Правительство Российской Федерации
Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет «Высшая школа экономики»

Факультет компьютерных наук
Образовательная программа бакалавриата 09.03.04 «Программная инженерия»

ОТЧЕТ
по производственной практике

на факультете компьютерных наук НИУ ВШЭ

Выполнила студентка
группы БПИ182
Антонова А.Т.

_____
(подпись)

Руководитель практики

Департамент больших данных и информационного поиска, доцент

Сухорослов Олег Викторович

Дата _____ _____ _____
(оценка) (подпись)

Москва – 2021

Содержание

Цель и задачи практики

Цель прохождения практики:

Решить задачу классификации текстовых сообщений с использованием алгоритмов машинного обучения и искусственного интеллекта.

Модель машинного обучения должна классифицировать тексты сообщений, новостей, постов в социальных сетях на две категории: относящиеся к теме финансовая новость или совет (financial news/advice) или нет. Затем необходимо провести анализ тональности текста для тех текстов, которые подходят под тему финансовая новость или совет.

Задачи практики:

1. Подготовить датасет для обучения
2. Изучить подходы и библиотеки для решения задачи
3. Обучить модель
4. Проанализировать результаты и при необходимости изучить альтернативные решения

Обзор изученных материалов, источников, аналогов, технологий, методов

Готовой обученной модели, в точности решающей данную задачу, найдено не было. Поэтому необходимо было определиться с базовой моделью для классификации и подготовить датасет для ее обучения.

Первая задача — бинарная классификация текстов. Подходов к классификации текстов много: логистическая регрессия, SVM (Support Vector Machine — метод опорных векторов), ансамблевые методы на основе решающих деревьев, различные модели глубокого обучения на основе нейронных сетей. Однако для работы был выбран BERT — модель на основе трансформеров.

BERT (Bidirectional Encoder Representations from Transformers) — нейронная сеть, разработанная компанией Google, результаты которой на ряде задач обработки естественного языка превосходят остальные модели с большим отрывом. В отличие от прежних классических языковых моделей, BERT обучает контексто-зависимые представления, причем двусторонние, что позволяет модели лучше понимать смысл многозначных слов.

Обучение модели BERT происходит в два этапа: предобучение (pre-training) и точная настройка (fine-tuning). При подаче на вход текста происходит его токенизация на основе предопределенного словаря. Предобучение состоит из двух задач: предсказания следующего предложения и генерации пропущенного токена. На вход BERT подаются токенизированные пары предложений, в которых некоторые токены скрыты. Таким образом, благодаря маскированию токенов, сеть обучается глубокому двунаправленному представлению языка, учится понимать контекст предложения. Задача же предсказания следующего предложения есть задача бинарной классификации — является ли второе предложение продолжением первого. Благодаря ей сеть можно обучить различать наличие связи между предложениями в тексте. Этапа предобучения можно интерпретировать как обучение модели языку. Этап точной настройки обучает модель решать уже конкретную задачу на конкретных данных, в данном случае — задачу бинарной классификации [1].

В библиотеке Transformers от компании Hugging Face представлены предобученные модели BERT для различных задач [2], они и были использованы для дальнейшей работы.

Следующей задачей был поиск датасетов для этапа точной настройки модели. Датасеты были найдены на следующих ресурсах: Kaggle (платформа для онлайн-соревнований по анализу данных), GitHub (платформа для контроля версий и публикации проектов).

Второй большой задачей является анализ эмоциональной окраски для тех текстов, которые подходят под тему финансовая новость или совет. Эту задачу тоже хорошо решает BERT. Можно было бы самостоятельно настроить предобученную модель на данных финансовых новостей, которые размечены по эмоциональной окраске, но такая модель уже существует и настроена — это модель FinBERT [3].

Описание методов, алгоритмов, моделей, технологий, средств разработки, использованных для решения поставленных задач

Данные, обработанные данные, код для обучения модели и ее применения доступны в репозитории на GitHub (https://github.com/atantonova/summer_practice)

Поиск и подготовка датасета для обучения и тестирования классификатора.

