АНАЛИЗ ТОНАЛЬНОСТИ РЕЦЕНЗИЙ НА КИНОПОИСКЕ

Авторы проекта:
1) Борисов Александр
2) Воронцов Александр
3) Рогожин Денис

РАСПРЕДЕЛЕНИЕ ЗАДАЧ

Рогожин Денис

- Предобработка
- Логистическая регрессия
- Word2Vec
- Анализ результатов

Борисов Александр

- Предобработка
- SGDClassifier
- Подбор параметров
- Анализ результатов

Воронцов Александр

- Работа с метриками
- Бейзлайн-модель
- Ансамбли
- Анализ результатов

ОПИСАНИЕ ДАННЫХ

Мы нашли размеченный датасет на кагле с русскоязычными отзывами пользователей Кинопоиска. Они были разбиты на три класса:

- Положительные(~ 80 тыс рецензий)
- Нейтральные(~ 25 тыс рецензий)
- Отрицательные(~ 25 тыс рецензий)

Каждый отзыв был в отдельном текстовом файле, файлы были распределены по папкам в соответствии с типом рецензии.

ПОСТАНОВКА ЗАДАЧИ

Наша задача – классифицировать отзывы по этим трём классам.

ВЫБОР МЕТРИКИ

1) Accuracy score с весами:
   (a1+b2+c3)/(a1+b2+c3+a2+b1+b3+c2+2*c1+2*a3)
2) Balanced accuracy score
3) Precision, Recall, F1-score для каждого класса.

Пример для положительных рецензий:

TP=c3 ; TN=a1+a2+b1+b2

FP=a3+b3 ; FN=c1+c2

	Негативные	**Нейтральные**	**Положительные**
Негативные	a1	a2	a3
Нейтральные	b1	b2	b3
Положительные	c1	c2	c3

ОПИСАНИЕ BASELINE МОДЕЛИ

1) Собрали датасет с полями review и type. В поле type значения 1, 0, -1
2) Далее подали данные методу CountVectorizer с ngram_range=(1, 1) и обучили с помощью SGDClassifier.

На обучение подали 0.7 всех данных, после чего проверили на наших метриках результат на тестовой части(0.3 всех данных)

	Precision	Recall	F1-score
Негативные	0.654852477268	0.598947725729	0.625653754099
Нейтральные	0.3884050081654	0.384999325509	0.386694668382
Положительные	0.8419892392766	0.860245839059	0.85101963746

Balanced_accuracy_score=0.6284155749
Accuracy_score с весами = 0.704784187565

Начальная предобработка данных:

- -Приведение к нижнему регистру
- -**Лемматизация**
- -Удаление цифр
- -Удаление знаков препинания
- -Токенизация
- -Стоп-слова из nltk

МОДЕЛЬ С ЛОГИСТИЧЕСКОЙ РЕГРЕССИЙ.

Для каждой рецензии предсказываем 3 вероятности: P(-1), P(0), P(1).
Изначально пытался выбрать некие границы для этих вероятностей так, чтобы максимизировать Метрику.
Однако наилучший результат вышел, если выбирать класс как argmax(P(-1), P(0), P(1))

	Precision	Recall	F1-score
Негативные	0.7161417322	0.630830587	0.67078454872
Нейтральные	0.4882415820416	0.252139111233	0.332544594102
Положительные	0.811636216544	0.9451619234	0.873324749279

balanced_accuracy_score = 0.6720065102897818
accuracy с коэффицентами= 0.7367515399386518

WORD2VEC.

- word2vec = gensim.models.Word2Vec(window=2,
- min_count=0,
- size=500,
- sample=6e-5,
- alpha=0.03,
- min_alpha=0.0007,
- negative=20)

	Precision	Recall	F1-score
Негативные	0.5188668	0.7653892838	0.6184671430
Нейтральные	0.48450057405	0.116478056858	0.1878059635
Положительные	0.816424094337	0.90500490677	0.8584354172

accuracy с весами = 0.6845006406691971
balanced_accuracy_score=0.6065971613180282

Предобработка данных:

- -Приведение к нижнему регистру
- -Удаление цифр
- -Удаление знаков препинания
- -Замена ‘ё’ на ‘е’
- -Лемматизация
- -Токенизация
- **-Создание списка стоп-слов**

SGDCLASSIFIER.

```
TfidfVectorizer(
  lowercase=True, ngram_range=(1,3), analyzer='word', token_pattern='[а-яё]+',norm='l2',
    stop_words=bad_tokens, min_df=10, max_df=0.8)
```

```
SGDClassifier(random_state=SEED, loss='modified_huber', class_weight='balanced')
```

	Precision	Recall	F1-score
Негативные	0.65670635812	0.76040582726	0.70476190476
Нейтральные	0.5104306864064	0.31545019	0.38992417427
Положительные	0.8519412151787	0.913559920028	0.881675273821

balanced_accuracy_score=0.6730260865689993
accuracy с коэффф. = 0.751118631321677

РЕЗУЛЬТАТЫ

В итоге решили добавить цифры в рецензии и результат сразу улучшился! Нейтральные рецензии стали распознаваться намного лучше.

Итоговые результаты:

	Precision	Recall	F1-score
Негативные	0.714087546	0.80336396740	0.7560995512
Нейтральные	0.59414955443	0.42326800993	0.4943584784
Положительные	0.8787540055	0.92577036310	0.90164968554

accuracy с весами = 0.7936475721751484
balanced_accuracy_score=0.7289970353966817

СПАСИБО ЗА ВНИМАНИЕ!