МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ М.В.ЛОМОНОСОВА
ЭКОНОМИЧЕСКИЙ ФАКУЛЬТЕТ
Направление Экономика
Программа Анализ данных в экономике

Магистерская диссертация

«Использование нейронных сетей для прогнозирования стоимости акций на основе новостных данных»

Выполнил студент:
Борисенко Георгий Александрович

Научный руководитель:
Андрейцев Антон Игоревич

Москва 2023

Аннотация

Данная работа посвящена прогнозированию стоимости акций крупных российских компаний, торгующихся на Московской бирже, на основе новостей. В качестве моделей для прогноза используются нейронные сети трансформеры. Более того, в анализе участвуют и классические методы машинного обучения для сравнения с нейросетевым подходом. В качестве новостных данных используются крупные российские новостные источники и Телеграмм-каналы. Также производится сравнение моделей, обученных на разных источниках.

В результате исследования получено, что классические методы машинного обучения справляются лучше с данной задачей в общем случае, но нейросети также показывают хорошее качество. Также в работе даются рекомендации по выбору источника новостей и выбора постановки задачи.

JEL-коды: C63, G14

Оглавление

Введение 5

Глава 1. Теоретические и практические основы прогнозирования цен акций на основе текстовых данных. 8

1.1 Обзор литературы 8

1.2 Выводы из обзора литературы. 13

2 Глава. Методы для прогнозирования изменения стоимости акций на основе текстовых данных. 16

2.1 Случайный лес 16

2.2 Градиентный бустинг 17

2.3 Эмбеддинги 17

2.3.1 TF-IDF 18

2.3.2 Нейросетевые Эмбеддинги 19

2.4 Трансформеры 20

3 Данные 22

3.1 Классические новости 22

3.2 Новости из Телеграмма 23

3.3 Предобработка текстов 26

3.4 Выбор ценных бумаг 27

3.5 Получение данных о стоимости ценных бумаг и создание целевой переменной. 28

3.6 Разметка данных с помощью регулярных выражений 29

4 Построение Бейзлайн-моделей 32

4.1 Результаты для Телеграмма 33

4.1.1 Задача классификации на 2 класса 33

4.1.2 Задача классификации на 3 класса 34

4.2 Результаты для Традиционных источников новостей 35

4.2.1 Задача классификации на 2 класса 36

4.2.2 Задача классификации на 3 класса 36

4.4 Итоги построения бейзлайн моделей. 37

5 Нейросетевой подход 39

5.1 Результаты для Телеграмма 41

5.1.1 Задача классификации на 2 класса 41

5.1.2 Задача классификации на 3 класса 42

5.2 Результаты для классических новостей 43

5.2.1 Задача классификации на 2 класса 43

5.2.2 Задача классификации на 3 класса 44

5.3 Итоговые результаты для нейросетевого подхода 45

5.4 Интерпретация предсказаний нейросети 46

6 Заключение 49

7 Список литературы *50*

8 Приложения *52*

Приложение 1 ..... **52**

Приложение 2 ..... **53**

Приложение 3 ..... **54**

Приложение 4 ..... **56**

Приложение 5 ..... **58**

Приложение 6 ..... **60**

Приложение 7 ..... **61**

Приложение 8 ..... **63**

Приложение 9 ..... **65**

Приложение 10 ..... **66**

Введение

Согласно теории эффективного рынка [Fama et al., 1969][1], капитализация публичных компаний зависит от событий, которые происходят вокруг нее. Соответственно, имея доступ к этой информации, можно делать предсказания о стоимости акций компании в будущем.

Наиболее доступным источником информации о компаниях являются публикации крупных новостных изданий, однако информация там публикуется с задержкой. Чтобы преодолеть этот недостаток можно воспользоваться новостями из социальных сетей, в частности из Телеграмма, где публикация новостей происходит максимально быстро. Таким образом объектом исследования моей магистерской диссертации являются новости о публичных компаниях, а предметом — их взаимосвязь с движением цен акций компаний, к которым они относятся.

На данный момент лучше всего справляются с задачей нейролингвистического программирования современные архитектуры нейронных сетей, так как они способны агрегировать в себе огромное число выученных взаимосвязей о имеющейся информации, зачастую они способны даже превосходить результат человека. Таким образом, «ключ» к рынку может быть найден с помощью глубокого семантического анализа полученных новостей.

Цель данной работы состоит в построении моделей машинного обучения, способных предсказывать направление движения стоимости акций голубых фишек на Московской фондовой бирже, а также в выявлении слов и словосочетаний, которые оказывают влияние на эту динамику. В качестве моделей будут использоваться классические методы машинного обучения, так и нейросетевые подходы, с целью сравнения качества их прогнозов. Более того, будет проведен сравнительный анализ предсказательной силы моделей для разных входных данных: традиционных новостей из крупных новостных ресурсов и новостей из Телеграмм каналов. Также задача будет решена в двух постановках: классификация новостей на положительный/отрицательный класс и положительный/нейтральный/отрицательный.

Актуальность данной работы состоит в том, что полученные модели позволят снизить неопределённость в прогнозировании «случайного» процесса. С практической точки зрения результаты данной работы могут быть полезны крупным

[1] Fama E. F. et al. The adjustment of stock prices to new information //International economic review. – 1969. – Т. 10. – №. 1. – С. 1-21.

инвестиционным фондам, которые решили изменить свою позицию по ценной бумаге, однако они не могут сделать это быстро, если объем позиции слишком большой. В таком случае трейдерам компании приходится совершать операции на рынке на протяжении некоторого промежутка времени. Новостной фон о компании за это время может изменяться, в таком случае модель сможет подсказать трейдерам будущее направление движения цены актива на некотором временном горизонте, и они смогут скорректировать свои действия по изменению позиции (то есть выбрать, какой объем актива торговать сейчас, а какой — позже).

На основе полученной модели также может быть создан торговый робот, который будет вовремя устранять арбитражные возможности на рынке и увеличивать его ликвидность, что полезно для финансовой системы в целом.

Также в публичном доступе работы, которые использовали нейросети для прогнозирования движений акций на Московской бирже на основе новостных данных найдено не было. В зарубежной литературе подобная задача уже решалась, однако, в качестве источников данных использовалось только что-то одно: новости из крупных новостных источников или социальных сетей. В моей же работе проводится сравнение источников. Более того, задача классификации новостей на 3 класса в найденных мною исследованиях не рассматривалась, в моей же работе присутствует анализ сразу двух подходов к решению: классификации на 2 и 3 класса.

С теоретической точки зрения данная работа актуальна тем, что нейросетевые методы прогнозирования развиваются очень быстро. Как правило, они тестируются на стандартных наборах данных, на которых с годами показывают все более высокое качество, однако в реальных задачах это может быть не так. Поэтому необходимо проверять существующие архитектуры нейронных сетей на отдельно взятой задаче.

В рамках данной работы решаются следующие задачи:

1. Выбор акций на Московской бирже, для которых будет осуществляться анализ;
2. Создание и применение парсинговых программ для сбора новостей с сайтов крупных новостных изданий и Телеграмм-каналов;
3. Обзор литературы по тематике работы;
4. Преобразование данных к виду, пригодному для анализа методами классического ML и нейросетевого анализа;
5. Построение моделей классического машинного обучения;

6. Построение нейросетевых моделей на основе архитектуры Трансформера;
7. Обучение моделей;
8. Выбор оптимальной модели по качеству прогноза;
9. Сравнительный анализ полученных результатов моделей для новостных изданий и Телеграмм-каналов;
10. Интерпретация полученных нейросетью результатов.

Глава 1. Теоретические и практические основы прогнозирования цен акций на основе текстовых данных.

1.1 Обзор литературы

Одна из статей, цель которой — предсказание стоимости акции на основе новостного фона с помощью нейростетевого анализа, является [Li Y., Pan Y. 2021][2].

Авторы утверждают, что стоимость акции зависит от спроса и предложения на ценную бумагу. Они же, в свою очередь, зависят от множества различных факторов, которые можно определить из финансовых новостей, официальных публичных писем компаний, заявлений топ-менеджмента или финансовых отчетов.

Также авторы считают, что подбор определенного временного окна очень важен для построения точного прогноза. Это связано с тем, что цену на активы определяют непосредственно участники рынка, то есть люди, а человеческой памяти с течением времени свойственно все забывать. Поэтому, если рассматривать слишком длинное временное окно, длиннее, чем человеческая память, то в модели слишком давние события будут негативно влиять на качества предсказаний. Аналогично и для окна короче, чем человеческая память.

В отличие от прошлых работ по тематике, авторы учитывают информацию не только о компании, но и информацию о ее прямых конкурентах в отрасли. Таким образом, можно учитывать не только прямое влияние новостей на поведение акций компании, но и косвенное.

В качестве новостных данных авторы используют крупнейшие новостные издания США. Однако они используют не весь текст интересующих статей, а только заголовки, так как по их заверениям использование полного текста может привести к сложностям при его анализе из-за возможного наличия шумовых слов, мешающих модели оценить его семантику. В качестве целевой метки используются данные по компаниям из индекса S&P 500, стоимость ценной бумаги берется по закрытию торгового дня. Так как биржа не работает по выходным, то новости, вышедшие в выходной день, никак не учитываются.

Подготовка текстовых данных происходит с помощью Aware Dictionary and Sentiment Reasoner (VADER). Данная модель преобразовывает текстовые данные статьи в числовую характеристику, которая позволяет судить о степени позитивности

[2] Li Y., Pan Y. A novel ensemble deep learning model for stock prediction based on stock prices and news //International Journal of Data Science and Analytics

или негативности новости. Данные по ценам акций авторы переформатировали в интервал от 0 до 1, чтобы избежать переобучения и увеличить точность.

Построенная модель машинного обучения представляет из себя комбинацию из нескольких рекуррентных нейронных сетей (RNN): нейросети долгой краткосрочной памяти (LSTM), управляемой рекуррентной нейронной сети (GRU) и полносвязная нейронная сеть. LSTM модель позволяет разделять новости про продолжительности их влияния, в то время как GRU модель имеет меньшее время вычислений. В зависимости от ситуации одна модель может быть лучше другой, поэтому необходима их комбинация.

Под комбинацией моделей понимается их параллельное обучение и получение предсказаний для временного ряда. Затем эти предсказания используются в качестве мета-признаков для обучения полносвязной нейронной сети, которая уже и будет давать конечный ответ. Настройка параметров для нейронной сети происходит с помощью разбиения на обучающую и валидационную выборки, причем выборки строго упорядочены по времени, то есть все наблюдения обучающей выборки идут до всех наблюдений валидационной выборки. Настройки требуют такие параметры как: количество слоев в нейросети, количество нейронов на каждом слое, параметр регуляризации (drop out) и количество эпох.

Настройка параметров происходила с помощью использования следующих метрик: среднее квадратичное отклонение (MSE), матрица ошибок (confusion matrix), средняя точность предсказания (MPA) и точность направления движения (MDA).

Как мне кажется, в статье авторы допускают некоторые неточности. Например, инвесторов интересуют обычно процентные, а не абсолютные доходы, то есть необходимо заменить метрику среднеквадратичное отклонение (MSE) на среднее процентное отклонение (MAPE). Таким образом, каждая ценная бумага будет вносить одинаковый вклад в функцию потерь и модель не будет учиться предсказывать наиболее точно только для бумаг с наибольшей абсолютной стоимостью.

В прошлой статье авторы использовали VADER для классификации новостей. То есть похожие новости будут получать примерно одинаковую семантическую оценку. Эта оценка сделана лишь на тексте самой новости, а не на результате новости, выраженном в изменении стоимости акции. То есть одинаковая по смыслу новость может по-разному влиять на различные компании. Например, основываясь на личном опыте, зачатую повышение долговой нагрузки воспринимается инвесторами негативно, однако существуют компании, которые всю свою историю существования

имеют высокую долговую нагрузку и ее повышение не приводит к изменению стоимости акций. Главное, чтобы компания могла обслужить свои обязательства.

В связи с этим пользоваться VADER не разумно, необходимо изучить процесс обработки текстов с целью дальнейшего построения модели классификации новости. Для этого рассмотрим статью [Kannan S. et al. 2014][3].

Основная проблема обработки текстов заключается в том, что может быть сложно выделить основную информацию в тексте из-за наличия различных форм или шумовых слов. Также каждое слово должно быть определено численно в соответствии с важностью информации, которое оно несет. Существует несколько способов решения этой проблемы перечисленными ниже способами.

Токенизация — это процесс разбиения текста на слова, фразы, символы и другие текстовые элементы, несущие информацию. Целью токенизации является исследование отдельных слов (или комбинаций слов, в зависимости от настроек алгоритма). Но токенизация лишь разделяет слова и пунктуационные символы, а не удаляет них, поэтому нужна дальнейшая обработка текста в зависимости от языка. Однако с русским языком проблем не возникнет, так как все слова в нем четко разделяются по сравнению, например, с китайским. (В самой статье не говорится по русский язык, однако, упоминается про английский и французский, с точки зрения разделимости слов эти языки похожи.)

Теперь необходимо удалить все лишние слова, не несущие никакой полезной информации и знаки препинания. Под лишними словами понимаются “стоп-слова” — слова, которые встречаются в тексте очень часто и используются для связки слов и предложений. К ним относятся предлоги, союзы, местоимения, частицы и междометия. Одним из плюсов этого процесса также является уменьшение объема данных без потери точности прогноза.

Далее автор статьи предлагает перейти к стемингу для нормализации форм слов, данный подход предполагает обрезание с конца каждого слова, что хорошо работает для английского языка, но не факт, что сработает для русского, поэтому в этом вопросе необходимо обратиться к опыту российских исследователей.

По сравнению с прошлой статьей методы в [Kozhevnikov V. A., Pankratova E. S. 2020][4] очень похожи, за исключением нормализации.

[3] Kannan S. et al. Preprocessing techniques for text mining //International Journal of Computer Science & Communication Networks. – 2014. – Т. 5. – №. 1. – С. 7-16.

В русском языке нормализация предполагает приведение слова к какой-то определенной канонической форме, такой подход называется “лемматизация”. Это надо для того, что компьютер воспринимал похожие с точки зрения смысла, но разные по форме слова, одинаково. То есть в отличии от стеминга в лемматизации учитывается морфологическая составляющая слова, таким образом, наши данные потеряют меньше информации при обработке. Более того, в результате данной процедуры объем данных также сократиться без качественных потерь.

Две данные статьи представляют из себя полноценную инструкцию по предобработке текстовых данных, поэтому проанализированные статьи я буду использовать в виде пошаговой инструкции при написании работы.

После получения обработанных текстов, необходимо перейти к задаче классификации. С методами соответствующего анализа можно ознакомиться в статье [Vajrala A. 2019][5].

