

## Поиск контактной информации мошенников в объявлениях маркетплейсов

**Смирнов Д.А.\***

Московский авиационный институт  
(национальный исследовательский университет)  
(ФГБОУ ВО МАИ (НИУ)), г. Москва, Российская Федерация  
ORCID: <https://orcid.org/0000-0002-7092-2612>  
e-mail: [daniil.smirnov2311@yandex.ru](mailto:daniil.smirnov2311@yandex.ru)

**Сологуб Г.Б.\*\***

Московский авиационный институт  
(национальный исследовательский университет)  
(ФГБОУ ВО МАИ (НИУ)), г. Москва, Российская Федерация  
ORCID: <https://orcid.org/0000-0002-5657-4826>  
e-mail: [glebsologub@ya.ru](mailto:glebsologub@ya.ru)

В статье изложен подход к определению наличия в объявлении о продаже товаров или предоставлении услуг контактной информации: номера телефона, электронной почты, ссылки на сайт, id в социальных сетях.

**Ключевые слова:** работа с данными, построение моделей.

### Для цитаты:

*Смирнов Д.А., Сологуб Г.Б.* Поиск контактной информации мошенников в объявлениях маркетплейсов // Моделирование и анализ данных. 2020. Том 10. № 4. С. 51–59.  
DOI: <https://doi.org/10.17759/mda.2020100405>

\**Смирнов Даниил Алексеевич*, студент, Московский авиационный институт (национальный исследовательский университет) (ФГБОУ ВО МАИ (НИУ)), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0002-7092-2612>, e-mail: [daniil.smirnov2311@yandex.ru](mailto:daniil.smirnov2311@yandex.ru)

\*\**Сологуб Глеб Борисович*, кандидат физико-математических наук, доцент кафедры, Московский авиационный институт (национальный исследовательский университет) (ФГБОУ ВО МАИ (НИУ)), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0002-5657-4826>, e-mail: [glebsologub@ya.ru](mailto:glebsologub@ya.ru)



## 1. ВВЕДЕНИЕ

В настоящее время существует много различных интернет-сервисов для размещения объявлений о товарах, вакансиях и резюме на рынке труда, а также услугах от частных лиц и компаний. Такими сервисами могут воспользоваться как добросовестные клиенты, так и злоумышленники, не по прямому назначению, а в целях обмана, мошенничества или спама. Представители этих сервисов заинтересованы в том, чтобы не допустить публикацию объявлений злоумышленниками, таким образом, возникает задача выявления таких объявлений, в том числе по их тексту.

В зависимости от конкретного сервиса принципы выявления объявлений злоумышленников могут быть различны. Так, в работе [1] описывается решение задачи выявления спама на сервисах типа Craigslist путем использования признаков на основе контента для обнаружения общего веб-спама и упоминается, что злоумышленники предпочитают не указывать номер телефона в объявлении. В то же время, на сервисах типа Avito, наоборот, правилами пользования ресурсом установлен запрет на указание любой контактной информации в любом месте объявления помимо специально выделенного окна сервиса [2], однако злоумышленники нарушают этот запрет и указывают номер телефона, маскируя его так, чтобы он не был автоматически распознан.

Так или иначе, наличие в тексте объявления контактной информации является важным признаком для выявления объявлений злоумышленников.

## 2. ПОСТАНОВКА ЗАДАЧИ

В качестве исходных данных составлены две таблицы, в которых каждая строка содержит информацию о конкретном объявлении: заголовок, описание, категория товара, подкатегория товара, цена, регион, город, дата публикации и маркер, указывающий на наличие и отсутствие контактной информации, нарушающей правила сервиса. Одна таблица используется для обучения модели, вторая – для валидации.

Требуется обучить модель, которая сможет для каждого объявления предсказывать, встретится ли там контактная информация. В результате работы обученная модель на вход получает объявление, а на выход выдает вероятность наличия контактной информации.