Требования к датасету: наличие полей "text" — текст сообщения или новости, "finance" — 1, если сообщение/новость является "financial news/advice", 0 — в противном случае.

1. Категории новостей

Первый датасет, который был использован, — News Category Dataset [4]. Он содержит сведения о новостях в интернет-газете HuffPost: заголовок, краткое описание, категория, а также дата публикации, автор и другие данные. В числе категорий были выделены money и business, как наиболее подходящие к теме финансов. Сформированный датасет содержал в поле "text" краткое описание новости (так как заголовок часто может не отражать суть статьи), в поле "finance" — 1, если новость относится к категориям money или business, 0 — в противном случае. Затем из него были удалены дубликаты по полю "text".

Так как новостей категорий не money или business было больше в десятки раз, были подготовлены сбалансированный и несбалансированный датасеты. Несбалансированный содержал 20% от всех новостей, не относящихся к категориям money или business, и все новости, относящиеся к ним, 40888 записей, процентное соотношение 92% - 0 и 8% - 1. Сбалансированный содержал равное количество новостей, относящихся и не относящихся к категориям money или business, 6750 записей.

2. Твиты и комментарии на Reddit, датасеты для анализа эмоциональной окраски

Следующие использованные датасеты — это объединение:

- Sentiment Analysis for Financial News [5] — датасет для анализа эмоциональной окраски на заголовках финансовых новостей.

- Sentiment Analysis on Financial Tweets [6] — датасет для эмоциональной окраски на твитах, относящихся к теме финансов. Твиты были собраны с аккаунтов финансовых компаний, интернет-изданий с финансовыми новостями, их обозревателей и других.
- Twitter and Reddit Sentimental analysis Dataset [7] — датасет для анализа эмоциональной окраски на твитах и постах на портале Reddit без конкретной темы сообщений. Данные содержат твиты и комментарии, связанные с обсуждением выборов в Индии.

Все три датасета изначально предлагаются для обучения модели анализа эмоциональной окраски сообщений, так как это задача, которая хорошо решается такого типа данных. Каждый из них содержит поля текста сообщения и индикатора эмоциональной окраски. Для решения задачи классификации необходим только текст.

Разметка датасетов строится на предположении, что в датасете Twitter and Reddit Sentimental analysis Dataset содержится незначительное количество данных, напрямую связанных с финансами. Оно было сделано после просмотра нескольких десятков текстов в случайном порядке, преобладают обсуждения религии, политики и личного отношения к тем или иным политикам; экономика встречалась редко. Поэтому тексты из этого датасета помечаются 0, из других двух, тема которых как раз финансы, — 1.

Для получения модели с хорошим результатом из эти трех датасетов и датасета категорий новостей были составлены следующие четыре:

1. Sentiment Analysis for Financial News [5] и Twitter and Reddit Sentimental analysis Dataset [7], данные из Twitter — данные для обучения модели *(тренировочный датасет - 1)*
2. Sentiment Analysis for Financial News [5], Twitter and Reddit Sentimental analysis Dataset [7], данные из Twitter и News Category Dataset [4] (краткие описания новостей категорий, не относящиеся к money, business, politics; размечены как не относящиеся к теме финансов)— другие данные для обучения модели *(тренировочный датасет - 2)*
3. Sentiment Analysis for Financial Tweets [6] и Twitter and Reddit Sentimental analysis Dataset [7], данные из Reddit — данные для тестирования модели после ее обучения *(тестовый датасет - 1)*
4. Sentiment Analysis for Financial Tweets [6] и News Category Dataset [4] без категорий business и money— другие данные для тестирования модели после ее обучения *(тестовый датасет - 2)*

3. Выявление ключевых слов

Еще один подход к решению задачи разметки датасета — поиск ключевых слов и разметка данных по наличию/отсутствию ключевых слов в тексте.

Была использована библиотека KeyBERT [8], также основанная на алгоритме BERT. С ее помощью можно выделить ключевые слова или словосочетания в тексте или массиве текстов, исключая служебные части речи, местоимения и т.д. Также можно получить значимость наиболее часто встречающихся ключевых слов, настроить "разнообразность" (diversity) выделяемых слов и их количество.