В статье предлагается оформить данные, полученные после подготовки текста в виде TF-IDF матрицы. Такая структура данных позволяет учесть то, как часто слово встречается в конкретном документе и как редко оно встречается в каком-либо другом. То есть учитываются не просто слова в тексте, а степень их принадлежности к какому-то классу или категории. Также плюс такого подхода заключается в том, что такую матрицу можно расширить на комбинации слов, то есть будет учитываться семантический смысл словосочетаний, однако при этом сильно возрастет размерность данных.

Автор приводит множество возможных вариантов моделей, которые могут быть использованы, но больше всего внимания уделяет нейронным сетям Двунаправленной долгой краткосрочной памяти (BLSTM). Данная структура нейронной сети в обработке текстов показывает себя хорошо, так как она способна “запоминать” события из прошлого, то есть как LSTM нейронная сеть, но при этом двунаправленная модификация позволяет учесть не только слова, которые шли до текущего слова, но и слова после. Такой подход позволяет учесть семантику настоящего текста, где смысл определенного слова не всегда зависит от смысла прошлых, но и будущих.

Вероятно, вычислительная сложность такого подхода будет высока, поэтому автор предлагает еще такие модели как: логистическая регрессия, байесовский

[4] Kozhevnikov V. A., Pankratova E. S. Research of text pre-processing methods for preparing data in machine learning.

[5] Vajrala A. Text Classification

классификатор и случайный лес. Также предложены различные модификации нейронных сетей: глубокая нейронная сеть с множеством связей и слоев, нейронная сеть со свертками, рекуррентная нейронная сеть, управляемая рекуррентная нейронная сеть, нейросеть долгой краткосрочной памяти (при проверке на данных она показала наилучший результат, но двунаправленная нейросеть не проверялась).

Подбор гиперпараметров предполагается на основе анализа метрик, вытекающих из матрицы ошибок (Precision, Recall., F1-score, ROC-AUC).

Польза данной статьи для моего исследования заключается в том, что она предлагает способы оценки позитивности или негативности новости. Таким образом, я могу провести предложенный анализ для новостей относительно каждой исследуемой компании.

Трансформеры на данный момент — наиболее успешные архитектуры нейронных сетей для работы с текстовыми данными, так как они в отличие от RNN и CNN способны акцентировать внимание сразу на всем тексте новости. Поэтому просто необходимо опробовать трансформер в сравнении с CNN или RNN. Этим уже занимались в статье [Liu J. et al. 2019][6].

Авторы с помощью нейросети с архитектурой CapTE на основе трансформера делали предсказание направление движения акции до закрытия торгового дня по твитам с соответствующими названию компании тегами. То есть они учитывали не только объективную информацию, но и мнения любого пользователя Твиттера, который решил о ней высказаться.

В результате авторы получили, что трансформер превосходит сверточные и рекуррентные нейросети. Однако, как мне кажется, то, что они учитывали мнение всех инвесторов, могло добавить шума в модель, ведь рынок не всегда движется так, как этого ожидают активные пользователи Твиттера, поэтому стоит рассматривать только объективную или кажущейся на момент выхода новости объективной информацию. Также они рассматривали только предсказание на день вперед, но можно рассмотреть и другие временные интервалы.

Политические новости — без сомнения, один из сильнейших источников волатильности на российском рынке ценных бумаг. Так как моя диссертация посвящена именно российскому рынку, этот тип новостей нельзя не разобрать

[6] Liu J. et al. Transformer-based capsule network for stock movement prediction //Proceedings of the First Workshop on Financial Technology and Natural Language Processing. – 2019. – С. 66-73.

подробно. В статье [Volodin S. N., Kuranov G. M., Yakubov A. P. 2017][7] рассматривается влияние новостей политики на показатель основного биржевого индекса MICEX, а также его второстепенных индексов, отвечающих за компании в различных отраслях экономики. Индекс состоит из крупнейших и наиболее ликвидных публичных компаний России.

Поведение ценных бумаг рассматривается на периоде начала 9 января 2014 года по 30 декабря 2015-ого года, данный период выбран, так как в этот момент в медиапространстве находилось множество новостей политики. Классификация новостей бинарная: позитивные и негативные. Позитивные новости являются таковыми, если после их выхода биржевые индексы растут в цене, для негативных новостей соответствует обратное.

Для анализа автор использует GARCH модель, для каждого сектора подбирается оптимальная модификация модели. Цена берется как цена закрытия торгов на дату.

Не удивительно, что автор получил следующие результаты: положительные новости влияют положительно на стоимость индексов, а отрицательные — отрицательно. Если говорить об отраслях, то было выявлено, что сильнее всего положительные новости влияют на нефтегазовую, телекоммуникационную и финансовые отрасли. Меньше всего положительные новости влияют на отрасль машиностроения/производства техники и на энергетическую отрасль.

Если говорить про негативные новости, то самое большое влияние они оказывают на те же отрасли, что и самое большое позитивное влияние. Однако наиболее устойчивыми к негативным новостям оказались такие сектора как: ритейл, производство удобрений и транспортный сектор.

Данная работа может быть мне полезна с точки зрения выбора источников данных. То есть мне стоит обратить внимание на категорию «Политика» в крупных новостных источниках, если она имеется, и получить данные оттуда в том числе.

1.2 Выводы из обзора литературы.

Проанализировав статьи по схожей с моей работой тематикой, можно сделать вывод, что прогнозирование стоимости акций на основе новостей с помощью нейросетей возможно.

[7] Volodin S. N., Kuranov G. M., Yakubov A. P. 2017 Impact of Political News: Evidence from Russia //Scientific Annals of Economics and Business. – 2017. – Т. 64. – №. 3. – С. 271-287.

Сверточные нейросети хорошо воспринимают локальный контекст слова, то есть они могут видеть заданное количество слов слева и справа от конкретного слова в некоторой последовательности, однако, они не могут выучить взаимосвязи слов, которые лежат за пределами этого окна. Рекуррентные нейросети воспринимают всю последовательность слов в предложении, однако они больше внимания уделяют именно концу последовательности, так как они работают последовательно и более ранние слова в предложении сетью просто забываются. Также один проход по данным занимает много времени, так как они не могут работать параллельно из-за своего дизайна. Более того, им надо много итераций для обучения, так как градиент при обратном распространении ошибки в рекуррентных сетях затухает.

Проблемы сверточных и рекуррентных нейросетей могут решить трансформеры, так как они одновременно анализируют весь текст новости одновременно, однако это делает модель очень тяжеловессной, то есть в ней присутствует огромное число параметров, и для обучения такой сети может понадобиться большое число вычислительных мощностей и данных. Однако веса трансформеров можно найти в Интернете. Эти веса будут заточены под общие языковые задачи, но модели можно дообучить так, чтобы сеть лучше воспринимала финансовые новости. Таким образом, в качестве нейросетевого подхода будет использована предобученная нейросеть трансформер.

Если говорить про выбор метрик для оценки результатов моделей, то можно сделать вывод, что для задачи классификации стоит использовать стандартные для этого метрики Accuracy, Precision. Recall, AUC-ROC, F1. Также существует еще метрика под названием «Каппа Коэна», она также используется для оценки качества классификации, однако не упоминается в статьях. Ее также можно рассчитать для получения дополнительной информации о качестве прогнозов.

Для применения методов классического машинного обучения текстовые данные необходимо предварительно обработать. Собранные тексты для надо очистить от часто встречающихся в русском языке слов, которые не несут особой смысловой нагрузки и знаков препинания. Затем каждое слово должно быть приведено к начальной форме. Таким образом, словарь уникальных слов существенно сократится при минимальной потере информации. Затем обработанные тексты необходимо преобразовать в понятный компьютеру вид, то есть в численный. Для этого подойдет метод TF-IDF как бейзлайн для классчиеского ML. В качестве классического ML будет использован

случайный лес и градиентный бустинг над деревьями. Затем с ним будет произведено сравнение нейросетевого подхода на основе трансформера.

2 Глава. Методы для прогнозирования изменения стоимости акций на основе текстовых данных.

2.1 Случайный лес

Полное описание алгоритма случайного леса приведено в статье [Biau, Scornet 2016][8]

Для описания алгоритма случайного леса для начала необходимо описать алгоритм решающего дерева, так как случайный лес является комбинацией решающих деревьев.

Итак, решающее дерево состоит из набора внутренних и листовых вершин. В каждой внутренней вершине V алгоритм пользуется решающем правилом:

1. В внутренних вершинах к выборке, находящейся в этой вершине, применяется предикат, который разбивает выборку в вершине на две части.
2. Каждой листовой вершине соответствует предикат целевой метки.

Принцип работы алгоритма случайного дерева заключается в жадном выборе переменной, для которой разбиение в конкретной вершине будет оптимально с точки зрения критерия расщепления:

$$Q(R, \Theta) = H(R) - \frac{|R_l|}{|R|} * H(R_l) - \frac{|R_r|}{|R|} * H(R_r)$$

Где Q — значение критерия расщепления, R — внутренняя вершина дерева, |R| — количество объектов в вершине R. Θ — параметры модели, то есть признак, по которому проведено разбиение и значение разбиения для этого признака. l — индекс левого листа, r — индекс правого листа. H(R) — мера информации в текущем листе. Может быть измерена с помощью коэффициента Джини (как правило, для задачи регрессии) или Энтропии (как правило, для задачи классификации).

Итого алгоритм случайного леса представим в следующем виде:

1) Семплируется бутстреп подвыборка наблюдений.
2) Случайным образом выбирается подвыборка признаков. Размер этой подвыборки задается гиперпараметром.

[8] Aizawa A. An information-theoretic perspective of tf–idf measures //Information Processing & Management. – 2003. – Т. 39. – №. 1. – С. 45-65

3) По полученной бутстреп подвыборке и набору признаков строится решающее дерево.
4) Повторяем пункты 1-3 для заданного гиперпараметром числа решающих деревьев.
5) Результаты полученных деревьев усредняются.

Алгоритм случайного леса обладает немаловажными преимуществами. Он способен улавливать нелинейные связи в данных. Вычислительно легкий и быстрый по сравнению с другими «недеревянными» алгоритмами машинного обучения. И устойчив к выбросам за счет большого числа различных аппроксиматоров (деревьев).

2.2 Градиентный бустинг

Полное описание алгоритма градиентного бустинга можно найти в статье [Natekin, Knoll 2013][9].

На данный момент алгоритмы бустинга являются наиболее точными и широко используемыми из алгоритмов классического машинного обучения. Идея градиентного бустинга заключается в том, что на каждой итерации алгоритма, он пытается исправить ошибки, допущенные на предыдущей итерации. Этот алгоритм является алгоритмом ансамблирования слабых моделей (модели, которые способны хотя бы с минимально отличной от случайной точность предсказать целевую переменную). Чаще всего в качестве слабого алгоритма используется решающее дерево за счет его вычислительной простоты.

Алгоритм градиентного бустинга представим в следующем виде:

1. Обучаем базовый алгоритм $b_i(x)$
2. Вычисляем градиент s_i^t функции потерь в точках ответа базового алгоритма.
3. Обучаем новое дерево на ошибках дерева с предыдущего шага, то есть на градиентах s_i^t.
4. Обновляем деревья в соответствии с градиентами функции потерь.

Каждое решающее дерево строится в точности так же, как описано в 2.1.

2.3 Эмбеддинги

[9] Natekin A., Knoll A. Gradient boosting machines, a tutorial //Frontiers in neurorobotics. – 2013. – Т. 7. – С. 21

Эмбеддинг — это сопоставление некоторой точки в численном пространстве некоторому наблюдению, в данном случае наблюдение — слова. То есть таким образом мы можем получить маломерное числовое представление текста. Это маломерное представление называют скрытым представлением текста. При использовании некоторых подходов мы можем самостоятельно задать размерность скрытотго представления. Для каждого слова рассчитывается его численное описание и из них затем составляется матрица конкатенацией этих слов в порядке, с котором они идут в предложении.

Для классических моделей машинного обучения эмбеддинги получаются стандартными алгоритмами, например, TF-IDF, Word2Vec и FastText. Однако на данный момент существуют обучаемые эмбеддинги как часть нейросети, что упрощает задачу выбора алгоритма получения эмбеддинга, сети надо лишь передать набор слов текста и их ключи, количество которых совпадает с уникальными элементами текста (элементом текста может быть как слово, так и несколько букв из слова). Далее во время обучения будет построена оптимальная для конкретной задачи матрица эмбеддингов с заданной размерностью скрытого пространства. Итого мы получаем матрицу для скрытого представления текстов размерности число ключей на размерность скрытого пространства.

Важной особенностью построения эмбеддингов является то, что смысловое положение слов в тексте отразиться и в числах, например, если мы будем складывать векторные представления слов «королева» и «мужчина», то мы получим векторное представление слова «король».

Далее, когда я буду говорить про работу моделей с текстами, будет подразумеваться, что модель работает не напрямую с текстом, а с некоторыми эмбеддингами.

2.3.1 TF-IDF

Полное описание алгоритма TF-IDF можно найти в статье [Aizawa 2003][10].

TF-IDF — наиболее простой способ преобразование текстов к векторному преобразованию после dummy-переменных, но с ним алгоритмы работают существенно лучше по сравнению с dummy подходом. Он должен применяться только на предобработанных текстах, соответствующие методы были описаны в обзоре

[10] Aizawa A. An information-theoretic perspective of tf–idf measures //Information Processing & Management. – 2003. – Т. 39. – №. 1. – С. 45-65.

литературы, так как размерность векторов напрямую зависит от словаря уникальных слов в данных.

TF-IDF представление текстов состоит из двух частей:

1. TF(Term Frequency) — частотность некоторого слова в отдельно взятом тексте. Можно рассчитать по формуле:

$$TF = \frac{n_t}{\sum_{i=1}^{k} n_i}$$

Где n_t — сколько раз конкретное слово t встретилось в конкретном тексте; $\sum_{i=1}^{k} n_i$— совокупное число слов в тексте.

2. IDF (Inverse Document Frequency) — отвечает за частотность использования некоторого слова.

$$IDF = ln\left[\frac{n_c}{1 + df(t)}\right]$$

Где n_c— количество текстов в наборе данных. df(t) —количество текстов, содержащих слово t.

Итого получаем формулу для TF-IDF:

$$TF - IDF(t, d, D) = TF(t, d) * IDF(t, D)$$

Где d — количество слов в одном тексте;

D — общее количество текстов в данных.

2.3.2 Нейросетевые Эмбеддинги

Преимуществом использования нейросетевого эмбеддинга является возможность выучить оптимальное векторное представление для любой заданной наперед размерности. То есть, если же в TF-IDF подходе мы задавали размерность эмбеддинга, то при достижении заданной размерности, метод просто отбрасывал бы слова, которые встречались меньше всего раз в наших текстах, тем самым это может приводить к потере информации от этих слов.

В нейросетевом эмбеддинге будут учитываться абсолютно все слова, при этом мы можем ограничить размерность векторного представления слова удобной нам величиной с точки зрения доступных ресурсов по памяти и скорости вычислений.