## 3. МЕТОД РЕШЕНИЯ

Для предсказания наличия контактной информации в объявлении нужно сформулировать алгоритм автоматических рекомендаций:

1. Проанализировать исходные данные, построить корреляционную матрицу для того, чтобы понять, как каждый столбец с информацией (заголовок, описание, категория и т.д.) влияет на целевую переменную (признак, который может принимать значения 0 и 1, где 0 указывает на отсутствие контактной информации,



а 1 – на ее наличие), построить графики зависимости, где переменными будут выступать столбцы с информацией.

2. Провести поиск слов, которые влияют на целевую переменную. На их основе составить новые столбцы с информацией об объявлении.
3. На полученном датасете подготовить обучающую и тестовую выборки.
4. Обучить предиктивные модели.
5. Оценить полученный результат.

Метрикой для оценки качества модели выбрана площадь (AUC – area under curve) под ROC-кривой (кривой ошибок). ROC-кривая – график, который позволяет оценить качество бинарной классификации. График показывает зависимость полноты (количество объявлений от общего числа реальных позитивных объектов, которые были предсказаны, как позитивный класс) от числа негативных объектов (объявления без контактной информации), которые предсказаны неверно.

Реализуем алгоритм на языке программирования Python. Для предобработки исходных данных хорошо подходят библиотеки pandas и numpy. Остальные шаги будем осуществлять при помощи библиотеки методов машинного обучения scikit-learn, библиотеки с реализацией градиентного бустинга lightgbm, морфологический анализатор pymorphy2, а также библиотеки для матчинга регулярных выражений re.

```
In [1]: from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier, RandomForestRegressor
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.feature_selection import RFECV, RFE
import lightgbm as lgb
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
import re
from sklearn.metrics import roc_auc_score
import pymorphy2
import phonenumbers
from statistics import mean
```

Рис. 1. Импорт необходимых для работы библиотек

В частности, обработку исходных данных проведем с помощью библиотеки pandas. Считаем файл с помощью функции read\_csv библиотеки pandas, после этого уберем строки, в которых есть незаполненные значения, с помощью функции dropna. Наконец, нужно переиндексировать строки, используя функцию reset\_index. Обработка данных проиллюстрирована на рис. 2.

Для поиска контактной информации проводится поиск слов, которые указывают на это. Например, «тел. 89991234567» видно, что номер телефона указан после сокращения слова телефон «тел.». Значит, наличие «тел.» увеличивает вероятность нахождения контактной информации в объявлении. Если встречаем такое слово, то в информацию об объявлении добавляется значение 1, иначе 0. Реализация показана на рис. 3.



```
In [6]: df_train = pd.read_csv("train.csv")
df_train.dropna(inplace=True)
df_train.reset_index(drop=True, inplace=True)
df_train.head(10)
```

Out[6]:

	title	description	subcategory	category	price	region	city	datetime_submitted	is_bad
0	Диван-хрвать	Продаем диван-хрвать. Удобный механизм - евро...	Мебель и интерьер	Для дома и дачи	7000.0	Россия	Москва	2019-06-01 00:00:15.180656	0
1	Кожух рулевой колонки Дач хф 91 4509834	Кожух рулевой колонки DAF XF 94 (60066004)/л ...	Запчасти и аксессуары	Транспорт	2290.0	Россия	Москва	2019-06-01 00:00:44.317933	0
2	Дешёвый буст аккумулятора Dosa 4	! Буст аккумуляторов емкостью выше 1000ммр не беру! ...	Предложение услуг	Услуги	200.0	Северная Осетия	Владикавказ	2019-06-01 00:00:50.249692	1
3	Телевизор sharp. Смарт тв. Интернет	Продам телевизор . Диагональ 450. наличие входа...	Аудио и видео	Бытовая электроника	25000.0	Калининградская область	Советск	2019-06-01 00:00:50.325799	1
4	Открытка-конверт	Открытки-конверты ручной работы/л.Выполнены в ...	Коллекционирование	Хобби и отдых	150.0	Ставропольский край	Ессентукская	2019-06-01 00:00:56.632655	0
5	Зимние шины Hankook Winter iPike RS6 W569	Размеры шин Hankook Winter iPike RS1 W319. Пр...	Запчасти и аксессуары	Транспорт	11000.0	Московская область	Железнодорожный	2019-06-01 00:01:03.471366	1
6	LADA Priora, 5046	примора 918 норма+кондиционер. 014 машина 16 в...	Автомобили	Транспорт	340000.0	Чеченская Республика	Грозный	2019-06-01 00:01:13.603386	1
7	Дверь входная	Продам дверь входную, данная дверь стояла от 3...	Ремонт и строительство	Для дома и дачи	3000.0	Россия	Санкт-Петербург	2019-06-01 00:01:33.818452	0
8	Джинсы фирмы Gulliver	Продаю джинсы фирмы Gulliver. Отличная посадка.	Детская одежда и обувь	Личные вещи	500.0	Россия	Москва	2019-06-01 00:01:41.136540	0
9	Кроссовки pike AIR MAX 570 premium	Куплены на ASOS, не подошел размер/л.Соответств...	Одежда, обувь, аксессуары	Личные вещи	8000.0	Россия	Москва	2019-06-01 00:01:55.506778	0

