****Московский государственный технический университет им. Н.Э. Баумана  
Кафедра «Системы обработки информации и управления»

Рубежный контроль №2  
по дисциплине  
«Методы машинного обучения»  
на тему

# **«Методы обработки текстов»**

Выполнил:  
студент группы ИУ5и-22М  
Джин Шуо

Москва — 2024 г.

## Варианты заданий

|  |  |  |
| --- | --- | --- |
| **Группа** | **Классификатор№1** | **Классификатор№2** |
| ИУ5И-22М | RandomForestClassifier | LogisticRegression |

**текст программы**

from sklearn.datasets import fetch\_20newsgroups

from sklearn.model\_selection import train\_test\_split

from sklearn.feature\_extraction.text import CountVectorizer, TfidfVectorizer

from sklearn.ensemble import RandomForestClassifier

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import classification\_report, accuracy\_score

categories = ['alt.atheism', 'comp.graphics', 'sci.space', 'talk.religion.misc']

newsgroups = fetch\_20newsgroups(subset='all', categories=categories)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(newsgroups.data, newsgroups.target, test\_size=0.3, random\_state=42)

# CountVectorizer

count\_vectorizer = CountVectorizer()

X\_train\_counts = count\_vectorizer.fit\_transform(X\_train)

X\_test\_counts = count\_vectorizer.transform(X\_test)

# TfidfVectorizer

tfidf\_vectorizer = TfidfVectorizer()

X\_train\_tfidf = tfidf\_vectorizer.fit\_transform(X\_train)

X\_test\_tfidf = tfidf\_vectorizer.transform(X\_test)

# CountVectorizer

# **Классификатор№1**RandomForestClassifier

rf\_classifier = RandomForestClassifier(random\_state=42)

rf\_classifier.fit(X\_train\_counts, y\_train)

y\_pred\_rf\_counts = rf\_classifier.predict(X\_test\_counts)

print("RandomForestClassifier with CountVectorizer")

print(classification\_report(y\_test, y\_pred\_rf\_counts))

print("Accuracy:", accuracy\_score(y\_test, y\_pred\_rf\_counts))

# **Классификатор№2**LogisticRegression

lr\_classifier = LogisticRegression(max\_iter=1000, random\_state=42)

lr\_classifier.fit(X\_train\_counts, y\_train)

y\_pred\_lr\_counts = lr\_classifier.predict(X\_test\_counts)

print("LogisticRegression with CountVectorizer")

print(classification\_report(y\_test, y\_pred\_lr\_counts))

print("Accuracy:", accuracy\_score(y\_test, y\_pred\_lr\_counts))

# TfidfVectorizer

# **Классификатор№1**RandomForestClassifier

rf\_classifier.fit(X\_train\_tfidf, y\_train)

y\_pred\_rf\_tfidf = rf\_classifier.predict(X\_test\_tfidf)

print("RandomForestClassifier with TfidfVectorizer")

print(classification\_report(y\_test, y\_pred\_rf\_tfidf))

print("Accuracy:", accuracy\_score(y\_test, y\_pred\_rf\_tfidf))

# **Классификатор№2**LogisticRegression

lr\_classifier.fit(X\_train\_tfidf, y\_train)

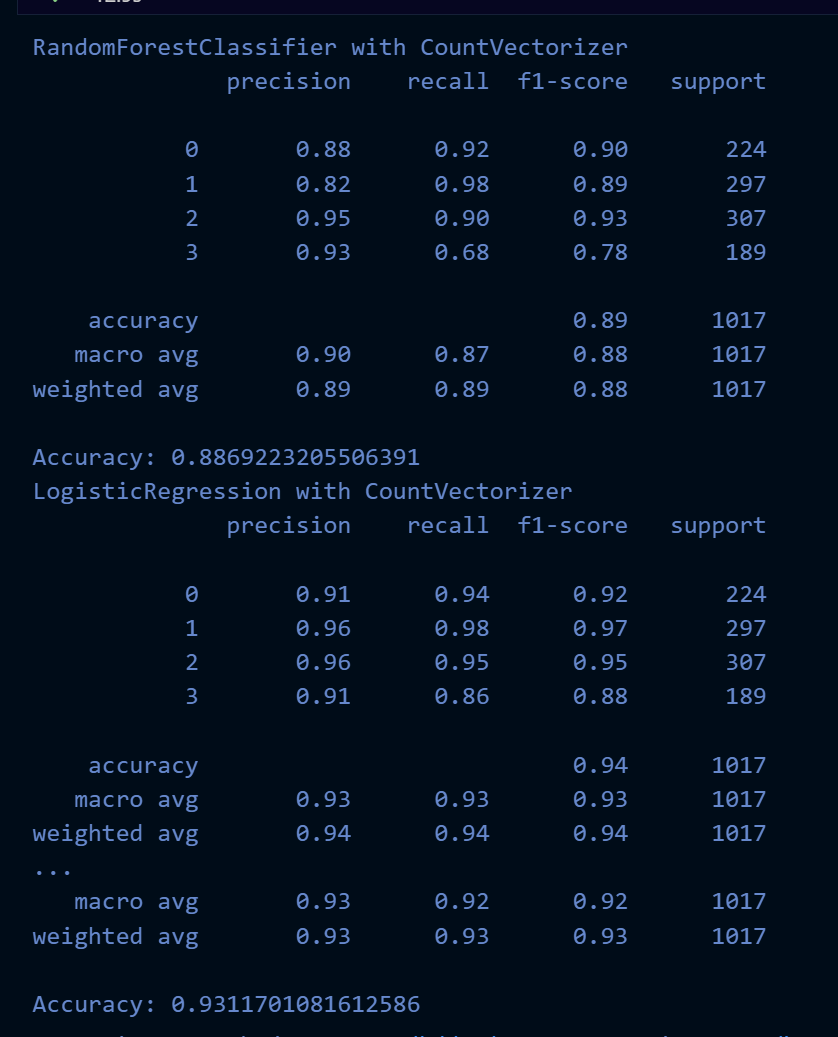
y\_pred\_lr\_tfidf = lr\_classifier.predict(X\_test\_tfidf)

print("LogisticRegression with TfidfVectorizer")

print(classification\_report(y\_test, y\_pred\_lr\_tfidf))

print("Accuracy:", accuracy\_score(y\_test, y\_pred\_lr\_tfidf))

**экранные формы с примерами выполнения программы**



**Список литературы**

[1] Гапанюк Ю. Е. LAB\_MMO\_\_DATA\_STORYЛабораторная работа №1Создание "истории о данных" (Data Storytelling)// GitHub. –– 2024. –– Режим доступа:https://github.com/ugapanyuk/courses\_current/wiki/MMO\_RK\_2