Слой эмбеддингов в нейросети — это просто матрица, строки которой соответствуют словам из текстов в наших данных. То есть она имеет размерность n * hidden_size, где n — количество уникальных слов в нашем словаре, hidden_size — гиперпараметр, который задает размерность вектора для описания наших слов.

Все слова в нашей выборе кодируются индексами, которые сопоставляются с соответствующими строками эмбеддинга. То есть при использовании сети из эмбеддингов просто достаются соответствующие словам в предложении вектора, к которым уже могут применены дальнейшие нейросетевые преобразования.

Так как, по сути, эмбеддинг является линейным слоем, то его можно обучать с помощью обратного распространения ошибки. То есть, таким образом можно получить оптимальные с точки зрения функции ошибки векторные представления слов для конкретной задачи непосредственно во время обучения нейросети.

Более того, веса нейросетевых эмбеддингов можно найти в Интернете, что упрощает процедуру обучения сети, так как мы уже имеем хорошие векторные представления слов. Остается лишь дообучить их под нужную задачу, чтобы учесть ее специфику.

Эмбеддинги для классчиского машинного обучения также можно найти в Интернете, но, к сожалению, их нельзя дообучить под конкретную задачу, что может помешать решению задачи.

2.4 Трансформеры

Трансформеры — наиболее современный нейросетевой подход для работы с текстами. Благодаря своей идее у них нет ограничений на размер «окна» (помимо максимального размера, который существенно больше, чем, например, для свреточных нейросетей), то есть нейросеть сразу смотрит на весь входной текст, и они могут эффективно обучаться, в отличие от рекуррентных нейросетей, за счет своей легко распараллеливаемой архитектуры.

В основе трансформеров лежит механизм самовнимания [Vaswani 2017][11]. Его идея заключатся в том, что он находит в тексте слова, лежащие в одном смысловом поле. Например, рассмотрим предложение: «Я продолжал лить в чашку воду, пока она, наконец, не заполнилась.» Выделенные жирным текстом слова модель определит как связанные между собой, хотя некоторые из них стоят на расстоянии и по отдельности не несут всего смысла.

В идею механизма самовнимания заложена работа с тремя векторами: вектор ключей, вектор значений и вектор запросов. Мы хотим взять вектор запроса и векторно перемножить его с векторами ключей. Наибольшее значение данного произведения

[11] Vaswani A. et al. Attention is all you need //Advances in neural information processing systems. – 2017. – Т. 30

будет соответствовать наиболее подходящему под запрос ключу. Затем полученные значения приводятся в шкалу 0-1 с помощью функции Softmax и скалярно перемножаются с вектором значений. Таким образом, получается, что наибольшее внимание получают самые важные в предложении слова, то есть у которых выход Softmax наиболее близок к 1. Также делается поправка на длину входного текста, так как, чем он длиннее, тем больше будет значение векторых произведений, что может привести к проблемам с обучением из-за переполнения.

Итак, формула механизма самовнимания выглядит следующим образом:

$$\boldsymbol{Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V}$$

Где:

Q — матрица запросов (queries);

K — матрица ключей (keys);

V — матрица значений (values);

d_k — размерность входного ембеддинга;

softmax — векторная функция вида $softmax_k = \frac{e_k}{\sum_{i=1}^{n} e_i}$.

Трансформер, как правило, состоит из комбинации нескольких механизмов самовнимания для увеличения выразительности сети. По сути, применяя механизм самовнимания к входному эмбеддингу мы преобразуем данные в другой эмбеддинг, где выделены наиболее важные его элементы с точки зрения модели.

Для того, что классифицировать входной текст необходимо в трансформер в начало нашей текстовой последовательности добавить специальный токен, отвечающий за классификатор, например, токен <CLS>. Именно в соответствующем этому токену векторе трансформер будет кодировать информацию, необходимую для классификации текста. На выходе из трансформера необходимо выстроить полносвязный классификатор, который будет обращаться к элементам выхода трансформера, соответствующим токену <CLS>.

3 Данные

Весь код и ссылки на все данные и модели можно найти по ссылке на репозиторий на GitHub https://github.com/BorisenkoGeorgy/Disser_news_stocks/tree/dev.

3.1 Классические новости

В качестве источника традиционных новостных данных были выбраны крупные новостные издания России (РИА новсти, Комерстант, Лента Ру, Ведомости). Новости с них были получены с помощью самостоятельно реализованных на языке программирования Python парсинговых программ. К сожалению, данные из РБК (данный ресурс был интересен как один из основных источников новостей бизнеса и экономики) не удалось получить, так как у них нет полного публичного архива, а мои неоднократные обращения в поддержку были проигнорированы.

Всего было получено 716740 уникальных текстов за интервал с 1 января 2010-ого года по 20 ноября 2022-ого года.

Источник	Раздел	Количество новостей
РИА новости	Экономика	271.4 тыс.
РИА новости	Политика	101.3 тыс.
Ведомости	Экономика	7.9 тыс.
Ведомости	Бизнес	36.3 тыс.
Ведомости	Финансы	11.9 тыс.
Ведомости	Личный счет	1.1 тыс.
Ведомости	Инвестиции	192
Комерсант	Экономика	36.2 тыс.
Комерсант	Политика	53.1 тыс.
Комерсант	Финансы	31.2 тыс.
Комерсант	Бизнес	64.4 тыс.
Комерсант	Потребительский рынок	40.9 тыс.
Лента Ру	Экономика	60.5 тыс.

По датам новости собирались с начала 2010-ого года по 22-ое ноября 2022-ого года. Распределение новостей по датам и источникам выглядит следующим образом (Рисунок 1):

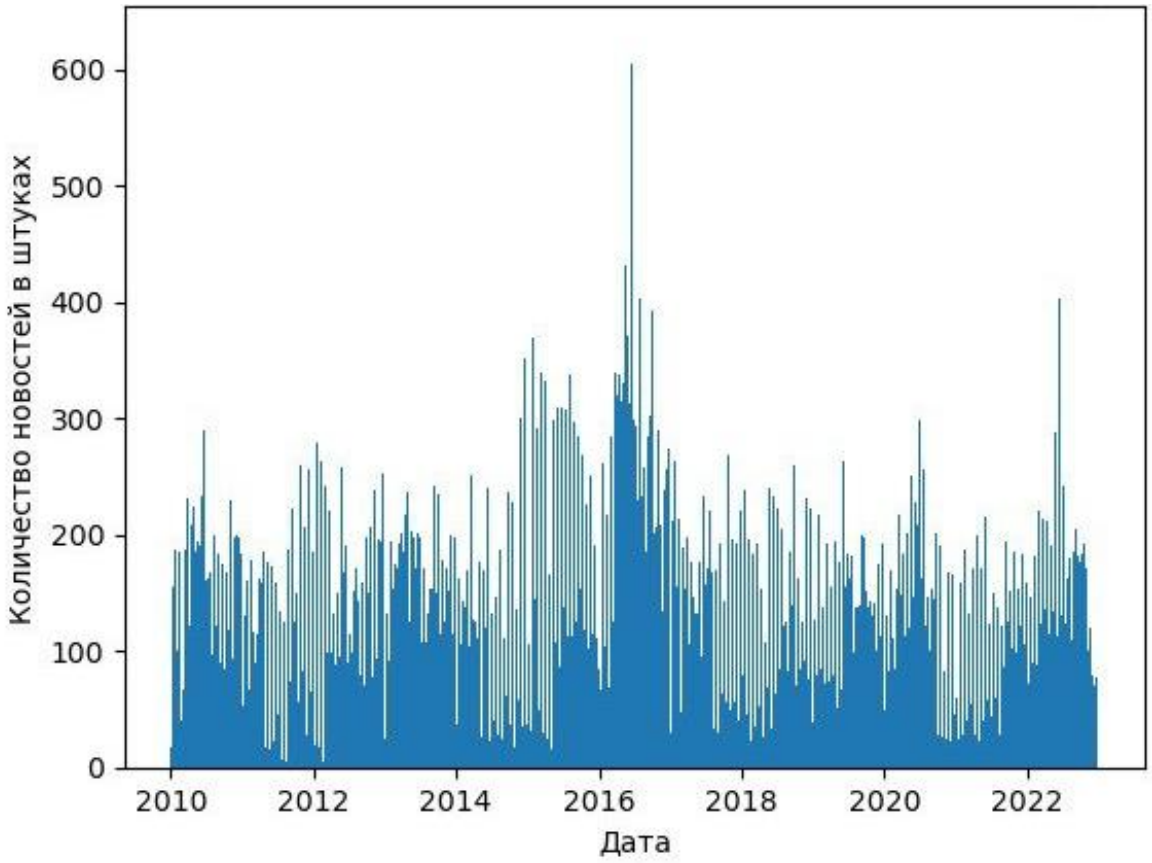


Рисунок 1 Количество новостей из классических новостных источников по дням

Количество новостей по дням с 2010-ого года выглядит довольно стабильным. Количество новостей в среднем примерно не изменялось с годами.

3.2 Новости из Телеграмма

В Телеграмме существует множество различных каналов, основная идея которых — освещение актуальных событий о публичных компаниях. Их существенно больше, чем крупных новостных источников, поэтому необходима процедура отбора каналов, чтобы максимально покрыть информационное поле публичных компаний.

На данный момент на рынке ценных бумаг я нахожусь уже более 5 лет, последние 3 года я получаю новостную информацию о компаниях только из Телеграмм-каналов. Поэтому, в качестве базового набора каналов, я выбрал каналы, которыми пользуюсь сам. Далее с помощью самомстоятельно написанной парсинговой программы, реализованной на языке Python, я получил все новости из этих каналов.

Телеграмм-каналы часто пересылают новости друг от друга, и эта информация может быть получена также может быть получена при парсинге. Таким образом, из полученных мной данных можно также извлечь информацию о каналах, на которые

ссылается конкретный выбранный канал. В результате я составил матрицу популярности каналов. То есть я определял популярность канала по тому, сколько каналов и как часто на этот канал ссылаются. Сделал отсечку, что в среднем должно быть 25 ссылок на некоторый канал, чтобы можно было его считать значимым. Некоторые каналы из полученных таким образом оказывались закрытыми каналами с торговыми сигналами, такие каналы я игнорировал, потому что они содержат в основном информацию о рекомендации по сделке, а не сами новости.

Далее, я итеративно совершал действия, описанные выше, пока к списку каналов не перестали добавляться новые каналы, таким образом, я считаю, мне удалось охватить максимально широкое информационное поле, ограничившись условно небольшим числом Телеграмм-каналов.

Итого было получено 1043208 уникальных текстов за период, начиная от создания каждого канала до 15 января 2023 из следующих каналов (Таблица 1):

Канал	Тег канала	Количество постов
Stock news	@StockNews100	53.6 тыс.
RT на русском	@rt_russian	142.1 тыс.
Сигналы РЦБ	@cbrstocks	43.1 тыс.
Никита Кричевский	@antiskrepa	13.6 тыс.
bitkogan	@bitkogan	19.1 тыс.
Energy Today	@energytodaygroup	17.7 тыс.
ФИНАСКОП	@FINASCOP	3.1 тыс.
ГазМяс	@gazmyaso	5.2 тыс.
Раскрытие корпоративной информации	@information_disclosure	29.8 тыс.
Кот Эльвиры	@kotelviry	3.9 тыс.
Московский Мамковед	@kremlin_mother_expert	13.6 тыс.
MarketOverview	@marketoverview	8.7 тыс.
Мысли-НеМысли	@mislinemisli	12.2 тыс.
MOEX – Московская бмржа	@moscowexchangeofficial	1.4 тыс.
Небрехня	@nebrexnya	9 тыс.

Газ-Батюшка	@papagaz	12.5 тыс.
РБК	@rbc_news	66.3 тыс.
РИА Новости	@rian_ru	191.1 тыс.
MMI	@russianmacro	14.9 тыс.
СМАРТЛАБ	@smartlabnews	8.2 тыс.
MarketTwits	@markettwits	213.7 тыс.
Smartlab news	@newssmartlab	30,5 тыс.
Банкста	@banksta	29,5 тыс.
Cbonds.ru	@cbonds	12.4 тыс.
Дивденды Forever	@divForever	15 тыс.
Dividend News	@DividendNews100	3,2 тыс.
Экономика	@economika	27 тыс.
Больше, чем экономика	@economylive	15 тыс.
Жирные коты	@FatCat18	8,6 тыс.
Газпром нефть	@gazpromneft_official	526
Газпром	@gazprom	1024
Кабинет инвестора	@investingcorp	2.7 тыс.
Не движется	@nedvizhna24	746
РынкиДеньгиВласть | РДВ	@AK47pfl	13.5 тыс.

Таблица 1 Распределение новостей по Телеграмм каналам

Рассмотрим гистограмму публикаций сообщений по дням (Рисунок 2). Явно виден тренд на повышение числа новостей в день. Это связано с тем, что не все перечисленные Телеграмм-каналы были основаны в один день. Телеграмм на данный момент активно развивается и такой вид графика вполне ожидаем. Из-за такого распределения новостей возможна проблема, что нейронная сеть будет не очень хорошо прогнозировать исход событий для стоимости ценной бумаги, похожих на те, которые произошли до 2018 года из-за отсутствия большого числа информации.

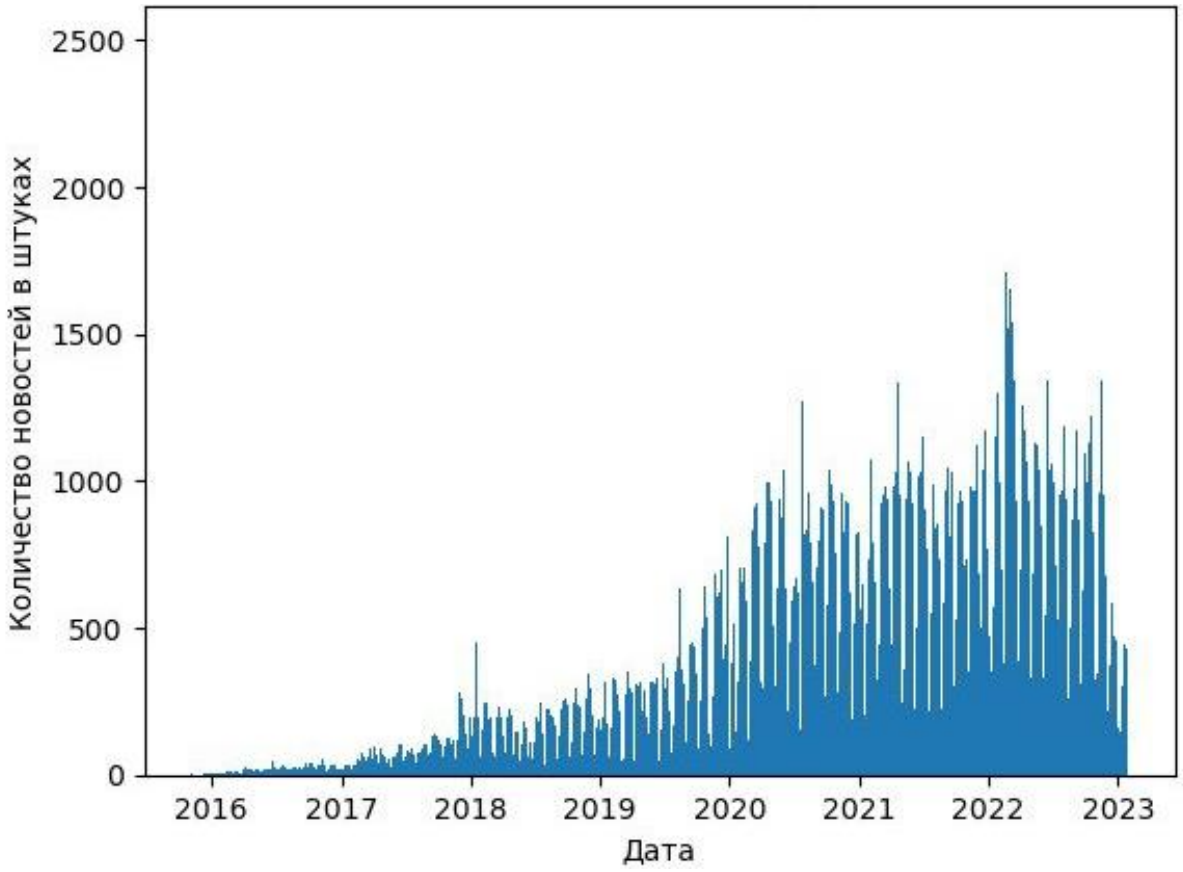


Рисунок 2 Количество новостей из Телеграмма по дням

3.3 Предобработка текстов

Для работы с текстовыми данными их необходимо преобразовать к виду, понятному для модели, то есть — к числовому.