Рис. 2. Обработка исходных данных и результат

```
if desc.find("vk") != -1 or desc.find("id") != -1 or desc.find("vk") != -1 or \
desc.find("вконтакте") != -1:
    line.append(1)
else:
    line.append(0)
```

Рис. 3. Пример обработки слов, влияющих на наличие информации о связи с продавцом в VK

Уже имеющиеся столбцы с информацией кодируем с помощью LabelEncoder(), который представляет текстовые признаки в виде чисел.

```
le.fit(df.category)
df_train['category'] = le.transform(df.category)
```

Рис. 4. Пример кодирования текстового столбца исходных данных

Таким образом, обогащаем наши данные информацией, которая сильно коррелирует с целевой переменной. Данные выглядят следующим образом: первые столбцы – это преобразованные категориальные признаки с числовыми значениями, а после столбцы, которые могут принимать значения 1 и 0 – наличие или отсутствие эвристики, указывающей на наличие контактной информации.

В таком виде данные можно готовить к обучению модели. Разбиваем данные на 2 множества X и Y, X – информация, указывающая на наличие или отсутствие контактной информации, а множество Y – маркер присутствия контактной информации.



```
In [10]: df_train[range(len(df_train.iloc[0]) - 1)]
```

Out[10]:

	0	1	2	3	4	5	6	7	8	9	...	90	91	92	93	94	95	96	97	98	99
0	1.0	5.0	7000.0	2.0	4.0	0.0	0.0	0.0	0.0	1.0	...	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	3.0	3.0	2290.0	2.0	4.0	0.0	0.0	0.0	0.0	1.0	...	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
2	4.0	7.0	200.0	3.0	0.0	0.0	1.0	1.0	0.0	0.0	...	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
3	0.0	1.0	25000.0	0.0	6.0	0.0	0.0	0.0	1.0	0.0	...	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	5.0	4.0	150.0	4.0	2.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	3.0	3.0	11000.0	1.0	3.0	0.0	0.0	0.0	0.0	0.0	...	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0
6	3.0	0.0	340000.0	5.0	1.0	0.0	0.0	0.0	0.0	1.0	...	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	1.0	8.0	3000.0	2.0	5.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	2.0	2.0	500.0	2.0	4.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	2.0	6.0	8000.0	2.0	4.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

10 rows × 100 columns

Рис. 5. Вид данных после обработки

```
In [11]: df_train = pd.DataFrame(data_for_train)
x = np.array(df_train[[df_train.columns[i] for i in range(len(df_train.columns)-1)]]
y = np.array(df_train[df_train.columns[len(df_train.columns)-1]])
```

Рис. 6. Разбиение данных

Дальше считываем тестовый датасет аналогично обучающему на рис. 2.

```
In [12]: df_test = pd.read_csv("val.csv")
df_test.dropna(inplace=True)
df_test.reset_index(drop=True, inplace=True)
```

Рис. 7. Считывание и обработка тестовых данных

Следующим этапом является обучение модели на подготовленном датасете. Будем обучать такие модели, как логистическая регрессия, случайный лес для классификации и градиентный бустинг для классификации.

