix(target_test, y_pred)  
    ac = accuracy_score(target_test, y_pred)  
    st.write(pd.DataFrame(cm, columns=unique_targets, index=unique_targets))  
    st.write(ac)
```

Slika 2.3.1 Funkcija za predikciju

Nakon toga se bira model algoritma po kojim će se nastaviti vršiti funkcija. Model se bira preko primljenog parametra iz main-a. Nakon toga su podatci fitani koristeći model zadanog algoritma, te je odrađena predikcija nad testnim podacima. Dump podaci o modelu su spremljeni pomoću biblioteke Pickle, te se izvršava prikaz matrice zabune i točnosti na Streamlit stranici.

3. KORIŠTENI ALGORITMI – TEORIJSKI OPIS

Nakon prikazanih podataka ćemo ponešto reći o svakome algoritmu od kojega ćemo dobiti rezultate i prikazati točnost pojedinog algoritma

Algoritmi koji su korišteni:

1. LinearSVC
2. LogisticRegression

Koristili smo algoritme u svrhu izračuna podataka te smo dobivene vrijednosti stavili u matricu zabune kako bi dobili određenu točnost za pojedini algoritam.

2.1. Logistička regresija

Logistička regresija je model klasifikacije, a ne regresijski model unatoč svog naziva. Logistička regresija je jednostavna i učinkovitija metoda za probleme binarne i linearne klasifikacije. To je klasifikacijski model koji je vrlo lako realizirati i postiže vrlo dobre performanse s linearno odvojitim klasama. Ona je vrlo opsežno korišten algoritam za klasifikaciju u industriji. Logistički regresijski model, poput Adaline i perceptrona, statistička je metoda za binarnu klasifikaciju koja se može generalizirati na višeklasnu klasifikaciju. Scikit-learn ima vrlo optimiziranu verziju implementacije logističke regresije, koja podržava zadatak klasifikacije više klasa.

2.2. LinearSVC

LinearSVC je linearni algoritam modela za klasifikaciju temeljen na SVM (Support Vector Machine). Njegov cilj je pronalazak optimalne hiperravnine kako bi ona razdvojila podatke različitih klasa. Često se koristi „One vs all“ pristup, u kojemu se svaka klasa razdvaja od ostalih. Ovaj je algoritam vrlo efikasan pri obradi podataka visoke dimenzionalnosti i velikih skupova podataka.

