Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Саратовский государственный технический университет
имени Гагарина Ю.А.»

Институт прикладных информационных технологий и коммуникаций
Кафедра «Информационно-коммуникационные системы и программная инженерия»

Курсовая работа
По дисциплине «Нейронные сети»
Разработка нейронной сети для классификации новостей

Выполнил: студент группы б1ИВЧТ-41
Номер зачетной книжки: 210013
Кудряшов Алексей Владимирович

подпись студента

Руководитель: канд. физ.-мат. наук, доцент кафедры ИКСП
Ивженко Сергей Петрович

подпись руководителя

Саратов 2022

Содержание

Оглавление

Введение

Нейронная сеть — математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма. Это понятие возникло при изучении процессов, протекающих в мозге, и при попытке смоделировать эти процессы.

Среди основных областей применения нейронных сетей:

- Прогнозирование;
- принятие решений;
- распознавание образов;
- оптимизация;
- анализ данных.

Нейросети лежат в основе большинства современных систем распознавания и синтеза речи, а также распознавания и обработки изображений.

Таким образом, целью данной курсовой работы является разработка нейронной сети для классификации новостей

Для достижения поставленной цели необходимо решить следующие задачи

- выбрать набор данных для обучения нейронной сети;
- реализовать нейронную сеть;
- обучить нейронную сеть на тренировочных данных;
- проверить работу нейронной сети на тестовых данных.

Создание нейронной сети

Датасет

Нейронная сеть обучится при помощи датасета новостей с сайта BBC.

Наборы данных новостных статей, созданные BBC News, предназначены для использования в качестве контрольных показателей для исследований в области машинного обучения. Исходные данные обрабатываются для формирования единого CSV-файла для простоты использования, заголовок новости и имя соответствующего текстового файла сохраняются вместе с содержанием новости и ее категорией.

Содержит в себе:

- 2225 документов с новостного сайта Би-би-си, соответствующих сюжетам в пяти тематических областях за 2004–2005 годы.
- 5 классов (бизнес, развлечения, политика, спорт, технологии)

Фрагмент датасета представлен на рисунке 1.

	category	filename	title	content
0	business	001.txt	Ad sales boost Time Warner profit	Quarterly profits at US media giant TimeWarne...
1	business	002.txt	Dollar gains on Greenspan speech	The dollar has hit its highest level against ...
2	business	003.txt	Yukos unit buyer faces loan claim	The owners of embattled Russian oil giant Yuk...
3	business	004.txt	High fuel prices hit BA's profits	British Airways has blamed high fuel prices f...
4	business	005.txt	Pernod takeover talk lifts Domecq	Shares in UK drinks and food firm Allied Dome...

Рисунок 1. Фрагмент датасета

Таким образом датасет состоит из: категории, имени файла, названия новости и её содержание.

Реализация нейронной сети

Нейронная сеть будет реализована на языке программирования Python. Для обучения сети будет использоваться пакет Scikit-learn.

Scikit-learn (sklearn) — это один из наиболее широко используемых пакетов Python для Data Science и Machine Learning. Он содержит функции и алгоритмы для машинного обучения: классификации, прогнозирования или разбивки данных на группы.

Для установки Scikit-learn нужно воспользоваться командой

pip install scikit-learn

Для работы нейронной сети нужно импортировать следующие библиотеки (рисунок 2)

```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
```

Рисунок 2. Требуемые библиотеки

Подготовка данных (рисунок 3): x – заголовок новости, y – категория новости

```
x = np.array(data["title"])
y = np.array(data["category"])

cv = CountVectorizer()
X = cv.fit_transform(x)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=42)
```

Рисунок 3. Подготовка данных

Как видно из рисунка 3 после подготовки данных идёт нормализация данных и разделяет данные на случайные тестовые множества.

После этого создается модель при помощи MultinomialNB (рисунок 4) - Наивный классификатор Байеса для моделей многочлена. Наивный классификатор Байеса многочлена подходит для классификации с дискретными функциями (например, подсчеты слов для классификации текстов).

```
model = MultinomialNB()
model.fit(X_train,y_train)
```

Рисунок 4. Создание модели

Далее метод fit обучает модель на выборке, которую мы сделали немного выше.