Для классического ML все текстовые данные были очищены от пунктуации и различных служебных символов, таких как \n и \t, затем очищены от стоп слов (различные предлоги, местоимения, частицы или другие части речи, которые часто встречаются в текстах и не несут значимой семантической нагрузки). Также при обработке текстов Телеграмм-каналов были удалены смайлики. После это все тексты были приведены к нижнему регистру и лемматизированы. Затем к полученным текстам был применен метод TF-IDF.

Нейросетевые модели используют уже предобученные эмбеддинги, которые могут работать с предложениями практически в сыром виде. Были удалены лишь специальные символы \n, \r и смайлики. Затем все слова в предложениях преобразуются к соответствующим номерам, соответствующим строкам обученного эмбеддинга.

3.4 Выбор ценных бумаг

Выбор ценных бумаг — один из важнейших этапов моего исследования, хоть и не очень сложный. Самое главное, чтобы по выбранной компании было относительно много новостей, а рынок по акциям этих компаний был ликвиден. Эти условия важны, чтобы можно было заявить, что акции подвержены влиянию новостей. Наибольшее число новостей выходит о крупных компаниях, также, чем крупнее компания, тем больше ее стоимость и тем больше она привлекает инвесторов, а они, в свою очередь, обеспечивают ликвидность на торгах. Более того, большая ликвидность уменьшает количество случаев манипулирования ценой акции, так как для этого нужны очень большие суммы, что делает акции более простим для прогнозирования, так как они меньше зависят от интересов игроков рынка, пытающихся незаконным путем повлиять на котировки. Также крупные компании заинтересованы в стабильности цены на свои акции, поэтому они могут привлекать маркетмейкеров для предоставления ликвидности, чтобы уменьшить вероятность возникновения необоснованных скачков цен. Перечисленным условиям удовлетворяют акции, входящие в Индекс Мосбиржи.

Для анализа мной были выбраны ценные бумаги, входящие в Индекс Мосбиржи на ноябрь 2021, когда данная работа задумывалась. Из выборки я исключил такие компании (далее компании будут назваться их биржевыми тикерами) как OZON, VKCO, POGR и HHRU, так как на тот момент они недавно появлялись на бирже и из-за этого по ним собрано довольно мало информации, более того, на данный момент POGR находится в процессе банкротства и акции этой компании не торгуются на бирже.

Также из списка бумаг я исключил привилегированные акции SBERP и TATNP, так как в моей выборке присутствует SBER и TATN и их бумаги коррелируют практически с коэффициентом 1 за очень редкими исключениями.

Итоговый список тикеров ценных бумаг, для которых были построены модели машинного обучения, выглядит следующим образом: AFKS, AFLT, ALRS, CBOM, CHMF, DSKY, FEES, GAZP, GMKN, HYDR, IRAO, LKOH, LSRG, MAGN, MOEX, MTSS, NLMK, NVTK, PHOR, PIKK, PLZL, ROSN, RTKM, RUAL, SBER, SNGS, TATN, TCSG, TRNFP, VTBR, YNDX.

3.5 Получение данных о стоимости ценных бумаг и создание целевой переменной.

Данные о стоимости ценных бумаг были также получены с помощью самостоятельно реализованной парсинговой программы на языке Python с сайта финансового портала Финам.ру[12]. По каждой компании были получены данные цены открытия, минимальной, максимальной, закрытия и объем торгов в денежном эквиваленте по интервалам 1, 5, 10, 15, 30 минут, 1 час и 1 день. Впоследствии были использованы только минутные данные, так как из них можно получить данные по всем остальным интервалам.

В работе было опробовано два разных вида целевой переменной:

1. Бинарная, где утверждается, что каждая новость влияет на движение акции. То есть, если средневзвешенная за n минут до выхода новости ниже, чем средневзвешанная цена за n минут после выхода новости, то относим такую новость к классу 1, иначе, к классу 0 (положительный и отрицательный класс в данном случае).
2. Классификация на 3 класса. Здесь в качестве предпосылки берется, то, что не каждая новость оказывает значимое влияние на движение акций компании. За интервал в n предыдущих минут от момента выхода новости вычислялась средневзвешенная цена и ее стандартное отклонение. Если в следующие n минут средневзвешенная цена отклонялась от средневзвешенной за последние n минут более, чем на полтора стандартных отклонения, то такая новость относилась к классу +1 или -1 в зависимости от направления отклонения (позитивный/негативный класс). Если же средневзвешенная цена остается в рамках полутора стандартных отклонений, то такой новости присваивается класс 0, то есть нейтральный. Такой подход к решению задачи кажется более логичным, так как вне зависимости от содержания новости, торги на бирже будут продолжаться, а значит, цена бумаги изменится в любом случае. Из-за этого новости, которые, казалось бы, должны быть нейтральными, при бинарной классификации будут иметь

[12] https://www.finam.ru/

позитивную или негативную разметку в зависимости от случайного движения цены, что добавит шум в модель.

Значение, относительно которого строилась целевая переменная классификации, было выбрана как средневзвешенная цена за определённый промежуток времени, идея о средней цене взята из исследования [Mittal, Goel 2012][13] (за исключением минутных данных для задачи 3-ех классов, так как нет данных по секундам и взвесить не выйдет). Для всех минутных данных рассчитываем среднее, как среднее между ценами открытия, минимума, максимума и закрытия. Затем на основе минутных средних считаем средневзвешенные цены по интересующим интервалам с учетом объемов торгов за минуту. Это представимо в следующем виде:

$$P_i = \frac{\sum_{j=1}^{n} p_j V_j}{\sum_{j=1}^{n} V_j}$$

Где n — количество минутных интервалов в интересующем таймфрейме (например, n=5 для 5-ти минутного таймфрейма);

p_j —средневзвешенная цена за минуту j;

V_j — объем торгов за j-ую минуту.

3.6 Разметка данных с помощью регулярных выражений.

Некоторую часть информации, которая заложена в тексте можно получить из текстов путем поиска ключевых слов в этом тексте. В языках программирования (в том числе и на Python) это реализовано через функционал Регулярных выражений (regex). То есть, если задать набор ключевых слов, можно проверить, какие из них входят в текст, и разметить все тексты соответствующим образом.

С помощью регулярных выражений были отобраны новости, которые относятся к конкретным компаниям и секторам. Например, если в тексте новости встречается «Сбербанк», то эта новость относится к ПАО «Сбербанк». Регулярные выражения состояли не только из названий компаний, но и из их сокращенных названий, биржевых тикеров и названий дочерних предприятий. Для секторов были перечислены основные элементы секторов, например, элементы финансовых рынков для финансового сектора или различные металлы для сектора цветной металлургии.

[13] Mittal A., Goel A. Stock prediction using twitter sentiment analysis //Standford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis. pdf). – 2012. – Т. 15. – С. 2352

Информация по секторам в работе никак не используется, но в дальнейших исследованиях мне бы хотелось применить и ее. Например, по одной новости, которая относится к сектору прогнозировать движение акций всех компаний из моей выборки, входящих в этот сектор.

Итоговые количества новостей по компаниям выглядят следующим образом (Таблица 2):

Компания	Классических новостей	Новостей в Телеграмме
AFLT	9432	5855
ALRS	2720	4273
CBOM	1351	1255
CHMF	3773	3205
DSKY	299	1562
FEES	1626	3903
GAZP	53285	36424
GMKN	2921	4682
HYDR	3760	3802
IRAO	1814	1672
LKOH	9850	7682
LSRG	681	1157
MAGN	2050	2930
MOEX	1633	12168
MTSS	6602	4362
NLMK	2716	3040
NVTK	5675	6898
PHOR	1429	2497
PIKK	1757	2086
PLZL	926	1482
ROSN	22499	11515
RTKM	5166	2742
RUAL	6179	4985
SBER	34915	19412
SNGS	3290	2874
TATN	2250	3687
TCSG	2796	4884
TRNFP	5845	3746

VTBR	31185	15531
YNDX	9222	7301

Таблица 2 Количество полученных новостей по каждой компании для Классических новостных источников и Телеграмма

4 Построение Бейзлайн-моделей

В качестве бейзлайн-моделей мной были выбраны модели Случайного леса и Градиентного бустинга над деревьями из-за своей относительной легкости работы с большими данными. Логистическая регрессия многой не применялась, так как очень плохо работает с данными большой размерности, также не применялся алгоритм SVM, так как его вычислительная сложность $O(n^2)$, что затрудняет его применение на большой выборке данных с использованием процессора, а ресурсы видеокарты лучше потратить на основную цель исследования — нейросети.

Для бейзлайна текстовые данные приводились к числовым с помощью метода TF-IDF. Метод применялся к прошедшим обработку текстам с удалением пунктуации, стоп-слов, лемматизации и приведению текстов к нижнему регистру. Максимальный размер векторного представления был ограничен 10000 сверху. После применения алгоритма оказалось, что этот максимум был достигнут. Также ограничением снизу выступала минимальная встречаемость токена не менее 5 раз. Под токеном подразумевается конкретное слово или пара слов (биграмма). Биграммы используются, чтобы учесть совместную встречаемость слов в текстах.

Подбор параметров для моделей классического машинного обучения проводился с помощью перебора по сетке параметров. Выбор оптимальных параметров осуществлялся с помощью кросс-валидации. Кросс-валидация использовалась обычная, а не для временных рядов, так как формально временного ряда в объясняющих переменных нет, а задача стоит в классификации новости по тексту. Сами тексты между собой имеют ограниченную временную структуру.

Перебиралась только глубина максимальная глубина деревьев, как один из самых важных параметров. Все параметры перебрать вычислительно долго. Обучение и кросс-валидация проводилось на всех доступных данных до 1 июня 2021 не включительно. Под тестовую часть были выделены данные с 1 июня 2021 по 1 января 2022 не включительно. Для каждого временного интервала, каждой компании и каждого источника была построена отдельная модель классического машинного обучения. Каждая модель была сохранена и ее можно найти по ссылке в репозитории на GitHub.

Для задачи бинарной классификации (положительный и отрицательный классы) сравнение происходило со случайным предсказанием, чтобы показать, что в полученных текстах есть информация, способная улучшить качество предсказания

относительно случайного. Под случайным понимается предсказание 0 или 1 с вероятностями, полученными на обучающей выборке. То есть, например, если на часовом временном промежутке на обучающей выборке конкретная бумага росла в 55% случаев, то случайный прогноз будет выдавать на тесте 1 с вероятностью 55% и 0 с вероятностью 45%. Далее 1000 раз производилось семплирование с заданными вероятностями для каждой новости в тестовой выборке и считались целевые метрики (Accuracy, так как в данном случае дисбаланс классов не очень большой и F1-score). Затем для полученных метрик вычислялось среднее и стандартное отклонение. Итого, если модель показывала результаты лучше, чем сумма среднего и стандартного отклонения, то такой результат считался значимым в пользу модели. Также вычислялась величина минимального гарантированного значимого эффекта как разность Accuracy модели и суммы среднего и стандартного отклонения для случайного предсказания. Например, если средняя точность случайного предсказания равняется 55%, стандартное отклонение для него равняется 2%, а точность классификатора составляет 60%, то значение минимального значимого эффекта рассчитывается как (60 – (55 + 2))% = 3%.

Для задачи классификации на 3 класса (положительный, нейтральный и отрицательные классы) процедура сравнения со случайным была полностью аналогичной. Но, так как есть еще и нейтральный класс, можно провести сравнение с предсказанием будущего полностью нейтральным классом. Идейно нейтральный класс означает, что сделки совершаться не будут, то есть таким образом можно проверить выгодно ли совершать сделки по сигналам модели. Для большинства компаний и временных интервалов баланс классов также соблюден, однако оценка F1-меры также проводилась, но результаты для нее вынесены полностью в приложение.

4.1 Результаты для Телеграмма

В данном разделе будут рассмотрены полученные результаты для моделей случайного леса и градиентного бустинга над деревьями. Результаты представлены для двух постановок задач: классификации на 2 класса (позитивный и негативный) и 3 класса (позитивный, нейтральный и негатиыный).

4.1.1 Задача классификации на 2 класса

Для алгоритма случайного леса были получены следующие результаты минимального эффекта по Accuracy (Таблица 3) (в таблице оставлены только самые

лучшие результаты, остальные вынесены в приложение 1 для случайного леса и приложение 2 для бустинга):

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
CHMF	8.49	3.94	3.14	0.81	1.94	1.84	-0.36
DSKY	5.55	3.80	4.51	7.38	7.20	5.35	-4.62
MAGN	5.09	2.61	0.56	1.95	6.94	3.01	2.27
NVTK	3.03	0.45	0.04	-2.15	1.12	0.21	1.46
SNGS	3.73	3.42	3.56	0.13	0.28	0.52	3.51

Таблица 3 Сокращенные результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Телеграмма на 2 класса алгоритмом случайного леса

Для алгоритма бустинга над деревьями были получены схожие результаты, однако качество их ниже (Таблица 4).

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
CHMF	1.68	1.79	0.97	2.58	-0.28	3.18	2.26
DSKY	1.18	1.91	2.45	6.52	5.06	3.27	-7.50
SNGS	1.47	-2.20	3.11	1.62	0.83	2.60	3.39

Таблица 4 Сокращенные результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Телеграмма на 2 класса алгоритмом бустинга над деревьями

Несмотря на то, что по некоторым компаниям на многих временных интервалах удается получить существенный прирост в качестве относительно случайного, общие результаты заставляют желать лучшего (из 29 компаний действительно хороший прогноз выходит только по 5). Именно для этого было принято решение перейти к задаче классификации на 3 класса.

