

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ
РАБОТА
СИНТЕЗ РЕКУРРЕНТНОЙ НЕЙРОННОЙ
СЕТИ ДЛЯ КЛАССИФИКАЦИИ ОТЗЫВА
ПОКУПАТЕЛЯ
ПО ПРОГРАММЕ ПРОФЕССИОНАЛЬНОЙ
ПЕРЕПОДГОТОВКИ:
АНАЛИЗ ДАННЫХ НА RUTNOM**

Выполнила: Исхакова Альфия Ильгизовна

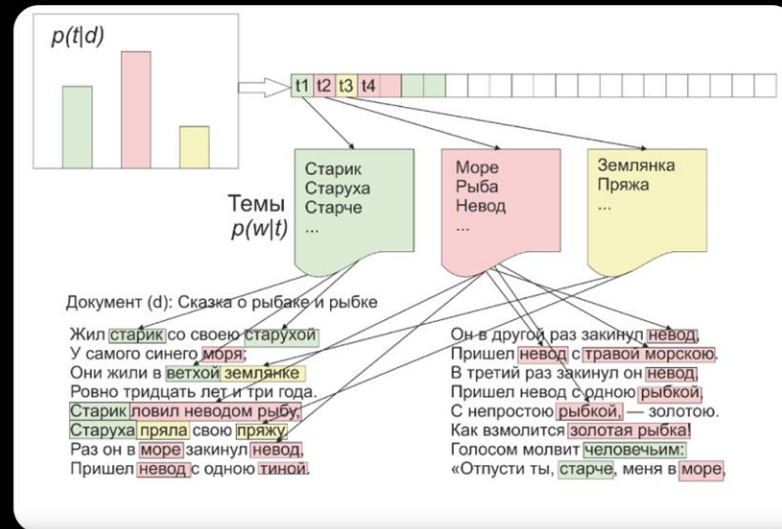
Научный руководитель: Семендяев Родион Юрьевич

Г. Санкт-Петербург, 2023

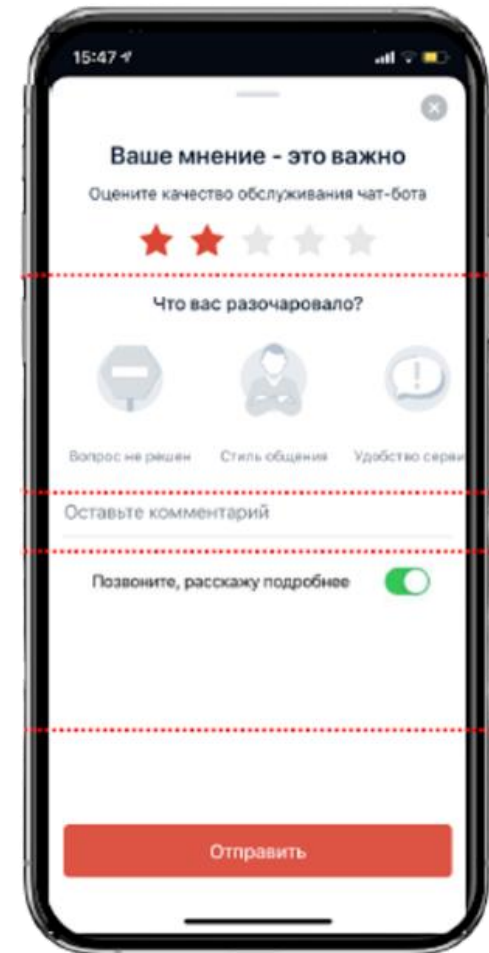
Важность

- «Альфа-банк» решил определить тональность отзывов. Было собрано более 100 млн. оценок и отзывов. И проведён их анализ с помощью машинного обучения.

КАК И ЗАЧЕМ мы начали искать бизнес-инсайты в отзывах клиентов с помощью машинного обучения



Alfa Digital

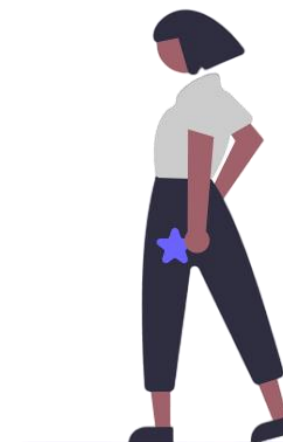
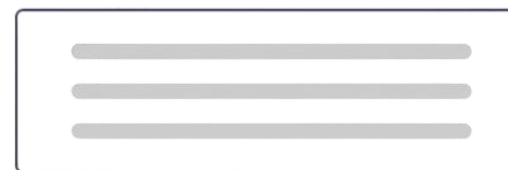
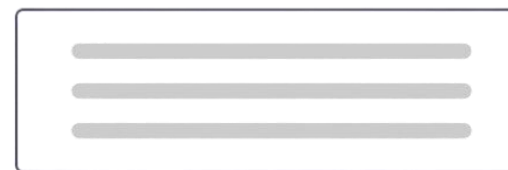


Цель

Цель работы: проанализировать тональность отзывов на товар на основе методов машинного обучения.