Отдельное приложение

Создание визуальной программы

Для создания визуальной программы потребуются следующие библиотеки (рисунок 5):

```
import tkinter as tk
import tkinter.font as tkFont
from tkinter.messagebox import showerror, showinfo
```

Рисунок 5. Требуемые библиотеки

По пунктам:

- Настройки главного окна: названия, расположения, размеров (рисунок 6)

```
#setting title
root.title("News category predict")
#setting window size
width=927
height=415
screenwidth = root.winfo_screenwidth()
screenheight = root.winfo_screenheight()
alignstr = '%dx%d+%d+%d' % (width, height, (screenwidth - width) / 2, (screenheight - height) / 2)
root.geometry(alignstr)
root.resizable(width=False, height=False)
```

Рисунок 6. Настройки главного окна

- Настройки каждой кнопки: шрифт, размер текста, размер кнопки, расположения. В данном случае (рисунок 7) настройки кнопки Load dataset

```
GButton_301=tk.Button(root)
GButton_301["bg"] = "#f0f0f0"
ft = tkFont.Font(family='Times',size=10)
GButton_301["font"] = ft
GButton_301["fg"] = "#000000"
GButton_301["justify"] = "center"
GButton_301["text"] = "Load Dataset"
GButton_301.place(x=30,y=40,width=118,height=30)
GButton_301["command"] = self.GButton_301_command
```

Рисунок 7. Настройки кнопок

- Для других элементов приложения код практически идентичен (рисунок 8)

```
GLineEdit_154=tk.Entry(root)
GLineEdit_154["borderwidth"] = "1px"
ft = tkFont.Font(family='Times',size=10)
GLineEdit_154["font"] = ft
GLineEdit_154["fg"] = "#333333"
GLineEdit_154["justify"] = "center"
GLineEdit_154["text"] = "Input text"
GLineEdit_154.place(x=30,y=300,width=873,height=30)
```

Рисунок 8. Настройка поле ввода текста

- Для наглядности были добавлены дополнительные окна предупреждения или ошибок (рисунок 9)

```
    showinfo(title="Информация", message="Сеть обучена" )
else:
    showerror(title="Информация", message="База данных не загружена" )
```

Рисунок 9. Код дополнительных окон

Показ работы программы

После запуска программы нас встречает такое окно (рисунок 10):

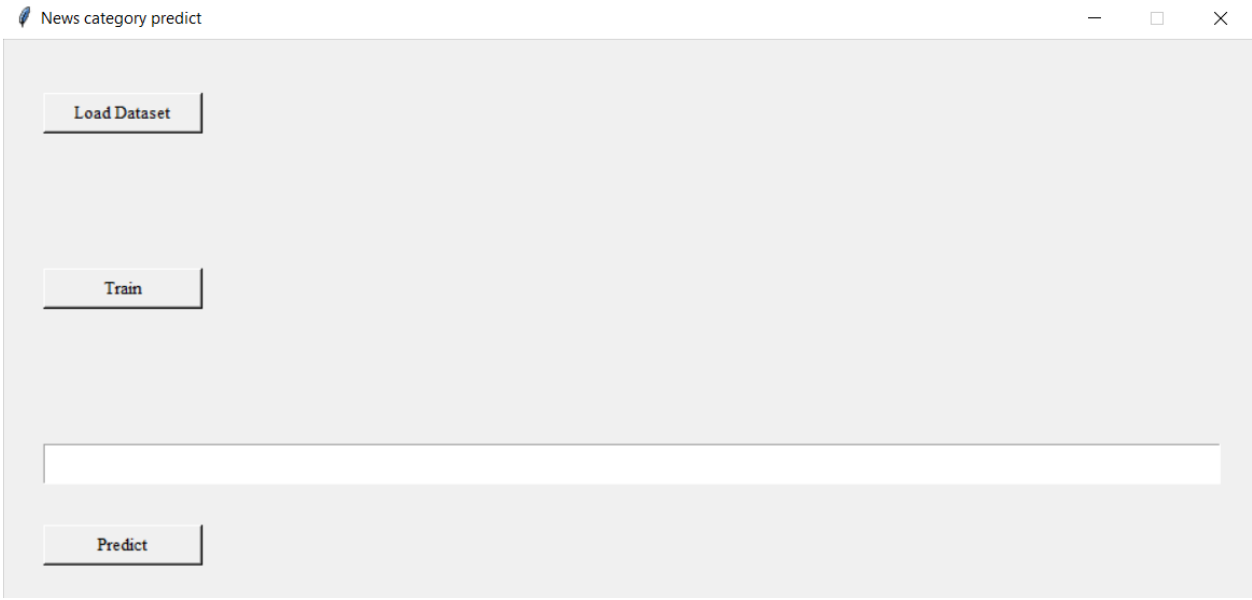

Рисунок 10. Главный экран программы

В данном окне реализованы кнопки для загрузки датасета, обучение сети, поля для ввода данных и кнопка для получения результата.