В основе логистической регрессии лежит логистическая функция [7]. Результат измеряется с помощью дихотомической переменной (в которой есть только два возможных результата), которой является целевая переменная  $Y$ . В поставленной задаче результатом является число, показывающее уверенность модели в наличии контактной информации в объявлении [6].

```
In [7]: from sklearn.linear_model import LogisticRegression

classifier = LogisticRegression(random_state = 0)

classifier.fit(xtrain, ytrain)
```

Рис. 8. Пример обучения логистической регрессии



Остаётся протестировать полученную модель. Выбранная ранее метрика оказалась равна лишь 0.77. (рис. 9)

```
In [11]: from sklearn.linear_model import LogisticRegression
from sklearn.metrics import roc_auc_score

classifier = LogisticRegression(random_state = 0)
classifier.fit(xtrain, ytrain)
y_pred = classifier.predict(xtest)

print ("ROC-AUC: ", roc_auc_score(ytest, y_pred))

/home/daniil/.local/lib/python3.6/site-packages/sklearn/utils/validation.py:72: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using.ravel().
  return f(**kwargs)
ROC-AUC:  0.7691636803246837
```

Рис. 9. Построение модели логистической регрессии для поставленной задачи

Случайным лесом называется метод, подразумевающий создание нескольких деревьев принятия решений [7], которые затем принимают коллективное решение на основе голосования о том, каким образом классифицировать входящие значения [7].

Для построения модели случайного леса разделяем данные на обучающую и тестовую выборки и обучаем на подготовленных данных. После тестирования полученной модели получаем значение метрики ROC-AUC равное 0.82.

```
In [48]: model_forest = RandomForestClassifier()
model_forest.fit(x,y)

Out[48]: RandomForestClassifier()

In [49]: print('roc-auc: {:.5f}'.format(roc_auc_score(y_test, model_forest.predict(x_test))))

roc-auc: 0.81983
```

Рис. 10. Пример обучения и тестирования случайного леса

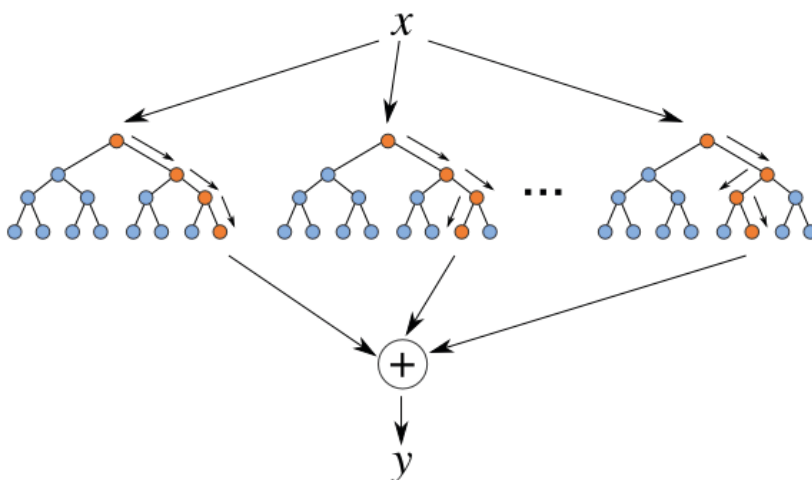


Рис. 11. Изображение случайного леса



Градиентный бустинг – это техника машинного обучения для задач классификации и регрессии, которая строит модель предсказания в форме ансамбля слабых предсказывающих моделей. Данный метод основан на градиентном спуске, основная идея которого заключается в том, чтобы идти в направлении наискорейшего спуска, которое задаётся антиградиентом:

$$\vec{x}^{[j+1]} = \vec{x}^{[j]} - \lambda^{[j]} \nabla F(\vec{x}^{[j]})$$

где  $\lambda^{[j]}$  задает скорость градиентного спуска,  $\vec{x}$  - аргумент функции  $F(\vec{x}) : X \rightarrow \mathbb{R}$ , которую оптимизируем  $F(\vec{x}) \rightarrow \min_{\vec{x} \in X}$ . Градиентом является вектор частных производных по каждой переменной.