Данный алгоритм с разными настройками diversity был применен к очищенному датасету Sentiment Analysis for Financial News [5]. Полученные ключевые слова в целом удовлетворяли теме финансов. Но на том же датасете, где все тексты относятся к теме финансов, с помощью 100 ключевых слов с высокой diversity выделялось около 10% текстов. Дальнейшее увеличение количества ключевых слов привело бы к бóльшему количеству ложных срабатываний, поэтому данный подход не оправдался.

Очистка данных

Текстовые данные были обработаны с помощью регулярных выражений следующим образом: удалены ссылки, теги и хэштеги (например, #finance, @NYT), удалены цифры, знаки препинания, лишние пробелы, все слова приведены к нижнему регистру.

Код для подготовки датасетов и очистки данных доступен в репозитории (./code/data_check.ipynb). Данные, в необработанном виде и подготовленные к обучению, также доступны в репозитории (./data).

Обучение модели

Код для обучения классификатора доступен в репозитории: ./code/message_classification.ipynb.

1. Категории новостей

В разделе "News category classification" находится код для обучения на первом датасете с краткими описаниями новостей. Для точной настройки использовался предобученный токенизатор BERT "bert-base-uncased" для английского языка без учета регистра и модель с таким же идентификатором. Эта модель базовая, поэтому быстро настраивается и не требует больших ресурсов для вычислений. Далее использован класс BertForSequenceClassification, который позволяет обучить модель для классификации, в данном случае бинарной.

Библиотека transformers предоставляет удобный функционал для настройки параметров обучения модели. Так можно настроить количество эпох обучения, размер батча, коэффициент скорости обучения, директории для логгирования и сохранения модели через определенное количество шагов оптимизатора, метрики для логгирования и другие параметры. В понимании этого функционала помогла статья, где разбирается пример классификации новостей с помощью этой библиотеки [9].

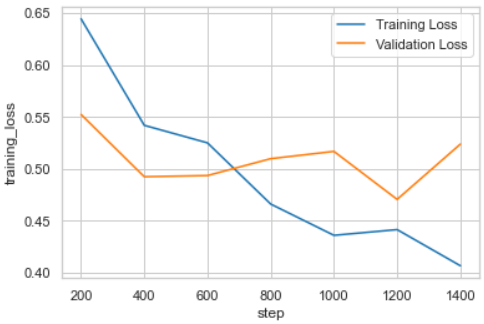

Рисунок 1. График loss-функции для тренировочного и тестового датасетов на сбалансированных данных

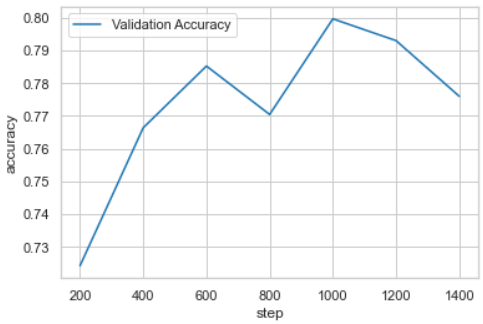

Рисунок 2. График метрики точности для тестового датасета на сбалансированных данных

Precision	0.79
Recall	0.81
F1 score	0.80
Accuracy	0.79

Таблица 1. Метрики для модели, обученной на сбалансированном датасете категорий новостей

На рисунках 1 и 2 показаны графики loss-функции и точности на тестовом датасете для сбалансированных данных из датасета с категориями новостей. Видно, что после 1200 шага модель переобучилась, максимальная точность на тестовых данных из того же датасета около 80%. В таблице 1 указаны ключевые метрики качества модели, можно сделать вывод, что ложно-положительных и ложно-отрицательных срабатываний одинаково около 20%. Учитывая то, что реальные данные, на которых должна работать модель, отличаются от этих и что разметка этих данных неточная, 80% — это недостаточная точность.