Для наглядной демонстрации функционала разберем каждую кнопку по отдельности:

- Load Dataset

После загрузки датасета появляется уведомление, что она была загружена (рисунок 11), также рядом с кнопкой загрузки появляется 5 строчек из датасета (рисунок 12)

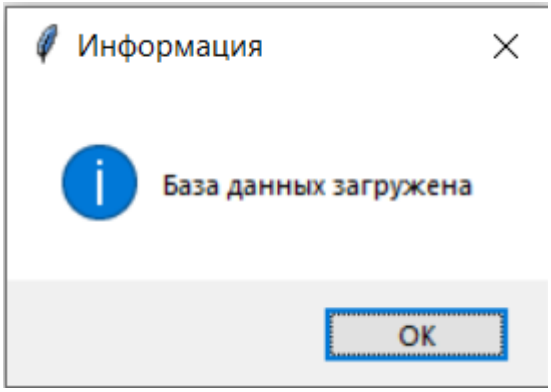

Рисунок 11. Уведомление

	category	filename	title	content
0	business	001.txt	Ad sales boost Time Warner profit	Quarterly profits at US media giant TimeWarne...
1	business	002.txt	Dollar gains on Greenspan speech	The dollar has hit its highest level against ...
2	business	003.txt	Yukos unit buyer faces loan claim	The owners of embattled Russian oil giant Yuk...
3	business	004.txt	High fuel prices hit BA's profits	British Airways has blamed high fuel prices f...
4	business	005.txt	Pernod takeover talk lifts Domecq	Shares in UK drinks and food firm Allied Dome...

Рисунок 12. Пара строк из датасета

- Кнопка train обучает сеть (рисунок 13)

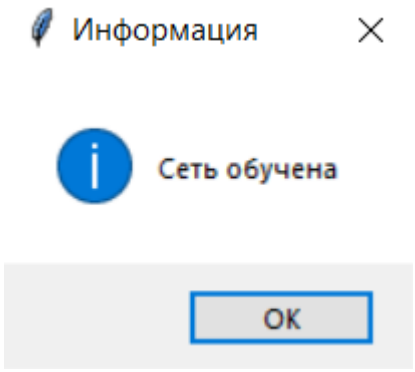

Рисунок 13. Сеть обучена

- Поле ввода текста (рисунок 15)

Рисунок 14. Поле ввода текста

- Predict – показать результат. После нажатия этой кнопки происходит анализ текста и рядом с кнопкой появляется возможная категория, которая соответствует тексту, написанному в поле ввода (рисунок 15)

Рисунок 15. Predict

Если мы не загрузим датасет или не обучим сеть нас встретит ошибка (рисунок 16)

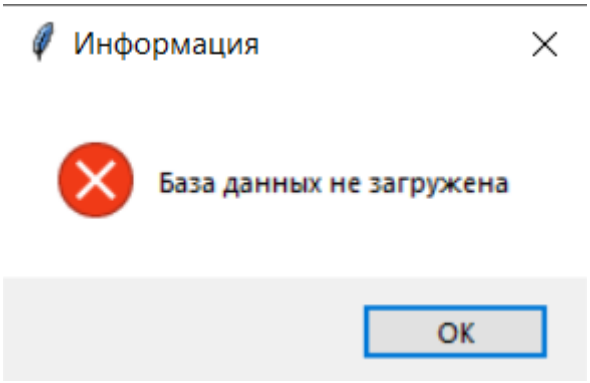

Рисунок 16. Ошибка

ЗАКЛЮЧЕНИЕ

Таким образом, в рамках курсовой работы была реализована нейронная сеть, решающая задачу классификации новостей, которая была обучена с помощью модуля sklearn.

В результате работы можно сделать вывод, что нейросеть хорошо определяет категорию новостей.