Как средство борьбы с переобучением используем метод recursive feature elimination. Он последовательно перебором исключает признаки из набора и считает точность на тестовой выборке. Набор признаков с наилучшим результатом является выходом алгоритма.

```
In [8]: from sklearn.metrics import roc_auc_score

predicted = model.predict(x_test)
roc_auc = roc_auc_score(y_test, predicted)
print('roc-auc: {:.2f}'.format(roc_auc))

roc-auc: 0.96
```

Рис. 12. Точность модели градиентного бустинга на тестовых данных

Далее на рис. 13 представлен пример работы алгоритма: для трех случайных объявлений модель предсказывает вероятность наличия контактной информации.

```
In [58]: predicted = model.predict(x_test)
for index, text in data_for_test.items():
    print()
    print('Описание объявления: \n{}\n'.format(text))
    print('Предсказание модели: {}'.format(predicted[index]))

Описание объявления:
Шины зимние 6/у Marshal Assimetric I Zen KW 61 681/41 г 63 диски AVA литые универсальные 4 J 63 3/600 или 3/663,2 ET 28-31. Забрать можно на Ярославском шоссе у МКАД или в г. Сергиев Посад. Могу подвезти(по договорённости) Состояние- на 6-5 сезона. БЕЗ ТОРГА и. 8 964 688 36 68

Предсказание модели: 0.9920922326106677

Описание объявления:
Продам авто в хорошем состоянии.все вопросы по тел.Двигатель стоит родной,611л/с ,самый надёжный в этой линейке.Салон фешаля-трансформер,очень редкая комплектация:диван раскладывается и получается двухспальная кровать,выдвижной столик,монитор,люк,электро жабры,электро стеклоподъемники,электро зеркала с подогревом,кожаный салон,передняя подушка безопасности,потолок алькантара,кондиционер,вебаста с таймером,очень удобно зимой. ABS, ESP, ASR, блокировка передних колёс (M), Теперь из не недостатков:присутствует не большая коррозия в углу на правом пороге, притертость не значит ельная на сдвижной двери.От машины не избавляясь, езжу каждый день, так что перекупам и стоянка просьба не беспокоит ь.Торг приветствуется,в подарок отдам зимнюю резину на литых дисках.звоните 8 девятьсот девятнадцать694 тридцать четыре 29 .

Предсказание модели: 0.9942272904703925

Описание объявления:
toyota observe g5-ice без грыж и порезов звонить на( 89155903052)

Предсказание модели: 0.9978764108431211
```

Рис. 13. Пример результата работы алгоритма

Также посчитаем метрику ROC-AUC по каждой категории объявлений для построенной модели. Реализация и результат продемонстрированы на рис. 14.





```
In [25]: ans_x = dict()
ans_y = dict()
cats = df_test["category"].unique()
for cat in cats:
    ans_x[cat] = []
    ans_y[cat] = []
for i in range(len(df_test.description)):
    ans_x[df_test.category[i]].append(x_test[i])
    ans_y[df_test.category[i]].append(y_test[i])
auc_res = {}
for cat in cats:
    data_x = ans_x[cat]
    data_y = ans_y[cat]
    pred = model.predict(data_x)
    roc_auc = roc_auc_score(data_y, pred)
    auc_res[cat] = roc_auc
print(auc_res)
mean(auc_res[i] for i in auc_res.keys())

{'Транспорт': 0.9699697072971366, 'Для бизнеса': 0.7988688582537999, 'Для дома и дачи': 0.9318111268237161, 'Личные вещи': 0.80
128781480516, 'Услуги': 0.8766427795312258, 'Бытовая электроника': 0.9437723925159236, 'Недвижимость': 0.9718166273316559, 'Хоб
би и отдых': 0.9466941015889163, 'Животные': 0.9122730573710965, 'Работа': 0.8337569654902727}
```

Рис. 14. Алгоритм оценки модели и результат по каждой категории.

## 4. ЗАКЛЮЧЕНИЕ

Поставлена задача обучения модели поиска контактной информации мошенников в объявлениях маркетплейсов. Сформулирован алгоритм решения этой задачи на основе методов машинного обучения путем применения градиентного бустинга для классификации и отбора признаков. Описана реализация этого алгоритма на языке Python с применением свободно-распространяемых библиотек подпрограмм.