3.1.1. Matrica zabune

Matrica konfuzije sadrži informacije o očekivanim i pretpostavljenim vrijednostima našeg modela gdje redci predstavljaju pretpostavljenu vrijednost a stupci očekivanu. Glavna dijagonala predstavlja točno pretpostavljene elemente.

		Expected			
		1	2	3	4
Predicted	1	52	3	7	2
	2	2	28	2	0
	3	5	2	25	12
	4	1	1	9	40

Slika 3.1.1.1 Matrica konfuzije

Također, pomoću matrice konfuzije možemo također izračunati i točnost, osjetljivost i preciznost. Točnost je omjer točno klasificiranih primjera u odnosu na ukupan broj primjera. Pozitivni primjeri su nam daleko važniji (medicina, točnost daje istu težinu i pozitivnim i negativnim primjerima) te također u mnogim problemima imamo veliki nesrazmjer između broja pozitivnih i negativnih primjera.

U ovom projektu, na algoritmu logističke regresije dobijamo ovakav primjer:

	action	adult	adventure	animation	biography	comedy	crime	documentary	drama	family	fantasy
action	96	1	8	1	0	16	7	14	64	0	6
adult	1	53	4	1	0	19	0	11	19	0	0
adventure	10	8	32	1	0	19	1	20	32	3	3
animation	5	0	5	22	0	8	0	9	11	9	0
biography	0	0	0	0	0	1	0	33	6	0	0
comedy	24	6	5	4	0	833	6	80	383	14	1
crime	5	0	0	0	0	11	9	9	43	0	0
documentary	8	5	10	4	3	55	2	2,152	190	11	0
drama	38	9	2	3	4	215	16	210	1,799	15	2
family	1	0	2	8	0	33	0	24	31	38	0
fantasy	5	0	4	8	0	8	0	2	14	1	5

Slika 3.1.1.2 Matrica dobivena algoritmom logističke regresije

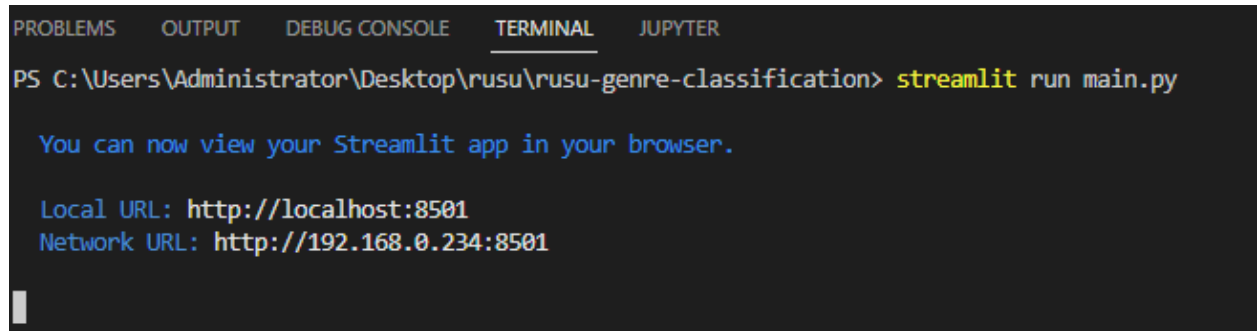
Uslikan je samo dio matrice zbog preglednosti (matrica zabune iz projekta ima 27 stupaca i redaka).

3. APLIKACIJA ZA PRIKAZ REZULTATA – STREAMLIT

Za prikaz rezultata modela se koristila Python-ova biblioteka Streamlit koja omogućuje podizanje web aplikacije.

Aplikaciju pokrećemo naredbom: `streamlit run main.py`

Nakon toga se podiže web sučelje, pali se default-ni browser na localhost:8501 i u terminalu VS Code-a se dobiva sljedeća povratna informacija:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  JUPYTER
PS C:\Users\Administrator\Desktop\rusu\rusu-genre-classification> streamlit run main.py

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.0.234:8501
```