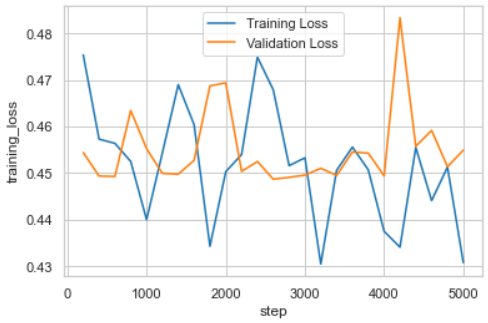

Рисунок 3. График loss-функции для тренировочного и тестового датасетов на несбалансированных данных

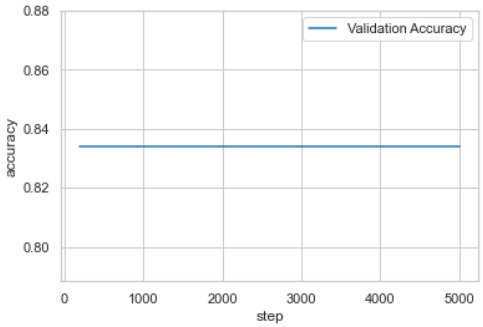

Рисунок 4. График loss-функции для тренировочного и тестового датасетов на несбалансированных данных

На рисунках 3 и 4 аналогичные графики для несбалансированных данных. Такой график loss-функции говорит о переобучении модели, точность почти не улучшается, поэтому обучение было остановлено. Точность не меняется и остается на уровне 84%, но так как это несбалансированный датасет, понятно, что взвешенная точность хуже в сравнении с предыдущей моделью.

Обучение на данных по кратким описаниям новостей не дало значимого результата.

2. Твиты и комментарии на Reddit, датасеты для анализа эмоциональной окраски

В разделе "BERT model for fine-tuning" находится код для этапа точной настройки модели. Функция *train_classification_bert* может быть вызвана на любом подходящем под формат датасете. Она использует в качестве основной метрики точность (accuracy), тренирует на 2 эпохах, сохраняет модель после каждых 700 шагов.

Эта функция применена на данных для обучения модели.

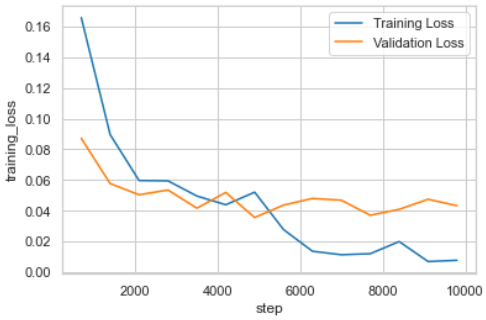

Рисунок 5. График loss-функции для тренировочного и тестового датасетов (tweets)

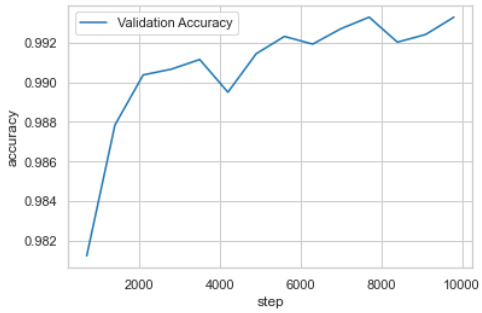

Рисунок 6. График loss-функции для тренировочного и тестового датасетов (tweets)

На рисунках 5 и 6 приведены графики обучения модели на тренировочном датасете 2 (см. стр. 5). По графику loss-функции видно, что после 5600 шага оптимизатора модель переобучилась, точность после него повышается незначительно, поэтому модель, сохранена после этого шага как классификатор-2.

Аналогично проходило обучение модель на тренировочном датасете 1, модель переобучилась после 2100 шага, модель после него была сохранена как классификатор-1.