4.1.2 Задача классификации на 3 класса

Для алгоритма случайного леса были получены следующие результаты (Таблица 5). С полными результатами можно ознакомиться в Приложении 3 для случайного леса и в Приложении 4 для алгоритма бустинга над деревьями.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	1.44	4.62	1.32	0.97	0.16	-0.07
AFLT	-0.78	1.94	2.43	1.44	1.47	2.81

	5 min	10 min	15 min	30 min	1 hour	1 day
ALRS	0.40	3.43	1.87	-0.38	2.97	2.76
CHMF	-0.63	7.14	7.67	5.11	7.57	0.17
DSKY	6.49	4.24	9.58	14.61	9.35	-0.27
GAZP	1.92	1.02	1.96	2.38	2.44	-0.46
HYDR	4.52	4.14	6.03	-1.56	1.60	4.14
MAGN	-2.22	2.53	4.70	2.66	5.22	-0.01
MOEX	1.15	1.39	1.74	1.74	2.46	1.29
MTSS	4.41	2.91	5.10	6.24	1.48	-1.19
NVTK	3.00	-0.67	2.36	2.29	6.05	3.54
PHOR	4.83	1.14	1.29	2.63	1.47	-0.16
PIKK	-1.88	4.85	5.43	2.83	2.69	0.07
ROSN	2.84	4.43	4.16	0.54	0.70	1.09
SNGS	6.96	6.36	4.10	1.36	9.93	8.47

Таблица 5 Сокращенные результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Телеграмма на 3 класса алгоритмом случайного леса

Для алгоритма бустинга имеем минимальный эффект на Accuracy для качественно предсказанных компаний (Таблица 6):

	5 min	10 min	15 min	30 min	1 hour	1 day
DSKY	3.44	3.84	8.41	9.97	6.49	0.44
GAZP	0.69	1.45	0.92	2.66	2.13	0.21
LKOH	1.56	1.07	1.51	2.24	0.11	1.01
NVTK	0.14	0.43	1.81	3.90	0.83	1.39
ROSN	-0.25	1.71	1.09	2.37	1.19	1.20
SNGS	0.12	1.17	1.20	2.82	3.31	7.30

Таблица 6 Сокращенные результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Телеграмма на 3 класса алгоритмом бустинга над деревьями

4.2 Результаты для Традиционных источников новостей

В данном разделе будут рассмотрены полученные результаты для моделей случайного леса и градиентного бустинга над деревьями для классических новостных

источников. Результаты представлены для двух постановок задач: классификации на 2 класса (позитивный и негативный) и 3 класса (позитивный, нейтральный и негативный).

4.2.1 Задача классификации на 2 класса

Для алгоритма случайного леса имеем следующие результаты (Таблица 7). Полные результаты вынесены в Приложение 5 для случайного леса и Приложение 6 для бустинга над деревьями.

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	1.59	4.47	3.59	4.78	3.79	3.38	3.12
TCSG	5.13	8.97	7.02	0.61	-3.29	0.92	-2.32
VTBR	5.86	0.37	1.86	1.16	1.04	1.73	-0.36

Таблица 7 Сокращенные результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Классических источников на 2 класса алгоритмом случайного леса

Для алгоритма градиентного бустинга над деревьями имеем схожие результаты с большим количеством незначимых эффектов (Таблица 8).

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	1.77	3.90	4.39	5.30	-5.31	4.06	5.65
ROSN	-2.31	-1.81	5.77	9.20	1.93	6.16	1.53
VTBR	3.35	0.28	0.55	3.24	-1.67	-0.41	-0.16

Таблица 8 Сокращенные результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Классических источников на 2 класса алгоритмом бустинга над деревьями

Снова получили неплохие результаты для некоторых временных интервалов и некоторых компаний, однако для абсолютного большинства других компаний и интервалов качество оставляет желать лучшего, и, более того, не удается выявить каких-либо закономерностей. Поэтому снова необходимо перейти к задаче классификации на 3 класса.

4.2.2 Задача классификации на 3 класса

Рассмотрим результаты для случайного леса (Таблица 9) (полные результаты можно найти в Приложении 7 для случайного леса и Приложении 8 для бустинга над деревьями):

	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	2.59	6.07	6.29	6.38	-6.34	-0.04
MTSS	5.63	0.53	13.68	1.16	6.05	-13.90
SBER	1.13	2.21	2.34	3.10	3.24	6.72
TCSG	1.25	-0.18	0.86	9.25	3.58	1.47
VTBR	2.63	9.73	10.50	10.37	10.02	-10.70
YNDX	4.10	7.26	8.00	4.30	2.11	-1.87

Таблица 9 Сокращенные результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Классических источников на 3 класса алгоритмом случайного леса

Теперь обратимся к результатам работы алгоритма градиентного бустинга над деревьями (Таблица 10).

	5 min	10 min	15 min	30 min	1 hour	1 day
DSKY	18.48	-18.10	13.18	6.70	-9.14	-23.21
GAZP	3.06	1.39	4.85	3.39	6.80	3.99
TCSG	-1.76	4.47	7.73	4.79	0.61	0.17
VTBR	-1.60	5.80	8.41	9.73	6.19	-8.49
YNDX	-0.32	6.43	8.03	7.54	0.23	-5.34

Таблица 10 Сокращенные результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Классических источников на 3 класса алгоритмом бустинга над деревьями

4.4 Итоги построения бейзлайн моделей.

Как видно из приведенных выше таблиц прогнозы для задачи классификации на 3 класса чаще оказываются статистически значимыми по сравнению с прогнозами для задачи классификации на 2 класса, но все-таки для полноты картины необходимо свести все результаты в таблицу, чтобы сравнить между собой различные подходы к решению задачи, модели и источники.

Минутный интервал для задачи классификации на 2 класса не включен в таблицу, так как для классификации на 3 класса он не может быть построен. Соответственно, сравнить его для разных постановок возможности нет.

Задача	Источник	Модель	5 min	10 min	15 min	30 min	1 hour	1 day
2 класса	Телеграмм	Случайный лес	-0.71%	-0.30%	-1.45%	-0.77%	-0.58%	-1.40%

2 класса	Телеграмм	Бустинг	-2.19%	-1.69%	-1.56%	-0.82%	-0.69%	-1.21%
2 класса	Новости	Случайный лес	-4.47%	-3.73%	-4.01%	-2.29%	-3.32%	-6.08%
2 класса	Новости	Бустинг	-4.77%	-4.71%	-5.43%	-4.53%	-4.44%	-5.30%
3 класса	Телеграмм	Случайный лес	0.80%	1.65%	2.02%	1.78%	2.36%	0.45%
3 класса	Телеграмм	Бустинг	-0.34%	-0.07%	0.38%	0.81%	0.50%	-0.68%
3 класса	Новости	Случайный лес	-4.42%	-3.11%	-1.91%	-1.79%	-1.57%	-6.30%
3 класса	Новости	Бустинг	-5.41%	-5.32%	-2.92%	-2.15%	-2.68%	-5.76%

Таблица 11 Сводная таблица по средним минимальным эффектам для Accuracy

Из собранной статистики (Таблица 11) можно сделать вывод, что лучше всего в среднем работает модель случайного леса для данных, полученных из Телеграмма, для задачи классификации новостей на 3 класса. Несмотря на средние отрицательные эффекты в других постановках задачи удалось получить значимые положительные эффекты для некоторых компаний и временных интервалов. На основе этих компаний будет произведено сравнение моделей классического машинного обучения с нейросетевым подходом. Итого компании и интервалы, с которыми будет проводиться сравнение:

Для Телеграмма для 2 классов: MAGN, CHMF, DSKY, SNGS, NVTK;

Для Телеграмма для 3 классов: AFKS, ALRS, AFLT, GAZP, LSRG, DSKY, MAGN, MOEX, MTSS, ROSN, NVTK, PIKK, HYDR, CHMF, AFKS, SNGS, PHOR;

Для Классических источников для 2 классов: GAZP, VTBR, TCSG;

Для Классических источников для 3 классов: VTBR, GAZP, MTSS, SBER, TCSG, YNDX, RTKM.

Также можно сделать вывод по поводу источников и постановки задачи. Как видно из приведенных таблиц, при решении задачи классификации на 3 класса вместо 2 модели начинают предсказывать лучше, то есть идея о добавлении нейтрального класса оправдала себя. Более того, можно заметить, что при обучении моделей на данных из Телеграмма, удается качественно предсказать большее количество компаний и временных интервалов качественно.

5 Нейросетевой подход

В качестве нейросетей модели были опробованы несколько вариантов трансформеров с сайта Hugging Face[14]. Это сайт, на котором в отрытом доступе можно найти уже обученные веса для интересующих моделей. sbert_large_nlu_ru[15], ruRoberta-large[16] и distilrubert-base-cased-conversational[17]. Но в результате была выбрана только одна модель distilrubert-base-cased-conversational, так как из-за своих небольших размеров данная модель позволяла выбрать размер батча как 32. Остальные модели также возможно было обучить, но с меньшем размером батча, что влияло на сходимость, модели очень медленно обучались, как в смысле времени на одну эпоху, так и в скорости падения функции потерь даже на обучающей выборке.

Для работы получения ответов для задачи классификации необходимо было достроить классификатор поверх трансформера. Было опробовано два варианта:

1) Из выходов трансформера получать только выход, который отвечает за токен <CLS> и далее сверху добавить два линейных слоя с нелинейностями и Dropout между ними;

2) Обрабатывать все выходы трансформера, усредняя контекст и конкатенируя его с выходами токена. Это описано в статье, посвященной решению соревнования с помощью трансформера sbert[18]. Хоть для решения поставленной в работе задачи и была использована другая архитектура трансформера, данный метод оказался лучше.

Итоговый вид архитектуры нейросети выглядит следующим образом (Рисунок 3):

[14] https://huggingface.co/
[15] https://huggingface.co/ai-forever/sbert_large_nlu_ru
[16] https://huggingface.co/ai-forever/ruRoberta-large
[17] https://huggingface.co/DeepPavlov/distilrubert-base-cased-conversational
[18] https://habr.com/ru/company/sberdevices/blog/527576/

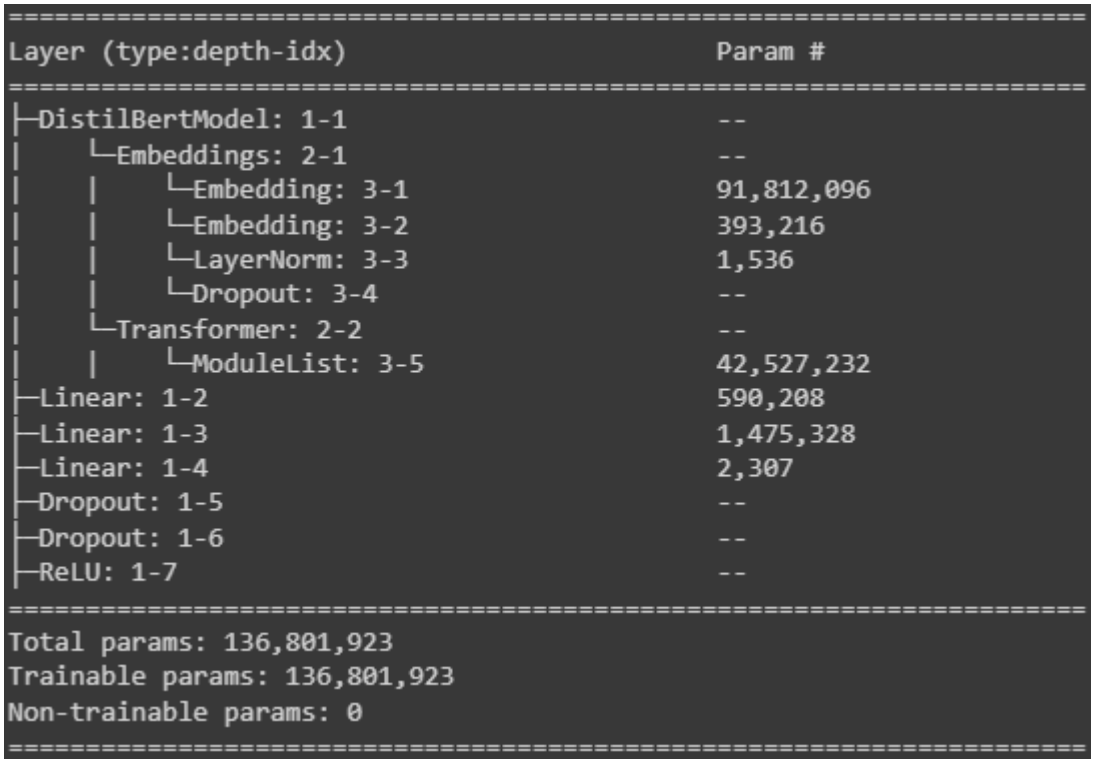


Рисунок 3Архитектура нейросети на основе Трансформера

Для обучения нейросети кросс-валидацию использовать затруднительно, так как это вычислительно трудно, поэтому придется разбить данные на обучение, валидацию и тест. То есть формально обучение по сравнению с классическим ML будет происходить на немного разных выборках, но другого варианта из-за ограничений по ресурсам нет. Итого данные были разбиты по интервалам от начала собранных данных до 1 января 2021 не включительно для обучающей выборки, от 1 января 2021 до 1 июня 2021 не включительно для валидационной выборки и от 1 июня 2021 до 1 января 2022 не включительно для тестовой выборки. Получаем, что тестовые выборки для нейросетевого подхода и классического ML совпадают.

Формально модель может иметь множество выходов и предсказывать сразу несколько целевых переменных. Однако в таком подходе есть проблема. Нейросеть будет оптимизировать среднее значение (арифметическое или геометрическое, в зависимости от того, как задать итоговую функцию потерь) функций потерь для каждого выхода сети, а не каждый выход в отдельности. Из-за этого результаты сети будут получены вероятно хуже, так как на примере результатов классического ML видно, что модели не всегда справляются с задачей. Для правильного сравнения нейросетевого подхода с классическим машинным обучением необходимо на каждую целевую переменную для каждой компании обучать свою модель, что займет очень много времени. Поэтому было принято решение сконцентрироваться на меньшем

числе компаний, а именно, на тех, для которых в соответствующей постановке задачи (2 или 3 класса) и временных интервалов получено не менее 4 положительных из 6 рассчитанных эффектов. Они перечислены в конце пункта 4.4. Не на всех временных интервалах для этих компаний модели классического ML показывают себя лучше случайного предсказания, поэтому также будет возможность сравнить результаты нейросетевого подхода с классическим ML там, где второй не справляется.