Slika 4.1 Podizanje web aplikacije

Da bi na stranici išta bilo prikazano, moramo to napisati u python skripti. Slika 4.2 prikazuje implementaciju main funkcije ovog projekta.



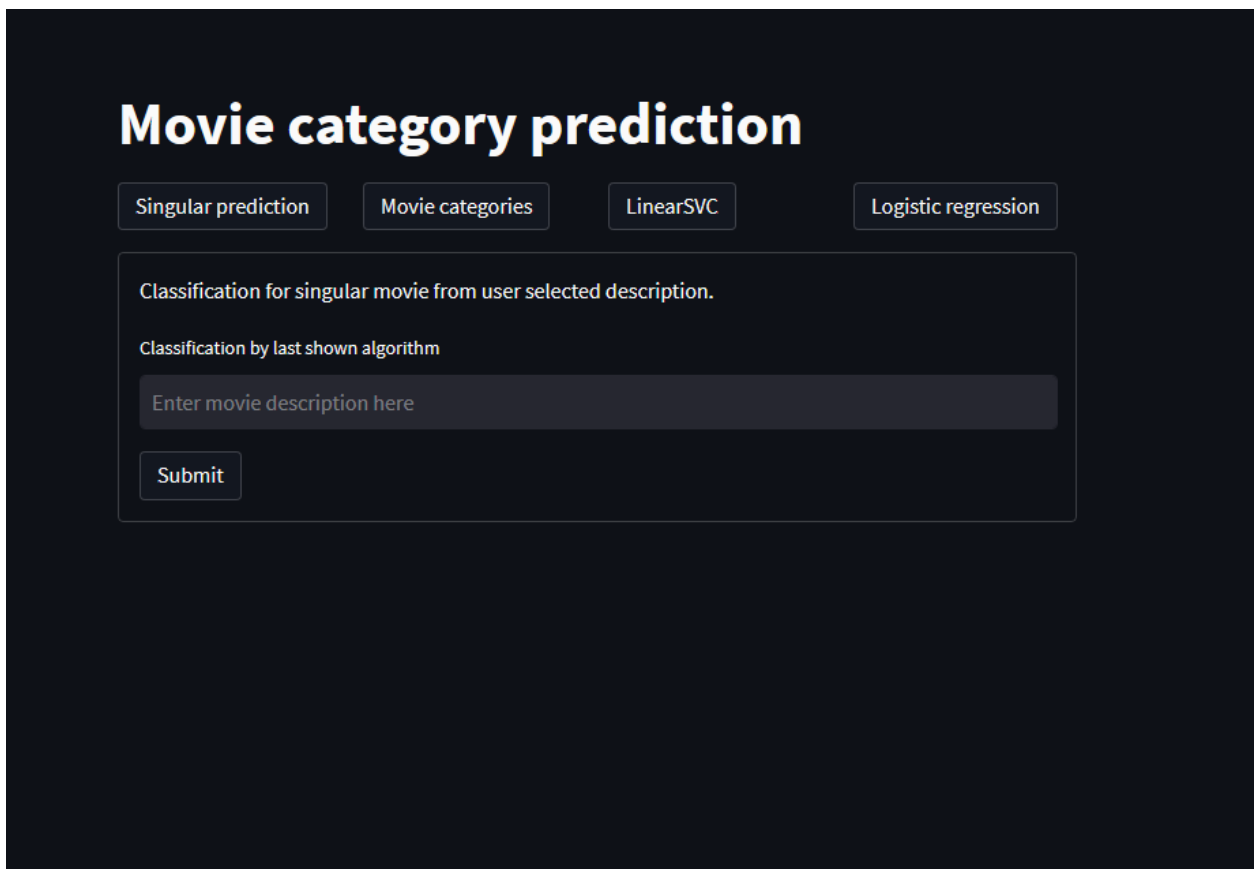
```
def main():
    st.title('Movie category prediction')
    col1, col2, col3, col4 = st.columns(4)
    with col1:
        SP = st.button('Singular prediction')
    with col2:
        MC = st.button('Movie categories')
    with col3:
        LinearSVC = st.button('LinearSVC')
    with col4:
        LR = st.button('Logistic regression')

    if MC:
        categories()
    elif LinearSVC:
        predict(1)
    elif LR:
        predict(2)
    else:
        with st.form("my_form"):
            st.write("Classification for singular movie from user selected description.")
            movieDesc = st.text_input('Classification by last shown algorithm', placeholder='Enter movie description here')

            submitted = st.form_submit_button("Submit")
            if submitted:
                st.write(predict_from_save(movieDesc))
```

Slika 4.2 Prikaz implementacije izgleda web sučelja

Web sučelje koje se pojavljuje izgleda kao prikazano na slici, uz otvorenu početnu stranicu Singular predistion.



Movie category prediction

Singular prediction

Movie categories

LinearSVC

Logistic regression

Classification for singular movie from user selected description.

Classification by last shown algorithm

Enter movie description here

Submit

Slika 4.3 Izgled web sučelja

Prilikom unošenja opisa filma i klikom na submit, dobiva se predikcija žanra filma. Za sljedeću fotografiju je unesen opis filma Flash (2023), te je očekivani rezultat Action / Adventure / Fantasy. Očekivani rezultat i opis se nalaze na sljedećem URL-u:

<https://www.imdb.com/title/tt0439572/>

Movie category prediction

Classification for singular movie from user selected description.