Задачи

1. Очистить данные
2. Дополнить классы до приемлемого уровня
3. Подготовить данные к работе с нейронной сети
 1. Токенизация
 2. Лемматизация
 3. Удаление стоп слов
4. Построение архитектуры рекуррентной нейронной сети
5. Подбор архитектуры нейронной сети

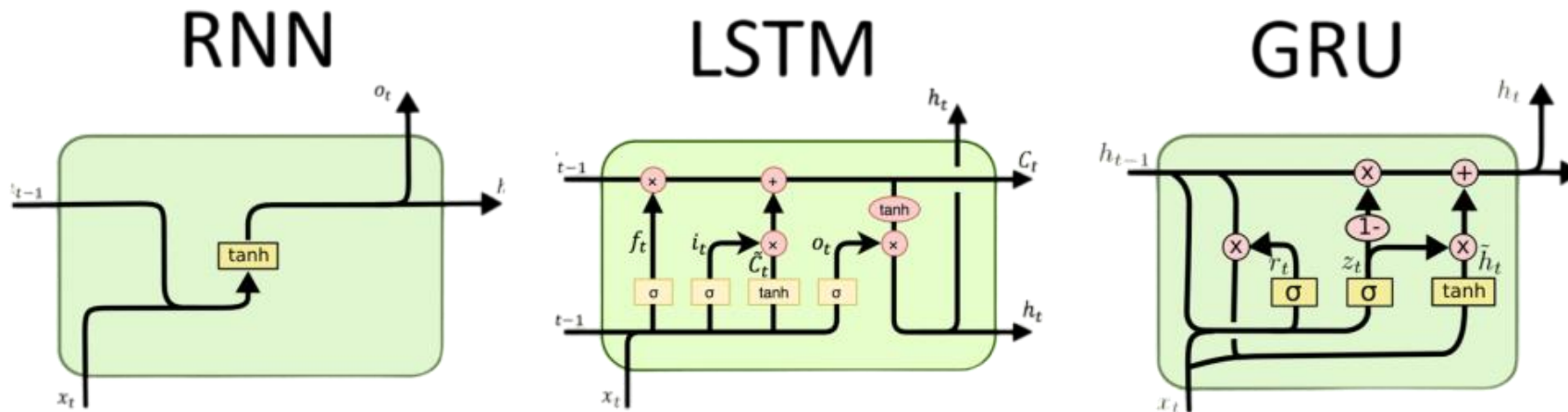


Рекуррентные нейронные сети

- Рекуррентные нейронные сети— нейронная сеть, основной особенностью которых является использование обратных связей между элементами сети, что позволяет ей учитывать контекст предыдущих входных данных.
- Основным принципом работы RNN заключается в использовании обратной связи.
- У RNN есть три входа - текущий вход, скрытое состояние и предыдущий выход.
- Имея такие входы, RNN может обрабатывать последовательные данные, сохраняя информацию о предыдущих состояниях.

Виды рекуррентных нейронных сетей

- Простая рекуррентная нейронная сеть (Simple RNN)
- LSTM-сети (Long Short-Term Memory)
- GRU-сети (Gated Recurrent Units)



Задача 1

- Данные в датасете:
 - Имя продукта: пропусков нет
 - Цена: пропусков нет
 - Оценка: пропусков нет
 - Краткий отзыв: 12% пропусков
 - Развёрнутый обзор: 0,005% пропусков
 - Настроение: пропусков нет
- Все данные имеют тип object

Column	Dtype
-----	-----
product_name	object
product_price	object
Rate	object
Review	object
Summary	object
Sentiment	object