Каждая модель обучалась на протяжении 8 эпох. После каждой эпохи на валидационной выборке вычислялись метрики Accuracy, Precision, Recall, F1-score, ROC-AUC и Каппа Коэна. На основе метрики Accuracy сохранялись оптимальные веса модели. После окончания обучения, оптимальные веса заново загружались в модель и выполнялось предсказание на тестовый набор данных.

5.1 Результаты для Телеграмма

5.1.1 Задача классификации на 2 класса

Рассмотрим минимальные эффекты нейросетевого подхода для классификации на 2 класса для Телеграмма (Таблица 12).

	5 min	10 min	15 min	30 min	1 hour	1 day
CHMF	6.52	3.32	5.93	-3.15	4.91	-2.98
DSKY	7.84	4.52	14.88	8.06	4.09	-4.04
MAGN	-0.72	-3.40	0.29	-0.05	-0.11	3.05
NVTK	0.12	1.04	0.82	1.00	0.00	-4.23
SNGS	-1.03	2.69	-2.68	0.36	3.08	2.01

Таблица 12 Результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Телеграмма на 2 класса нейросетью

Сравним с эффектами для случайного леса. Сравнение показано как разность минимальных эффектов для нейросетевого подхода и алгоритма случайного леса (Таблица 13):

	5 min	10 min	15 min	30 min	1 hour	1 day
CHMF	2.62	0.1	5.06	-5.25	2.77	-2.78
DSKY	4.15	0.18	7.7	0.87	-1.32	0.7
MAGN	-3.37	-3.86	-1.73	-7.09	-3.14	0.91
NVTK	-0.22	1.08	3	-0.1	-0.25	-5.64

SNGS	-4.36	-1.04	-2.61	-0.22	2.47	-1.56

Таблица 13 Сравнение по Accuracy Нейросетевого подхода и алгоритма случайного леса для задачи классификации на 2 класса по новостям из Телеграмма

Как видно из сравнения получили, что в среднем нейросеть справляется лучше с прогнозом по 2 из 5 приведенных компаний (CHMF и DSKY). Но выделить какие-то закономерности в предсказаниях не удается.

5.1.2 Задача классификации на 3 класса

Рассмотрим результаты нейросетевого подхода для классификации на 3 класса для Телеграмма (Таблица 14). (Результаты по F1 мере вынесены в Приложение 9.)

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	-3.59	1.96	-5.60	-2.38	-2.12	0.09
AFLT	2.06	2.28	1.56	0.89	-4.61	4.86
ALRS	-3.16	1.35	1.17	-1.98	-3.79	-0.09
CHMF	-2.18	-8.67	-7.05	-4.76	-7.31	-2.99
DSKY	-9.19	-19.53	-17.59	-17.72	-13.51	-8.28
GAZP	4.06	3.79	2.97	3.30	1.93	2.77
HYDR	4.37	4.93	-0.70	-0.45	-2.50	10.21
LSRG	-11.36	-10.61	-10.04	-7.43	-9.05	-6.43
MAGN	3.15	-0.67	2.87	6.46	-0.23	-0.80
MOEX	-2.91	-4.84	-5.44	-2.58	-4.00	-4.62
MTSS	-3.47	-0.82	0.15	-1.80	2.22	-8.15
NVTK	5.73	2.18	0.86	0.91	0.75	3.83
PHOR	-14.94	-10.11	-9.73	-12.94	-11.20	-2.26
PIKK	1.84	0.82	-2.40	5.01	0.88	7.28
ROSN	2.46	1.16	3.57	-0.77	0.50	2.62
SNGS	-0.72	4.76	1.91	-2.57	4.30	4.85

Таблица 14 Результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Телеграмма на 3 класса нейросетью

Теперь сравним разность минимальных эффектов для нейросетового подхода и случайного леса (Таблица 15).

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	-5.06	-2.67	-6.87	-3.27	-2.23	0.22
AFLT	2.83	0.37	-0.97	-0.55	-6.17	1.98
ALRS	-3.66	-1.97	-0.67	-1.63	-6.72	-2.91
CHMF	-1.66	-15.71	-14.77	-9.91	-14.78	-3.04
DSKY	-15.51	-23.54	-27.28	-32.44	-22.77	-8.08
GAZP	2.12	2.79	1	0.93	-0.52	3.22
HYDR	-0.13	0.66	-6.73	1.13	-4.13	6.04
LSRG	-16.49	-13.83	-10.52	-11.06	-9.32	-4.95
MAGN	5.32	-3.04	-1.66	3.6	-5.47	-0.75
MOEX	-4.07	-6.31	-7.24	-4.32	-6.43	-5.88
MTSS	-7.81	-3.78	-4.91	-8.08	0.85	-7.17
NVTK	2.73	2.75	-1.27	-1.4	-5.3	0.33
PHOR	-19.87	-11.23	-11.16	-15.84	-12.65	-2.31
PIKK	3.7	-3.81	-7.81	2.19	-1.68	7.24
ROSN	-0.28	-3.33	-0.55	-1.35	-0.27	1.59
SNGS	-7.58	-1.9	-2.27	-3.77	-5.58	-3.45

Таблица 15 Сравнение по Accuracy Нейросетевого подхода и алгоритма случайного леса для задачи классификации на 3 класса по новостям из Телеграмма

Как видим, нейросеть справляется практически везде хуже по сравнению со случайным лесом, за исключением прогнозирования акций Газпрома.

5.2 Результаты для классических новостей

Таблицы также приводятся одновременно и по нейросетевому подходу, и случайному лесу для сравнения результатов моделей для Классических новостных источников.

5.2.1 Задача классификации на 2 класса

Рассмотрим результаты задачи классификации на 2 класса на основе классических новостных источников с помощью нейросети (Таблица 16).

	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	4.52	5.04	-7.83	-1.54	-8.07	5.06
TCSG	-17.54	-11.33	-8.51	-4.88	-2.73	-2.83
VTBR	6.11	8.49	8.94	6.65	4.41	0.22

Таблица 16 Результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Классических источников на 2 класса нейросетью

Сравним их со случайным лесом (Таблица 17).

	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	0.03	1.33	-12.65	-5.42	-11.42	1.95

TCSG	-26.47	-18.77	-9.22	-1	-3.95	-0.64
VTBR	5.59	6.81	7.87	5.66	2.67	0.49

Таблица 17 Сравнение по Accuracy Нейросетевого подхода и алгоритма случайного леса для задачи классификации на 2 класса по новостям из Классических источников

Как видим, качественно лучше нейросеть справляется только с предсказаниями для ВТБ.

5.2.2 Задача классификации на 3 класса

Рассмотрим результаты задачи классификации на 3 класса для классических новостных источников с помощью нейросети (Таблица 18). Результаты по F1-мере вынесены в Приложение 10.

	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	2.06	2.38	3.77	4.58	1.40	-0.94
MTSS	-5.42	-0.51	-0.65	-1.29	-6.83	-10.19
RTKM	-19.40	-8.49	-8.13	-6.77	-7.01	-5.62
SBER	0.59	-0.61	-0.21	-4.26	-4.57	-9.07
TCSG	-17.46	-17.81	-10.06	-23.44	-6.89	-5.00
VTBR	4.47	5.58	6.50	9.21	9.51	12.66
YNDX	-2.09	-6.79	-6.22	1.73	1.62	-9.23

Таблица 18 Результаты по минимальному эффекту на Accuracy для задачи классификации новостей из Классических источников на 3 класса нейросетью

Теперь рассмотрим в сравнении с эффектами для случайного леса (Таблица 19).

	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	-0.52	-3.77	-2.53	-1.78	7.73	-0.94
MTSS	-11.25	-1.1	-14.4	-2.71	-12.72	3.57
RTKM	-8.03	-15.27	-12.34	-6.9	-10.84	6.22
SBER	-0.48	-2.77	-2.63	-7.46	-7.74	-15.79
TCSG	-18.93	-18.01	-11.13	-32.91	-10.48	-6.51
VTBR	1.7	-4.12	-4.01	-1.26	-0.75	23.22
YNDX	-6.39	-13.78	-14.06	-2.39	-0.65	-7.26

Таблица 19 Сравнение по Accuracy Нейросетевого подхода и алгоритма случайного леса для задачи классификации на 3 класса по новостям из Классических источников

В данном случае нейросеть очень сильно проигрывает случайному лесу. Из значимых эффектов явно выделяется VTBR на интервале в 1 день, там нейросеть показывает сильное превосходство в качестве.

5.3 Итоговые результаты для нейросетевого подхода

Как видно из попарного сравнения таблиц метрик для нейросетевого подхода и случайного леса по результатам исследования получаем, что нейросети на российской фондовой бирже показывают качество в среднем хуже случайного леса для всех постановок задач за некоторыми исключениями:

1. VTBR, движение которых угадывается для задачи 2 классов лучше нейросетью, чем случайным лесом, а на задаче 3-ех классов показывает в среднем качество не хуже для классических новостей;
2. CHMF и DSKY для задачи 2 классов на новостях из Телеграмма;
3. GAZP для задачи классификации на 3 класса для Классических новостных источников.

Еще для некоторых компаний на некоторых временных интервалах нейросеть показывала качество лучше случайного леса, но там сложно однозначно подтвердить превосходство нейросетевого подхода, так как на соседних временных интервалах нейросеть уже проигрывала случайному лесу.

Превосходство случайного леса над нейросетью можно объяснить несколькими способами:

1. Кросс-валидация для случайного леса проводилась обычная, а не специальная для временных рядов, так как формально временной ряд не присутствует в независимых переменных, исследование заключается в влиянии текста новости на движение акций. При дальнейшем исследовании и добавлении авторегрессионности в модель так делать уже будет нельзя. Из-за этого модели классического ML были обучены на данных, которые максимально приближены во времени к данным для теста, то есть модель имела возможность обучаться на самых свежих данных относительно тестовых. Для нейросетей кросс-валидацию использовать слишком вычислительно дорого, поэтому полученные нейросети нельзя было обучить на самых свежих данных.
2. Векторные представления слов для классического ML были получены в результате алгоритма TF-IDF, то есть они были рассчитаны на основе имеющихся данных. Таким образом, эти векторные представления максимально точно (насколько это возможно ввиду простоты метода)

описывали новостную область (область финансовых текстов). Векторные представления нейросети уже были обучены на данных их постов и комментариев социальных сетях, что не относится к области финансовых текстов. Дообучение нейросети частично эту проблему, но только частично. Для полного решения нейросеть необходимо учить с нуля.

3. При встрече в тестовой выборке с новым словом, которого не было в обучающей выборке, алгоритм TF-IDF его просто пропустит, а нейросетевой подход все-так приведет к численному виду, который был оптимален для задачи, на которой изначально обучалось это числовое представление, а не для задачи классификации финансовых текстов.

5.4 Интерпретация предсказаний нейросети

Несмотря на то, что качество для нейросети в среднем хуже, чем для случайного леса, у нейросетевого подхода есть преимущество — благодаря механизму самовнимания, заложенного в идею архитектуры трансформера, можно визуализировать, на что ориентируется модель при принятии решений о классификации. Компании и интервалы, для которых строились визуализации выбирались по F1-мере. Чем она больше, тем более разнообразные ответы модель давала в предсказании, что дает возможность выбирать наиболее интерпретируемые правильно классифицированные новости.

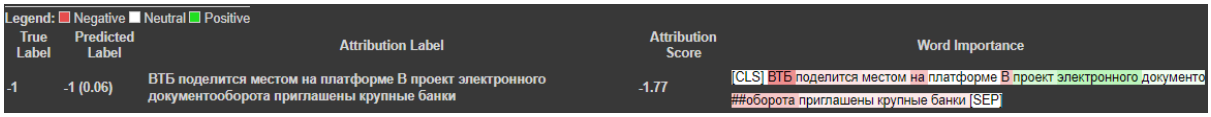
Рассмотрим примеры новостей Телеграмма из тестовой выборки на задаче 3-ех классов (так как для нее было получено наибольшее число отличных от случайных предсказаний), для которых нейросеть успешно определила метку класса и попробуем их проинтерпретировать.

Рассмотрим новость о Сургутнефтегазе от 2021-12-30 из Телеграмм канала finascop: «Сургутнефтегаз ап растет вместе с курсом доллара».

Как видно из визуализации механизма внимания, нейросеть делает наибольший положительный акцент на связи нефти и газа с ростом доллара. Действительно, ПАО «Сургутнефтегаз» занимается добычей нейти и газа, которые в основном идут на экспорт. То есть доходы компании в рублях напрямую зависят от курса доллара. Чем он выше, тем и выше доходы, а соответственно, и стоимость акций компании. И

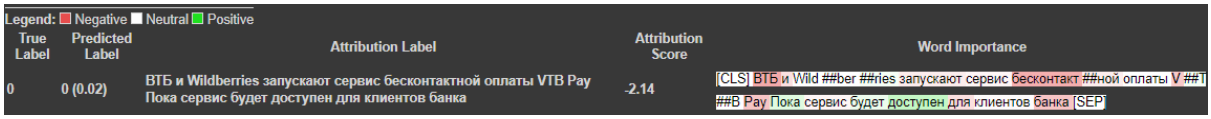
модель верно предсказывает, что акции компании на горизонте в один день покажут рост. Более того, Сургутнефтегаз на момент выхода новости хранил большие суммы валюты в виде денежных средств на банковском счету[19]. В результате валютных переоценок в годы резкого курса Сургутнефтегаз рекомендовал к выплате большие дивиденды, что приводило к росту стоимости его акций. В итоге имеем, что стоимость акций Сургутнефтегаза стремится к росту вместе с курсом доллара, что смогла определить нейросеть.

Теперь рассмотрим новости о ПАО «ВТБ». «ВТБ поделится местом на платформе В проект электронного документооборота приглашены крупные банки».



Банк ВТБ был одним из создателей в партнёрстве с «Ростелекомом» программы электронного документооборота для упрощения взаимодействий между гражданами, государством и бизнесом[20]. Однако на момент выхода новости в «Комерсанте» 2021-09-08 в развитие программы были приглашены и другие банки. Соответственно, акции компании должны были отреагировать негативно, так как доля банка ВТБ в сегменте снизится. Модель верно угадывает направление движение акции и подчеркивает негативным весом, что ВТБ будет делится с крупными банками.

Также рассмотрим нейтральную новость из издания «Ведомости» от 2021-12-07. «ВТБ и Wildberries запускают сервис бесконтактной оплаты VTB PAY Пока сервис будет доступен для клиентов банка».