Classification by last shown algorithm

Barry Allen uses his super speed to change the past, but his attempt to save his family creates a wor

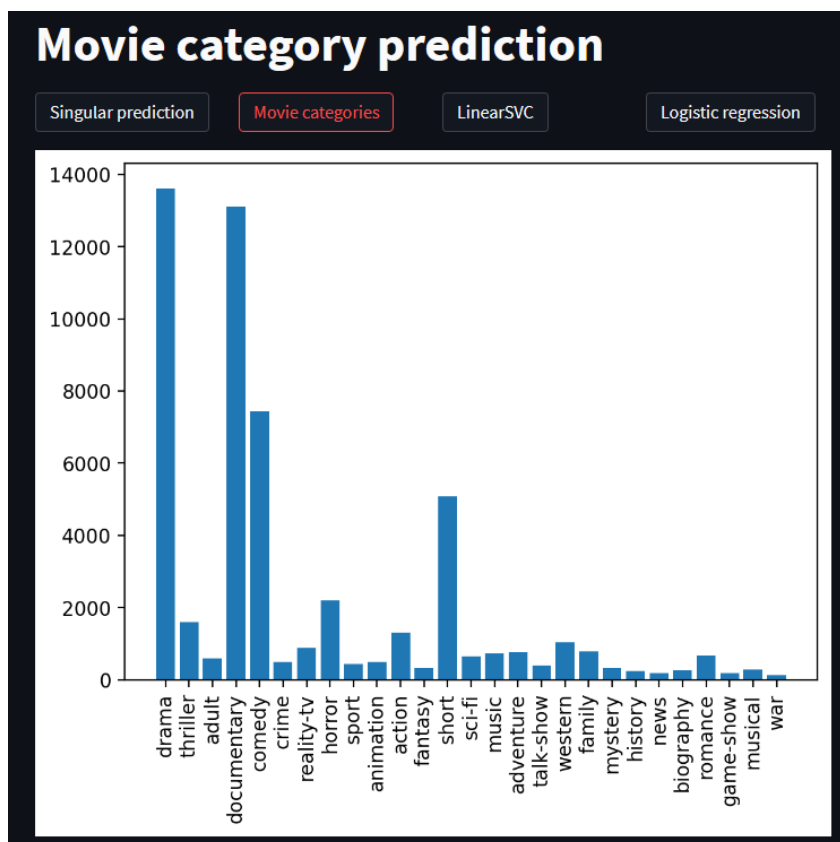
Submit

value

action

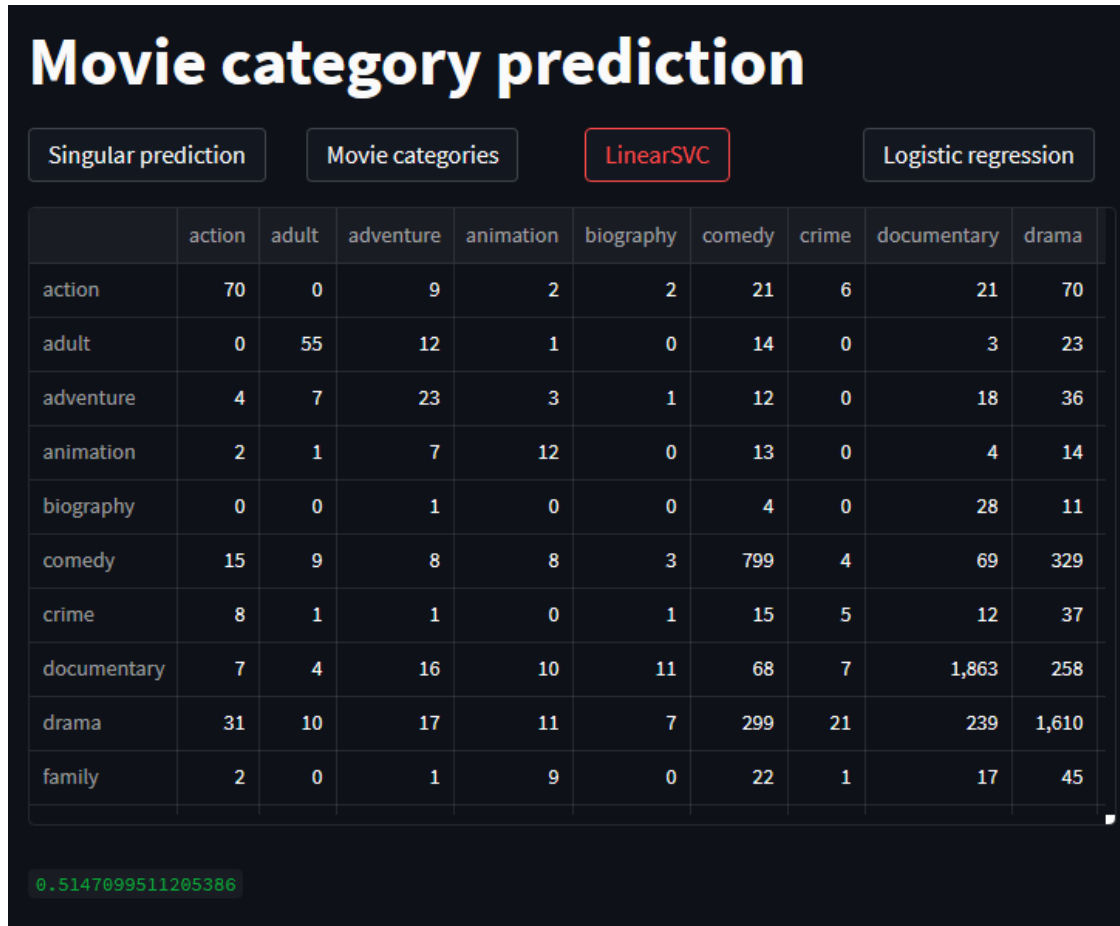
Slika 4.4 Prikaz rezultata za film Flash (2023)

Klikom na gumb „Movie categories“ se prikazuje sljedeći graf:



Slika 4.5 Prikaz podataka za filmske žanrove u datasetu

Klikom na gumb LinearSVC se prikazuje sljedeća matrica zabune s pripadnom točnosti:



Slika 4.6 LinearSVC s pripadajućom točnosti

Iz slike 4.6 se vidi da LinearSVC ima točnost od oko 51.5%. To je iz razloga loše raspodjeljenosti klasa u datasetu. Također, većina filmova se može svrstati u više od jednog žanra, a sukladno s time i više od jedne klase, no dataset korišten za ovaj projekt ima samo jedan žanr po filmu.

Klikom na zadnji gumb „Logistic Regression“ se prikazuje sljedeći graf:

Movie category prediction

Singular prediction

Movie categories

LinearSVC