	Классификатор-1, метрики для тестовых данных-1	Классификатор-1, метрики для тестовых данных-2	Классификатор-2, метрики для тестовых данных-2
Precision for class 0	0.96	0.89	0.74
Recall for class 0	0.81	0.31	0.98
F1 score for class 0	0.88	0.46	0.84
Precision for class 1	0.84	0.57	0.97
Recall for class 1	0.97	0.96	0.6
F1 score for class 1	0.90	0.72	0.74
Accuracy	0.89	0.63	0.81

Таблица 2. Метрики для классификаторов, обученных на данных 2

В таблице 2 представлены метрики для классификаторов. Класс 0 — это класс текстов, не относящихся к теме финансов, класс 1 — относящихся к ней. Precision for class 0 измеряет точность модели при предсказании класса 0, отражает количество ложноположительных срабатываний. Recall for class 0 — сколько текстов класса 0 было предсказано верно. F1 score for class 0 отражает баланс между precision и recall. Аналогично для класса 1.

Из этих метрик можно сделать следующие выводы:

- Классификатор-1 на тестовых данных, похожих на его тренировочные данные, точно распознает тексты, не относящиеся к финансам, но пропускает около 20% из них. Он причисляет к классу 1 больше текстов, чем нужно, но пропускает только 3% относящихся к финансам. Тестовый датасет - 2 как раз и был сделан из-за того, что данные для тестирования похожи на тренировочные, поэтому не отражали реальную точность модели.
- Классификатор-1 на тестовых данных, куда добавлены описания новостей, не пропускает тексты класса 1, но около 70% текстов класса 0 относит к классу 1.
- Таким образом, классификатор-1 не пропускает тексты, относящиеся к финансам, но причисляет к этой теме достаточно много текстов, к ней не относящиеся. Может быть полезен, когда необходимо с высокой вероятностью найти все тексты по финансам, несмотря на ложноположительные срабатывания.

- Классификатор-2 был обучен на данных, куда добавлены краткие описания новостей, не относящихся к теме финансов, чтобы разнообразить тексты класса 0. Он точно распознает тексты, относящиеся к финансам, но много из них пропускает. При этом он мало пропускает тексты класса 0.
- По метрике F1 score результат классификатора-2 лучше, чем классификатора-1. То есть, баланс между precision и recall ближе к идеальному у второго, а также выше точность.
- Таким образом, классификатор-2 может быть полезен, если нужно найти тексты, которые точно относятся к теме финансов, имея в виду, что многие менее очевидные будут пропущены.

Оба классификатора доступны на Google Drive для скачивания, ссылка на них есть в файле readme.md в репозитории проекта.

FinBERT

Модель, использованная для анализа эмоциональной окраски текста основана на предобученной модели BERT, она была настроена на датасете Sentiment Analysis for Financial News [5] и размещена в открытом доступе на Hugging Face для использования [3].

Модель выдает вероятности того, что текст является положительно, нейтрально и негативно окрашенным. Например, текст *"on wednesday he announced that the government would propose granting a licence for two reactors one to be built by fennovoima and the other by tvo"* с вероятностью 75% имеет положительную окраску, 23% — нейтральную и 8% негативную.

Описание полученных результатов

Конечным результатом работы можно считать модели, настроенные на классификацию текстов по принадлежности к теме финансов, а также код для их использования, который находится в файле ./code/predict.ipynb. В нем содержатся функции, описанные в таблице 3.