Здесь также модель выделяет некоторые положительные и отрицательные моменты новости, но относит ее все же к нейтральному классу. На тот момент Apple Pay и Sumsung Pay в России еще работали и недобности в подобном сервисе не было, поэтому, видимо, участники рынка никак на эту новость и не отреагировали. С одной стороны, она может принести дополнительную прибыль банку, но с другой — очень

[19]https://www.tadviser.ru/index.php/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B0%D0%BD%D0%B8%D1%8F:%D0%A1%D1%83%D1%80%D0%B3%D1%83%D1%82%D0%BD%D0%B5%D1%84%D1%82%D0%B5%D0%B3%D0%B0%D0%B7#:~:text=%D0%9A%D0%B0%D0%BA%20%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D0%B5%D1%82%20%C2%AB%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B0%D0%BA%D1%81%C2%BB%2C%20%D0%BB%D0%B8%D0%BA%D0%B2%D0%B8%D0%B4%D0%BD%D1%8B%D0%B5,%D0%B4%D0%BE%D1%81%D1%82%D0%B8%D0%B3%D0%BB%D0%B8%204%2C13%20%D1%82%D1%80%D0%BB%D0%BD%20%D1%80%D1%83%D0%B1%D0%BB%D0%B5%D0%B9.

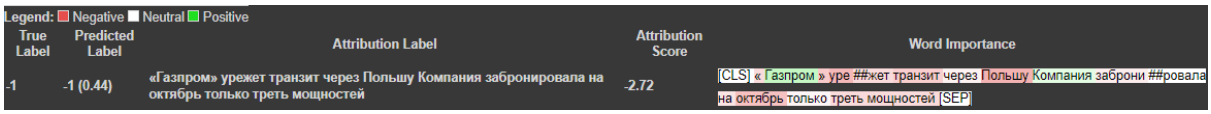
[20] https://regnum.ru/news/polit/3366298.html

сложно бороться с конкурентами, которые уже широко распространены в сегменте бесконтактных платежей, поэтому новость неоднозначная, и модель верно это угадывает.

Теперь рассмотрим новость также о ПАО «ВТБ» из издания Лента.ру от 2021-07-28. «ВТБ создаст экосистему рынка имущественных торгов». Модель больше всего внимания акцентирует на том, что ВТБ займется созданием некоторого проекта по имуществу. Банку этот проект будет выгоден с точки зрения реализации заложенного имущества, которое попало в собственность банка. Тем самым банк будет быстрее избавляться от ненужных ему активов и сможет эффективнее направлять свободные денежные средства в операционную деятельность. Рынок положительно отреагировал на эту новость, и модель смогла предсказать это.

Также стоит обратить внимание, что в приведенных примерах о ПАО «ВТБ» токен ВТБ всегда выделяется негативно. Получается, что модель смогла распознать, что с точки зрения инвестиции, банк «ВТБ» не самый лучший актив. С момента начала датасета по его конец (период 2010-2021) акции ВТБ упали примерно на 40%. А с момента IPO (апрель 2007) на 70%.

Еще стоит обратиться к самой популярно и наиболее часто упоминаемой компании в новостях, к ПАО «Газпром». Рассмотрим новость: «Газпром урежет транзит через Польшу Компания забронировала на октябрь только треть мощностей» —, опубликованную в Коммерсанте 2021-09-20.



Здесь явно виден акцент модели на «урежет транзит» и «треть мощностей». Газпром зарабатывает в основном на поставках газа за рубеж. И снижение поставок естественно приведет к снижению выручки и прибыли, что негативно сказывается на стоимости компании, и модель это правильно предсказывает.

Таким образом, можно сказать, что нейросеть действительно способна понимать суть полученных новостей.

6 Заключение

В данной работе были построены модели классического машинного обучения (случайный лес и бустинг над деревьями) и глубокого обучения (нейросети) для прогнозирования движения цен акций публичных компаний из индекса Московской биржи. Задача была решена в двух постановках: классификация влияния новости на 2 класса (положительное/отрицательное) и на 3 класса (положительное/нейтральное/отрицательное). Также было произведено сравнение моделей, обученных на разных источниках данных.

В результате работы удалось выявить, что при прогнозировании направления движений акций желательно перейти к задаче классификации на 3 класса, так как это позволяет точнее предсказывать движение акций точнее и для большего числа компаний и временных горизонтов по сравнению с предсказаниями моделей для задачи классификации на 2 класса.

Также удалось выявить, что новости из Телеграмма — более надежный источник с точки зрения качества предсказания моделей по сравнению с классическими новостными источниками в лице крупных изданий.

К сожалению, для нейросетевого подхода в общем случае не удалось получить качество лучше, чем для случайного леса. Однако в частном случае было получено, что нейросеть лучше прогнозирует движение акций для GAZP и VTBR. Также удалось показать, что нейросеть делает свои предсказания обосновано путем визуализации матриц внимания и их содержательной интерпретации.

Также при написании работы было реализовано 5 парсинговых программ для получения новостей из использованных источников. Код для них хранится в свободном доступе и может быть использован исследователями в дальнейшем. Более того, все собранные данные также хранятся в свободном доступе.

В качестве идей для дальнейших исследований возможностей машинного обучения прогнозировать движение акций на российском фондовом рынке можно попробовать добавить в модели авторегрессинную компоненту. Также в модели можно добавить временные ряды биржевых товаров и валюты.

7 Список литературы

1. Aizawa A. An information-theoretic perspective of tf–idf measures //Information Processing & Management. – 2003. – Т. 39. – №. 1. – С. 45-65. [10]
2. Atsalakis G. S., Valavanis K. P. Surveying stock market forecasting techniques–Part II: Soft computing methods //Expert Systems with applications. – 2009. – Т. 36. – №. 3. – С. 5932-5941.
3. Biau G., Scornet E. A random forest guided tour //Test. – 2016. – Т. 25. – С. 197-227. [9]
4. De Fortuny E. J. et al. Evaluating and understanding text-based stock price prediction models //Information Processing & Management. – 2014. – Т. 50. – №. 2. – С. 426-441.
5. Kannan S. et al. Preprocessing techniques for text mining //International Journal of Computer Science & Communication Networks. – 2014. – Т. 5. – №. 1. – С. 7-16. [3]
6. Kozhevnikov V. A., Pankratova E. S. RESEARCH OF TEXT PRE-PROCESSING METHODS FOR PREPARING DATA IN RUSSIAN FOR MACHINE LEARNING //Theoretical & Applied Science. – 2020. – №. 4. – С. 313-320 [4]
7. Li Y., Pan Y. A novel ensemble deep learning model for stock prediction based on stock prices and news //International Journal of Data Science and Analytics. – 2022. – С. 1-11. [2]
8. Liu J. et al. Transformer-based capsule network for stock movement prediction //Proceedings of the First Workshop on Financial Technology and Natural Language Processing. – 2019. – С. 66-73. [6]
9. Luan Y., Lin S. Research on text classification based on CNN and LSTM //2019 IEEE international conference on artificial intelligence and computer applications (ICAICA). – IEEE, 2019. – С. 352-355.
10. Mittal A., Goel A. Stock prediction using twitter sentiment analysis //Standford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis. pdf). – 2012. – Т. 15. – С. 2352. [13]
11. Natekin A., Knoll A. Gradient boosting machines, a tutorial //Frontiers in neurorobotics. – 2013. – Т. 7. – С. 21. [9]

12. Peng Y., Jiang H. Leverage financial news to predict stock price movements using word embeddings and deep neural networks //arXiv preprint arXiv:1506.07220. – 2015
13. Roll R. Eugene F. Fama, Lawrence Fisher, Michael C. Jensen //Modern Developments in Investment Management: A Book of Readings. – 1978. – С. 177.
14. Rong X. word2vec parameter learning explained //arXiv preprint arXiv:1411.2738. – 2014.
15. Vajrala A. Text Classification. – 2019. [5]
16. Vaswani A. et al. Attention is all you need //Advances in neural information processing systems. – 2017. – Т. 30. [11]
17. Volodin S. N., Kuranov G. M., Yakubov A. P. Impact of Political News: Evidence from Russia //Scientific Annals of Economics and Business. – 2017. – Т. 64. – №. 3. – С. 271-287. [7]
18. Xu Y., Cohen S. B. Stock movement prediction from tweets and historical prices //Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). – 2018. – С. 1970-1979.
19. Fama E. F. et al. The adjustment of stock prices to new information //International economic review. – 1969. – Т. 10. – №. 1. – С. 1-21. [1]

8 Приложения

Приложение 1

Полная таблица для результатов Accuracy классификации на 2 класса алгоритмом случайного леса для Телеграмма.

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	0.22	-1.34	-0.25	0.23	-0.95	-1.11	-4.30
AFLT	-0.97	-0.96	-4.63	-3.63	-1.76	-1.35	-2.33
ALRS	-3.34	-5.59	-1.50	-2.36	-3.71	-0.91	-4.94
CBOM	-3.30	-6.12	-2.88	-5.38	-4.73	-0.37	-0.51
CHMF	8.49	3.94	3.14	0.81	1.94	1.84	-0.36
DSKY	5.55	3.80	4.51	7.38	7.20	5.35	-4.62
GAZP	-0.32	0.68	-0.33	-0.11	-0.31	1.17	-0.27
GMKN	-0.13	-1.52	-0.80	-2.23	-0.57	-1.28	0.77
HYDR	-6.43	0.25	2.34	-4.87	-5.01	-3.02	0.80
IRAO	0.60	1.47	-3.35	-6.70	-4.19	-3.25	-5.22
LKOH	-1.12	-1.46	-1.56	-0.35	-1.61	2.96	0.22
MAGN	5.09	2.61	0.56	1.95	6.94	3.01	2.27
MOEX	0.60	1.67	0.31	-1.94	-1.87	0.42	-1.20
MTSS	-0.32	0.87	0.60	0.88	2.47	1.53	-1.30
NVTK	3.03	0.45	0.04	-2.15	1.12	0.21	1.46
PHOR	-3.50	-1.04	-3.50	-4.73	-2.38	-1.20	-2.77
PIKK	-2.43	-1.67	4.13	-0.80	3.07	-3.57	-3.97
ROSN	0.75	-1.00	1.00	0.79	-2.27	0.13	-0.24
RTKM	-1.04	0.01	0.24	-1.95	0.33	-2.25	0.88
RUAL	-3.79	-0.72	-1.78	-4.71	-3.23	-4.90	-0.90
SBER	-0.70	-0.51	-0.69	-0.62	-1.02	-0.82	-1.74
SNGS	3.73	3.42	3.56	0.13	0.28	0.52	3.51
TATN	-7.84	-13.34	-4.65	-1.10	-1.12	2.71	1.06
TCSG	4.30	0.86	0.89	-0.71	-0.88	-1.86	-3.01

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
TRNFP	-7.27	-4.43	-4.79	-2.11	-6.03	-2.46	-5.93
VTBR	2.10	0.04	3.15	0.12	-0.39	-0.34	-3.02
YNDX	0.75	0.36	-1.97	-5.10	-2.16	-6.86	-2.19

Приложение 2

Полная таблица для результатов Accuracy классификации на 2 класса алгоритмом бустинга над деревьями для Телеграмма.

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	0.17	-1.70	-0.85	-1.73	1.03	-0.75	-4.27
AFLT	0.17	-4.28	-5.64	-3.73	-2.32	1.38	-1.60
ALRS	-0.34	-5.67	0.85	-2.92	-2.40	0.66	-2.76
CBOM	-2.84	-7.97	-6.19	-1.67	-2.75	-6.46	2.32
CHMF	1.68	1.79	0.97	2.58	-0.28	3.18	2.26
DSKY	1.18	1.91	2.45	6.52	5.06	3.27	-7.50
GAZP	-0.27	0.81	0.59	-0.25	0.34	0.32	-0.42
GMKN	-1.72	-5.57	-5.06	-2.41	-1.25	-1.55	-1.26
HYDR	-3.59	0.23	-2.39	-4.60	-0.09	-1.46	4.02
IRAO	-0.28	-2.51	-3.08	-6.15	-2.16	-6.11	-3.09
LKOH	0.23	-1.94	-0.72	-1.63	-0.71	-0.31	0.81
MAGN	0.70	0.19	-1.25	-0.15	0.20	-2.71	-1.85
MOEX	-1.30	-0.13	-0.90	-0.01	-1.41	0.62	-1.48
MTSS	0.06	-2.04	-0.76	-3.71	-0.64	0.58	-3.76
NVTK	2.92	-2.04	-1.04	-1.60	1.18	1.22	-0.35
PHOR	-4.49	-4.42	-7.04	-7.69	-4.01	0.36	-5.82
PIKK	-3.40	-2.75	-0.40	-1.00	1.07	-0.81	-1.97
ROSN	-1.30	-1.36	3.63	2.78	1.46	-1.23	-1.02
RTKM	-0.28	-5.56	-4.57	-3.07	1.31	-1.79	0.62
RUAL	1.35	-3.02	-0.11	-0.25	-2.17	-0.36	1.57

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
SBER	-0.02	0.83	-0.41	-0.31	-2.39	0.18	-0.05
SNGS	1.47	-2.20	3.11	1.62	0.83	2.60	3.39
TATN	-6.99	-6.10	-5.28	-4.23	-2.88	-1.56	-1.54
TCSG	2.48	-1.39	-5.08	-5.65	-2.84	-3.89	-2.58
TRNFP	-1.97	-0.41	-4.42	-1.32	-3.89	-1.22	-3.53
VTBR	-0.84	-2.24	-1.17	0.20	0.26	-0.69	-3.07
YNDX	1.30	-1.64	-0.92	-1.66	-2.68	-2.23	0.15

Приложение 3

Полная таблица для результатов Accuracy классификации на 3 класса алгоритмом случайного леса для Телеграмма.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	1.44	4.62	1.32	0.97	0.16	-0.07
AFLT	-0.78	1.94	2.43	1.44	1.47	2.81
ALRS	0.40	3.43	1.87	-0.38	2.97	2.76
CBOM	-2.28	-3.87	-5.69	4.53	1.21	-0.75
CHMF	-0.63	7.14	7.67	5.11	7.57	0.17
DSKY	6.49	4.24	9.58	14.61	9.35	-0.27
GAZP	1.92	1.02	1.96	2.38	2.44	-0.46
GMKN	-2.29	3.44	1.47	1.80	2.98	-6.20
HYDR	4.52	4.14	6.03	-1.56	1.60	4.14
IRAO	-3.45	-3.86	-4.72	-0.81	-3.87	-5.72
LKOH	-1.07	-0.62	-0.26	0.09	2.38	1.82
MAGN	-2.22	2.53	4.70	2.66	5.22	-0.01
MOEX	1.15	1.39	1.74	1.74	2.46	1.29
MTSS	4.41	2.91	5.10	6.24	1.48	-1.19
NVTK	3.00	-0.67	2.36	2.29	6.05	3.54
PHOR	4.83	1.14	1.29	2.63	1.47	-0.16

	5 min	10 min	15 min	30 min	1 hour	1 day
PIKK	-1.88	4.85	5.43	2.83	2.69	0.07
ROSN	2.84	4.43	4.16	0.54	0.70	1.09
RTKM	1.93	4.88	-2.57	4.31	1.37	-0.45
RUAL	-1.80	2.21	0.17	-0.83	-1.73	-1.39
SBER	-0.23	0.17	3.31	1.13	0.86	-0.84
SNGS	6.96	6.36	4.10	1.36	9.93	8.47
TATN	-5.77	-5.63	1.87	1.23	4.91	3.37
TCSG	-0.43	-2.72	-2.15	-1.24	-0.93	-1.74
TRNFP	3.82	-5.02	-0.85	-6.74	1.10	-0.47
VTBR	-1.26	3.17	2.03	1.97	0.28	-5.52
YNDX	1.85	2.98	2.29	-0.25	-0.48	7.99