Logistic regression

	action	adult	adventure	animation	biography	comedy	crime	documentary	drama
action	83	1	10	5	0	18	4	18	58
adult	0	39	10	0	0	17	0	7	13
adventure	5	4	39	2	1	18	0	13	39
animation	0	0	2	16	0	19	1	10	13
biography	0	0	0	0	0	3	0	30	10
comedy	11	4	7	2	0	931	1	58	354
crime	7	1	0	0	0	15	11	7	25
documentary	7	1	11	2	1	64	1	2,079	231
drama	36	7	11	1	2	279	7	206	1,884
family	1	1	2	6	0	34	0	27	32

0.5784377017430601

Slika 4.7 Logistic regression s pripadajućom tačnošću

Iz slike 4.7 se vidi da je tačnost Logističke regresije oko 57.8%. Tačnost je veća u odnosu na tačnost od LinearSVC, no i dalje je mala, a to je i dalje problem dataset-a koji je objašnjen kod slike 4.6.

4. ZAKLJUČAK

Korištenje prepoznavanja uzoraka i strojnog učenja se može koristiti za prikazivanje žanra filma ovisno o opisu istog, samo je potrebno imati podjednako raspoređenu količinu klasa i popis svih žanrova filma u tom datasetu.

Algoritam logističke regresije se ukazao boljim za oko 6.5% od LinearSVC algoritma.