Функция	**Описание**	**Параметры**	**Возвращаемые значения**
`load_model_tokenizer`	Функция для загрузки классификатора и токенизатора	• ***path_to_classifier*** : str — путь в файловой системе к сохраненной настроенной модели • ***path_to_tokenizer*** : str — путь в файловой системе к сохраненному настроенному токенизатору или указание идентификатора для поиска в библиотеке (например, "bert-base-uncased")	• ***classifier***: BertForSequenceClassification — классификатор • ***tokenizer_classifier*** : BertTokenizerFast — токенизатор • model_sent : AutoModelForSequenceClassification — модель для предсказания эмоциональной окраски текста, • ***tokenizer_sent*** : AutoTokenizer — токенизатор для предсказания эмоциональной окраски текста
`clean_data`	Функция для очистки данных: удаление стоп-слов, знаков препинания и т.д., приведение к нижнему регистру	• ***test_data*** : DataFrame — датасет для классификации и определения эмоциональной окраски, 1 столбец: "text" с текстом сообщения	• ***test_data*** : DataFrame — очищенные данные
`predict_with_sentiment`	Функция для для классификации и определения эмоциональной окраски текстов	• ***test_data*** : DataFrame — очищенные данные • ***classifier***: BertForSequenceClassification — классификатор • ***tokenizer_classifier*** : BertTokenizerFast — токенизатор • ***model_sent*** : AutoModelForSequenceClassification — модель для предсказания эмоциональной окраски текста, • ***tokenizer_sent*** : AutoTokenizer — токенизатор для предсказания эмоциональной окраски текста	• ***results*** : DataFrame — результат классификации и определения эмоциональной окраски текстов, столбцы: - "*text*" — текст, - "*finance_proba*" — вероятность того, что текст относится к теме финаносового совета или новости, - "*positive*" — вероятность позитивной окраски текста, - "*neutral*" — вероятность нейтральной окраски текста, - "*negative*" — вероятность негативной окраски текста

Таблица 3. Описание функций файла predict.ipynb

```
classifier, tokenizer_classifier, model_sent, tokenizer_sent =
load_model_tokenizer(path_to_classifier, path_to_tokenizer)
test_data = pd.read_csv(path_to_test_data)
test_data = clean_data(test_data)
results = predict_with_sentiment(test_data, classifier,
tokenizer_classifier, model_sent, tokenizer_sent)
```

Выше приведен пример использования этих функций для получения результатов классификации и анализа эмоциональной окраски текстов.

Здесь:

- ***path_to_classifier***: string — путь к директории с сохраненной моделью в формате, который может быть распознан функцией BertForSequenceClassification.from_pretrained
- ***path_to_tokenizer***: string — путь к директории с токенизатором или его идентификатор, при обучении и тестировании был использован предобученный с идентификатором *'bert-base-uncased'*
- ***test_data***: DataFrame — тестовые данные, 1 столбец: 'text'
- ***results***: DataFrame — результат работы классификатора и анализа эмоциональной окраски текстов, столбцы: 'text', 'finance_proba', 'positive', 'neutral', 'negative'. Пример на рисунке 7.

	text	finance_proba	positive	neutral	negative
9634	ted is the only possibility to stop trump says...	0.995019	0.054270	0.900015	0.045715
9648	britain is currently only launching strikes ag...	0.000156	0.040631	0.449064	0.510305
106	in fiskars cash flow from operating activities...	0.995056	0.950055	0.023019	0.026926
1446	ulefos group is the leading supplier of manhol...	0.995006	0.711260	0.282133	0.006607
1513	currently yit builds a housing estate zapadnye...	0.995043	0.086874	0.903513	0.009613
...	...	...	...	...	...
7684	a defeat in saturdays election would keep the ...	0.994883	0.067182	0.295905	0.636912
4960	she leads among those who say they arent plann...	0.005780	0.060370	0.770263	0.169367
2766	activities range from the development of natur...	0.995047	0.035639	0.949661	0.014700
7664	hillary clinton looks at this through the lens...	0.995009	0.071151	0.914571	0.014279
416	alexandria va march pertti salmi and hanna vuo...	0.994907	0.063330	0.919846	0.016824

Рисунок 7. Пример результата работы классификатора и анализа эмоциональной окраски текстов

Заключение

В результате работы над проектом были изучены алгоритмы анализа естественного языка на основе трансформеров, а именно модель BERT и ее применение к задачам классификации, выделения ключевых слов и анализа эмоциональной окраски текста. Были проведены поиск и подготовка датасетов для обучения и тестирования моделей классификатора, применены методы очистки данных, составлены несколько разных датасетов для тестирования для более точного анализа работы модели.