Предложенный алгоритм может быть использован для построения моделей поиска любой контактной информации человека в текстовых данных.

### Литература

1. Spam Detection in Online Classified Advertisements / Hung Tran [и др.] // WebQuality 11: Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality / Association for Computing Machinery, 2011, pp.35–41, ISBN: 978–1–4503–0706–2, doi:10.1145/1964114.1964122
2. URL: [https://m.avito.ru/info/polzovatel'skoe\\_soglashenie](https://m.avito.ru/info/polzovatel'skoe_soglashenie)
3. Daniel Jurafsky, James H. Martin. Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Second Edition. Pearson Education International, 2009. 1024 pp.
4. Максимов Ю.А. Алгоритмы линейного и дискретного программирования. – М.: МИФИ, 1980.
5. Jones K.S. A statistical interpretation of term specificity and its application in retrieval // Journal of Documentation. – MCB University Press. – 2004. – Vol. 60, no. 5. – P. 493–502. – ISSN 0022–0418.
6. Электронный ресурс: URL:<https://machinelearningmastery.com/logistic-regression-tutorial-for-machine-learning/>
7. Грас Джозел. DataScience. Наука о данных с нуля, [пер. с англ. Андрея Логунова]. – Санкт-Петербург, 2017. ISBN 978–5–9775–3758–2.





## Search Contact Information of Scams in Classifieds of Marketplace

### **Daniil A. Smirnov\***

Moscow Aviation Institute (National Research University), Moscow, Russia

ORCID: <https://orcid.org/0000-0002-7092-2612>

e-mail: [daniil.smirnov2311@yandex.ru](mailto:daniil.smirnov2311@yandex.ru)

### **Gleb B. Sologub\*\***

Moscow Aviation Institute (National Research University), Moscow, Russia

ORCID: <https://orcid.org/0000-0002-5657-4826>

e-mail: [glebsologub@ya.ru](mailto:glebsologub@ya.ru)

The article describes an approach to determining whether an ad for sale or provision of services contains contact information: phone numbers, email addresses, website links, social media IDs etc.

**Keywords:** working with data, building models.

### **For citation:**

Smirnov D.A., Sologub G.B. Search Contact Information of Scams in Classifieds of Marketplaces. *Modelirovanie i analiz dannykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 4, pp. 51–59. DOI: <https://doi.org/10.17759/mda.2020100405> (In Russ., abstr. in Engl.).

### **References**

1. Spam Detection in Online Classified Advertisements / Hung Tran [и др.]. WebQuality'11: Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality / Association for Computing Machinery, 2011, pp.35–41, ISBN: 978–1-4503–0706–2, doi:10.1145/1964114.1964122
2. URL:[https://m.avito.ru/info/polzovatel'skoe\\_soglasenie](https://m.avito.ru/info/polzovatel'skoe_soglasenie)
3. Daniel Jurafsky, James H. Martin. Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Second Edition. *Pearson Education International*, 2009. 1024 pp.
4. Maksimov Yu. A. Algoritmy lineinogo i diskretnogo programmirovaniya. – Moscow.: MEPHI, 1980.
5. Jones K.S. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation. MCB University Press*, 2004. Vol. 60, no. 5, pp. 493–502. ISSN 0022–0418.
6. Elektronii resurs: URL:<https://machinelearningmastery.com/logistic-regression-tutorial-for-machine-learning/>
7. Gras Dzhoel. Data Science. Nauka o dannykh s nulya, [per. s angl. Andrey Logunova]. – Sankt-Peterburg, 2017. ISBN 978–5-9775–3758–2.

\***Daniil A. Smirnov**, undergraduate student of the Institute of Information Technology and Applied Mathematics, Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: <https://orcid.org/0000-0002-7092-2612>, e-mail: [daniil.smirnov2311@yandex.ru](mailto:daniil.smirnov2311@yandex.ru)

\***Gleb B. Sologub**, candidate of physical and mathematical sciences, associate professor of the Department of Mathematical Cybernetics, Institute of Information Technologies and Applied Mathematics, Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: <https://orcid.org/0000-0002-5657-4826>, e-mail: [glebsologub@ya.ru](mailto:glebsologub@ya.ru)