Задача 1

- Заполняем нулевые значения
- Избавляемся от мусорных данных

```
df['Review'][47616] = 'perfect product!'
df['Summary'][[28417, 36859, 39838, 47616, 79972, 91051, 95049]] = 'awesome product'
```

```
df['Summary'][[11037, 40502, 132139]] = 'good'
```

```
df['Summary'][50559] = 'bad'
df['Review'][50559] = 'not good'
```

```
mask1 = (df['Review'].isnull()) & (df['Rate'] == 1)
mask2 = (df['Review'].isnull()) & (df['Rate'] == 2)
mask3 = (df['Review'].isnull()) & (df['Rate'] == 3)
mask4 = (df['Review'].isnull()) & (df['Rate'] == 4)
mask5 = (df['Review'].isnull()) & (df['Rate'] == 5)
```

```
df['Review'] = df['Review'][mask1] = 'horrible'
df['Review'] = df['Review'][mask2] = 'not good'
df['Review'] = df['Review'][mask3] = 'fair'
df['Review'] = df['Review'][mask4] = 'good choice'
df['Review'] = df['Review'][mask5] = 'perfect product!'
```

```
df['Rate'].unique()
```

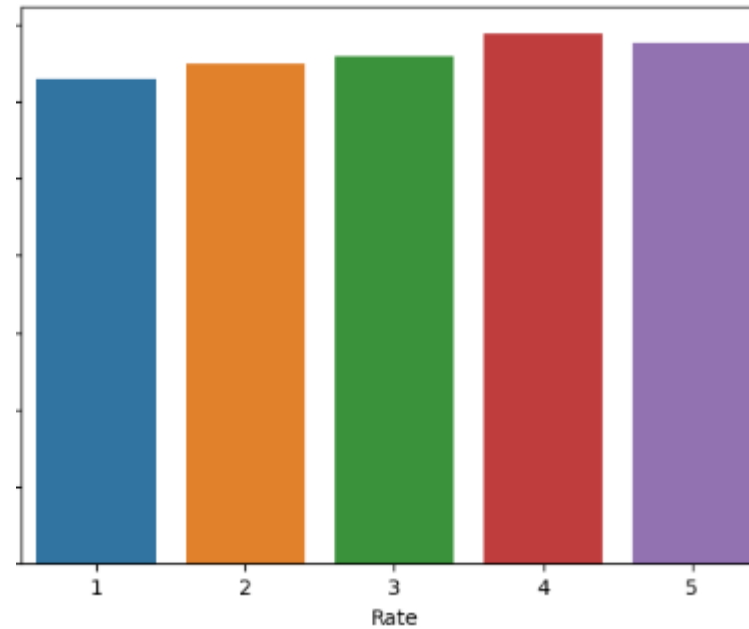
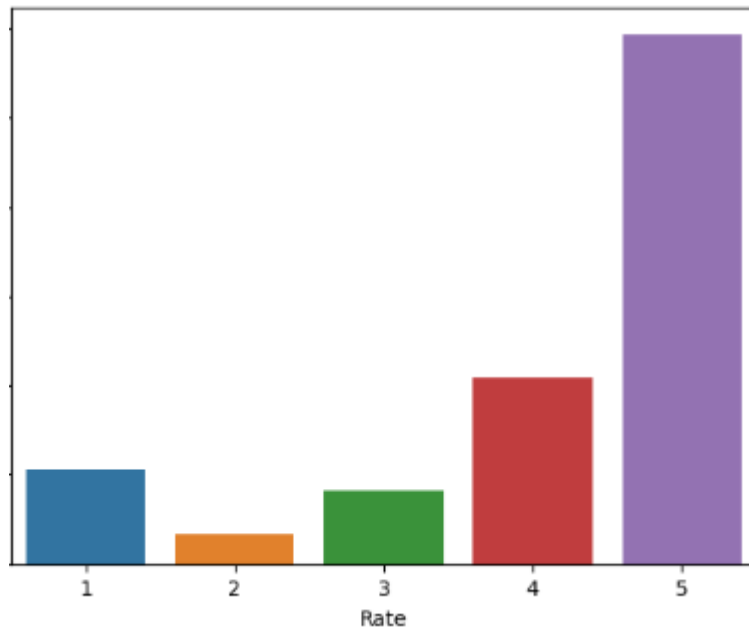
```
array(['5', '3', '1', '4', '2',
       'Pigeon Favourite Electric Kettle?????(1.5 L, Silver, Black)',
       'Bajaj DX 2 L/W Dry Iron',
       'Nova Plus Amaze NI 10 1100 W Dry Iron?Ã\x83Â¿?Ã\x83Â¿(Grey & Turquoise)'],
      dtype=object)
```

```
df = df.drop(index = [175906, 175895, 17299])
```

```
df['Rate'] = df['Rate'].apply(lambda x: int(x))
df['product_price'] = df['product_price'].apply(lambda x: int(x))
```