Были обучены три модели, две из которых показали хорошие результаты. Первая модель не пропускает тексты, относящиеся к финансам, но причисляет к этой теме достаточно много текстов, к ней не относящиеся, она быть полезна, когда необходимо с высокой вероятностью найти все тексты по финансам, несмотря на ложноположительные срабатывания. Вторая точнее отсеивает тексты, не относящиеся к финансам, но может пропускать некоторое количество текстов, относящихся к ней, может быть полезна, если нужно найти тексты,

которые с большой долей вероятности относятся к теме финансов, имея в виду, что многие менее очевидные будут пропущены. Максимальная достигнутая точность классификации — 81%.

С помощью разработанных функций можно получить классификацию текстов по принадлежности к теме финансовых новостей или советов, а также проанализировать их эмоциональную окраску.

В дальнейшем можно обучать модели на новых данных с помощью написанных в ходе проекта функций, анализировать порог вероятности принадлежности текста к теме финансов и, в зависимости от требований, повышать или понижать приемлемую вероятность.

Список использованных источников

1. Transformers Documentation [Электронный ресурс] / Hugging Face. Режим доступа: https://huggingface.co/transformers/index.html , свободный (дата обращения: 01.07.2021)
2. BERT (языковая модель) [Электронный ресурс] / Университет ИТМО. Режим доступа: https://neerc.ifmo.ru/wiki/index.php?title=BERT_(языковая_модель) , свободный (дата обращения: 01.07.2021)
3. FinBERT [Электронный ресурс] / GitHub. Режим доступа: https://github.com/ProsusAI/finBERT , свободный (дата обращения: 01.07.2021)
4. News Category Dataset [Электронный ресурс] / Kaggle. Режим доступа: https://www.kaggle.com/rmisra/news-category-dataset , свободный (дата обращения: 01.07.2021)
5. Sentiment Analysis for Financial News [Электронный ресурс] / Kaggle. Режим доступа: https://www.kaggle.com/ankurzing/sentiment-analysis-for-financial-news , свободный (дата обращения: 01.07.2021)
6. Sentiment Analysis for Financial Tweets [Электронный ресурс] / Kaggle. Режим доступа: https://www.kaggle.com/vivekrathi055/sentiment-analysis-on-financial-tweets , свободный (дата обращения: 01.07.2021)
7. Twitter and Reddit Sentimental analysis Dataset [Электронный ресурс] / Kaggle. Режим доступа: https://www.kaggle.com/cosmos98/twitter-and-reddit-sentimental-analysis-dataset , свободный (дата обращения: 01.07.2021)
8. KeyBERT [Электронный ресурс] / GitHub. Режим доступа: https://github.com/MaartenGr/KeyBERT , свободный (дата обращения: 01.07.2021)
9. How to Fine Tune BERT for Text Classification using Transformers in Python [Электронный ресурс] / PythonCode. Режим доступа: https://www.thepythoncode.com/article/finetuning-bert-using-huggingface-transformers-python , свободный (дата обращения: 01.07.2021)

Рабочий план-график прохождения практики с отметками о выполнении

№ п/п	Сроки проведения	Планируемые работы	Отметка о выполнении
1	01.07.2021	Инструктаж по ознакомлению с требованиями охраны труда, техники безопасности, пожарной безопасности, а также правилами внутреннего трудового распорядка	+
2	01.07.2021 — 02.07.2021	Изучение технологий, методов для решения задач практики	+
3	03.07.2021 — 20.07.2021	Выполнение задания практики	+
4	20.07.2021	Представление результатов работы	+
5	21.07.2021	Подготовка отчета по практике	+

Приложения

Словарь терминов

Датасет — это обработанная и структурированная информация в табличном виде. Строки такой таблицы называются объектами, а столбцы – признаками.

Разметка — шаг в разработке датасета, когда человек решает задачу, которую в дальнейшем будет решать искусственный интеллект. В данном случае это определение принадлежности текста к теме финансовых новостей или советов.

Твит — запись в социальной сети Twitter.

Точность (accuracy) — отношение верно предсказанных значений к их общему числу, является одной из метрик качества модели. В тексте под этим термином имеется в виду именно accuracy, а не precision.