Задача 2

- Дополняем классы до приемлемого уровня



Задача 3

- Токенизация - это процесс разбиения текста на отдельные единицы, называемые токенами
- Лемматизация - это процесс приведения слова к его базовой форме (лемме)
- Удаляем стоп-слова

```
poor quality plastic material is not good  
poor quality plastic material not good  
5086, 4, 112, 40, 3, 1
```

Задача 3

- Токенизация

```
tokenizer = Tokenizer()
tokenizer.fit_on_texts(texts)

total_words = len(tokenizer.word_index) + 1

# Convert text to sequences

sequences = tokenizer.texts_to_sequences(texts)
```

- Лемматизация и удаление стоп-слов

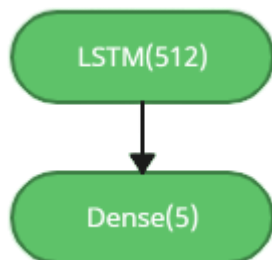
```
tokenized_texts = []
for text in texts:
    tokens = word_tokenize(text)
    # Lemmatize and remove stop words from the text
    filtered_tokens = [lemmatizer.lemmatize(token.lower()) for token in tokens if token.lower() not in stop_words]
    # Append the filtered tokens to the list of tokenized texts
    tokenized_texts.append(filtered_tokens)
texts = [' '.join(tokens) for tokens in tokenized_texts]
```

Задача 4

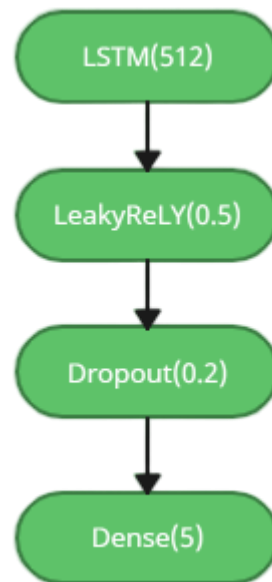
- Строим архитектуры рекуррентных нейронных сетей
- Сравниваем точности архитектур
- Выбираем лучшую

```
model.add(Embedding(total_words, 128, input_length=max_sequence_len))
model.add(LSTM(128, return_sequences=True))
model.add(LeakyReLU(alpha=0.5))
model.add(Dropout(0.5))
model.add(LSTM(128))
model.add(Dense(5, activation='softmax'))
```

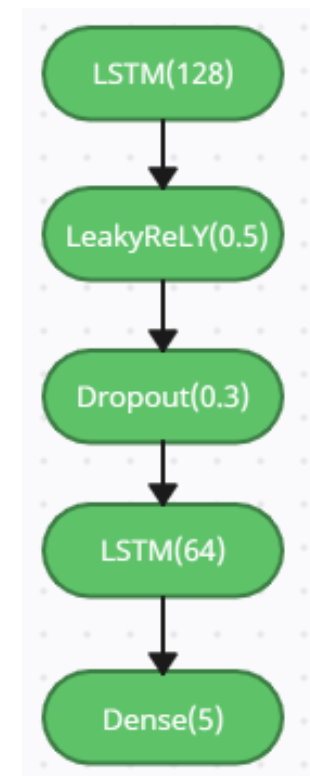
Результаты



Точность(f1-score):
0,54



Точность(f1-score):
0,87



Точность(f1-score):
0,35

Заключение

- Была проведена работа по очистке данных, приведению классов к одному уровню, подготовке текстовых данных к работе с нейронными сетями, построению архитектуры рекуррентных нейронных сетей и анализу их точности
- Было разработано несколько архитектур рекуррентных нейронных сетей с различными параметрами
- Лучше всего себя показала модель из четырёх слоёв

**СПАСИБО ЗА
ВНИМАНИЕ!**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ
РАБОТА
СИНТЕЗ РЕКУРРЕНТНОЙ НЕЙРОННОЙ
СЕТИ ДЛЯ КЛАССИФИКАЦИИ ОТЗЫВА
ПОКУПАТЕЛЯ
ПО ПРОГРАММЕ ПРОФЕССИОНАЛЬНОЙ
ПЕРЕПОДГОТОВКИ:
АНАЛИЗ ДАННЫХ НА RUTRON**

Выполнила: Исхакова Альфия Ильгизовна

Научный руководитель: Семендяев Родион Юрьевич

Г. Санкт-Петербург, 2023