Полная таблица для результатов F1 меры для предсказания 0 для алгоритма случайного леса для Телеграмма.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	0.15	0.19	0.18	0.18	0.17	0.11
AFLT	0.13	0.14	0.16	0.17	0.17	0.14
ALRS	0.11	0.24	0.19	0.20	0.26	0.22
CBOM	0.08	0.11	0.10	0.24	0.21	0.16
CHMF	0.07	0.29	0.31	0.26	0.27	0.16
DSKY	0.28	0.33	0.36	0.46	0.39	0.19
GAZP	0.15	0.18	0.17	0.20	0.15	0.15
GMKN	0.10	0.18	0.15	0.18	0.21	0.02
HYDR	0.10	0.11	0.16	0.11	0.15	0.00
IRAO	0.20	0.19	0.14	0.19	0.12	0.09
LKOH	0.10	0.16	0.16	0.18	0.15	0.18
MAGN	0.12	0.17	0.24	0.23	0.25	0.15
MOEX	0.12	0.15	0.19	0.20	0.17	0.10

	5 min	10 min	15 min	30 min	1 hour	1 day
MTSS	0.07	0.12	0.16	0.19	0.14	-0.01
NVTK	0.14	0.16	0.19	0.19	0.17	0.15
PHOR	0.14	0.16	0.19	0.22	0.19	0.16
PIKK	0.06	0.17	0.19	0.14	0.15	0.04
ROSN	0.17	0.19	0.18	0.19	0.18	0.15
RTKM	0.16	0.18	0.12	0.22	0.19	0.04
RUAL	0.16	0.19	0.16	0.18	0.18	0.09
SBER	0.13	0.18	0.18	0.18	0.19	0.11
SNGS	0.10	0.15	0.14	0.18	0.21	0.23
TATN	0.15	0.12	0.21	0.17	0.27	0.04
TCSG	0.11	0.10	0.07	0.19	0.20	0.11
TRNFP	0.05	-0.01	0.07	0.05	0.16	0.07
VTBR	0.09	0.21	0.21	0.19	0.18	0.04
YNDX	0.14	0.18	0.18	0.14	0.14	0.08

Приложение 4

Полная таблица для результатов Accuracy классификации на 3 класса алгоритмом бустинга над деревьями для Телеграмма.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	-0.54	-0.68	0.38	0.18	0.69	-0.34
AFLT	1.09	-1.31	1.52	2.32	1.88	3.72
ALRS	-5.13	-0.27	0.74	1.43	-0.67	-0.16
CBOM	-2.14	-2.58	1.61	1.89	5.16	-3.09
CHMF	-1.18	4.20	6.65	5.15	5.73	-2.33
DSKY	3.44	3.84	8.41	9.97	6.49	0.44
GAZP	0.69	1.45	0.92	2.66	2.13	0.21
GMKN	-2.34	0.63	-0.48	1.12	2.21	-1.64

	5 min	10 min	15 min	30 min	1 hour	1 day
HYDR	-1.51	0.25	2.77	-3.69	-0.46	0.52
IRAO	-0.97	-3.70	-0.71	-4.05	-4.71	-3.11
LKOH	1.56	1.07	1.51	2.24	0.11	1.01
MAGN	-2.96	0.79	-0.03	1.27	-0.41	-1.30
MOEX	-0.32	0.26	-0.66	-0.70	0.55	-2.33
MTSS	-0.39	1.87	2.89	2.28	-0.59	-0.86
NVTK	0.14	0.43	1.81	3.90	0.83	1.39
PHOR	-0.68	-2.03	-2.13	0.92	-2.67	4.07
PIKK	-1.89	0.95	0.32	1.91	-0.39	-5.73
ROSN	-0.25	1.71	1.09	2.37	1.19	1.20
RTKM	-0.94	-2.84	-4.15	3.38	-0.15	-1.69
RUAL	4.21	0.88	1.60	-0.24	-0.69	4.18
SBER	-0.49	0.75	2.28	0.84	0.45	-2.51
SNGS	0.12	1.17	1.20	2.82	3.31	7.30
TATN	-5.23	-4.78	-2.01	0.29	1.07	1.29
TCSG	1.60	0.76	-0.79	-2.94	0.07	-0.71
TRNFP	4.71	0.52	-3.17	-3.34	-1.20	-2.60
VTBR	-2.15	-0.27	-1.86	-0.52	0.10	-4.14
YNDX	-0.45	0.50	1.66	-0.66	1.90	1.47

Полная таблица для результатов F1 меры для предсказания 0 для алгоритма бустинга над деревьями для Телеграмма.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	0.14	0.15	0.17	0.18	0.17	0.10
AFLT	0.15	0.13	0.17	0.19	0.18	0.16
ALRS	0.06	0.21	0.20	0.24	0.24	0.19
CBOM	0.11	0.15	0.20	0.22	0.27	0.15

	5 min	10 min	15 min	30 min	1 hour	1 day
CHMF	0.06	0.26	0.29	0.26	0.26	0.13
DSKY	0.26	0.33	0.35	0.41	0.38	0.20
GAZP	0.14	0.19	0.18	0.22	0.19	0.16
GMKN	0.10	0.16	0.13	0.18	0.21	0.07
HYDR	0.06	0.12	0.17	0.11	0.17	-0.02
IRAO	0.24	0.19	0.19	0.16	0.11	0.12
LKOH	0.13	0.21	0.21	0.23	0.18	0.18
MAGN	0.11	0.15	0.20	0.22	0.20	0.15
MOEX	0.11	0.16	0.17	0.18	0.19	0.08
MTSS	0.05	0.13	0.14	0.17	0.13	0.01
NVTK	0.12	0.18	0.19	0.22	0.14	0.14
PHOR	0.12	0.16	0.19	0.22	0.17	0.23
PIKK	0.11	0.15	0.15	0.13	0.16	0.03
ROSN	0.15	0.18	0.18	0.22	0.21	0.16
RTKM	0.15	0.12	0.11	0.22	0.19	0.05
RUAL	0.22	0.18	0.18	0.19	0.20	0.15
SBER	0.13	0.19	0.20	0.19	0.18	0.10
SNGS	0.06	0.13	0.14	0.22	0.18	0.23
TATN	0.16	0.14	0.18	0.16	0.23	0.02
TCSG	0.14	0.18	0.11	0.16	0.22	0.12
TRNFP	0.06	0.06	0.05	0.09	0.14	0.10
VTBR	0.10	0.20	0.18	0.18	0.20	0.06
YNDX	0.13	0.17	0.19	0.16	0.18	0.05

Приложение 5

Полная таблица для результатов Accuracy классификации на 2 класса алгоритмом случайного леса для Классических новостных источников.

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	-12.93	-9.32	-3.98	-8.05	-3.96	-14.41	-19.67
AFLT	3.06	-1.88	-4.27	-10.32	-4.66	-4.86	-1.23
ALRS	4.46	-1.46	-3.35	2.28	-4.34	-4.97	4.20
CBOM	0.26	-4.95	-1.00	1.37	-7.45	-2.89	-7.21
CHMF	-4.00	3.60	6.76	-1.04	-1.72	2.63	-9.41
DSKY	-28.72	-32.43	-14.56	3.69	-1.37	-7.60	6.38
GAZP	1.59	4.47	3.59	4.78	3.79	3.38	3.12
GMKN	-4.99	-2.17	-5.52	-13.89	-10.16	-5.42	-4.54
HYDR	-0.27	-5.19	-14.34	-10.30	-13.22	-12.97	-17.25
IRAO	-10.12	-5.08	-1.23	-9.78	3.71	1.77	-10.11
LKOH	-6.41	-6.58	-3.32	-7.11	-6.04	-4.23	-4.49
LSRG	-11.35	16.15	-26.52	-27.09	-38.03	-66.10	-39.88
MAGN	-14.09	-7.59	-9.34	-5.07	-0.58	12.72	-15.15
MOEX	1.69	1.88	-0.13	-1.26	-0.05	-0.31	-2.14
MTSS	3.50	-1.34	1.25	-6.69	0.51	-1.18	-20.79
NVTK	0.45	-4.90	-1.93	2.58	-4.40	4.01	-12.30
PHOR	-10.21	-5.08	-1.10	3.61	2.35	-10.50	-6.77
PIKK	1.22	-1.28	-2.28	1.70	-2.28	-3.44	-0.02
ROSN	-4.31	2.91	2.10	-0.49	7.54	-2.94	2.79
RTKM	-11.97	-5.52	-8.70	-13.87	-7.99	-13.99	-4.92
RUAL	-8.36	-11.07	-12.13	-8.74	0.42	-11.15	-16.75
SBER	-0.94	-3.35	-1.72	-0.49	-2.65	-0.05	-5.62
SNGS	-9.22	-7.01	-6.89	-8.97	2.40	-15.99	-6.96
TATN	-13.14	-15.79	-5.48	-11.22	-5.60	-2.49	3.85
TCSG	5.13	8.97	7.02	0.61	-3.29	0.92	-2.32
TRNFP	0.70	-4.84	-14.39	-14.82	-1.25	-1.64	-10.57
VTBR	5.86	0.37	1.86	1.16	1.04	1.73	-0.36

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
YNDX	-2.21	-8.03	-4.44	1.70	-3.66	2.84	-5.12

Приложение 6

Полная таблица для результатов Accuracy классификации на 2 класса алгоритмом бустинга над деревьями для Классических новостных источников.

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	-2.15	-7.50	-1.62	-2.54	-10.30	-7.91	-10.23
AFLT	2.82	-1.44	-4.32	-3.89	-10.18	-6.11	1.01
ALRS	-12.42	-17.89	-17.85	-21.06	-24.61	1.14	1.54
CBOM	-4.12	-8.38	-1.12	-4.75	-5.09	-10.17	4.39
CHMF	-1.30	3.62	4.54	-2.35	-5.85	-4.27	1.76
DSKY	2.40	-9.31	-7.51	-3.56	-1.28	-22.95	-25.85
GAZP	1.77	3.90	4.39	5.30	-5.31	4.06	5.65
GMKN	-5.30	-9.25	-3.63	-13.87	-9.26	-9.67	-3.79
HYDR	-9.94	-2.27	-8.90	-15.88	-18.71	-20.47	-24.10
IRAO	-12.11	-1.24	-6.38	-14.79	4.27	8.47	-15.46
LKOH	-7.75	-8.50	-5.66	-7.21	-8.54	-5.80	-2.75
LSRG	-40.73	-28.11	-40.99	-26.69	-9.38	-38.44	-9.65
MAGN	-21.47	-9.19	-10.13	-3.42	2.98	8.21	-4.51
MOEX	1.88	-1.89	-1.94	-3.19	-3.16	-6.92	-3.36
MTSS	-1.98	-5.77	-14.14	-17.60	-17.09	-9.34	-12.43
NVTK	-5.49	-9.16	-0.97	-7.85	-3.37	-0.41	-6.33
PHOR	2.56	-8.72	-8.94	-10.45	2.12	-4.85	-10.68
PIKK	-0.25	-2.65	3.12	-1.92	-1.89	-2.28	0.08
ROSN	-2.31	-1.81	5.77	9.20	1.93	6.16	1.53
RTKM	-7.06	-11.93	-13.84	-1.10	-10.18	-10.04	-2.61
RUAL	-11.55	-11.88	0.66	-9.05	9.02	-2.53	-1.99
SBER	-4.96	-2.65	-2.13	-1.51	-1.69	-2.15	-3.81

	1 min	5 min	10 min	15 min	30 min	1 hour	1 day
SNGS	-9.01	-2.00	-6.60	4.67	2.22	-10.74	-13.05
TATN	-9.51	-1.60	-13.40	-15.26	-4.01	-8.33	-0.60
TCSG	-8.09	-10.96	-7.74	-0.51	-0.34	1.05	-4.79
TRNFP	4.44	10.54	-3.08	-10.81	-1.19	-7.14	-10.97
VTBR	3.35	0.28	0.55	3.24	-1.67	-0.41	-0.16
YNDX	-1.10	-0.81	-5.29	2.21	-0.23	3.07	-1.48

Приложение 7

Полная таблица для результатов Accuracy классификации на 3 класса алгоритмом случайного леса для Классических новостных источников.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	-16.31	-3.20	-12.75	-11.49	-3.74	-10.51
AFLT	-5.97	-10.27	-3.21	2.06	1.25	-4.03
ALRS	4.42	-13.07	-14.56	2.68	-0.74	-0.90
CBOM	2.33	-1.91	-4.42	-9.43	-11.84	-5.98
CHMF	-6.21	-0.50	1.29	-5.32	-2.69	-5.08
DSKY	-20.00	-33.30	-10.60	5.96	-0.81	-24.31
GAZP	2.59	6.07	6.29	6.38	-6.34	-0.04
GMKN	2.42	-10.64	-3.65	-2.44	-0.26	-4.26
HYDR	-11.92	-10.84	-6.59	-13.50	-2.32	-16.17
IRAO	-11.07	-6.33	-11.70	-1.74	-4.73	0.84
LKOH	-2.24	-0.25	-0.21	6.51	-0.88	-5.93
MAGN	-18.21	-7.93	-8.78	-12.02	-3.15	-12.94
MOEX	-6.01	-5.12	-4.76	-1.70	2.45	-3.14
MTSS	5.63	0.53	13.68	1.16	6.05	-13.90
NVTK	-4.97	-2.46	4.12	-3.37	-0.73	-2.72
PHOR	0.85	-3.59	-10.82	-7.34	7.64	2.35
PIKK	-4.58	-2.91	-3.91	-5.92	-5.57	-3.88

	5 min	10 min	15 min	30 min	1 hour	1 day
ROSN	-1.98	-2.69	-0.95	1.52	0.39	0.41
RTKM	-11.88	6.76	4.39	0.56	3.95	-11.96
RUAL	3.85	5.85	-5.97	-6.42	-14.46	0.57
SBER	1.13	2.21	2.34	3.10	3.24	6.72
SNGS	-14.74	13.32	-1.27	-14.81	-10.00	-12.58
TATN	-20.84	-14.49	-11.69	-4.23	-13.88	-17.00
TCSG	1.25	-0.18	0.86	9.25	3.58	1.47
TRNFP	3.94	-4.85	13.69	-3.85	-1.50	-16.16
VTBR	2.63	9.73	10.50	10.37	10.02	-10.70
YNDX	4.10	7.26	8.00	4.30	2.11	-1.87

Полная таблица для F1 меры для предсказания 0 для алгоритма случайного леса для Классических новостных источников.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	-0.01	0.14	0.08	0.01	0.12	0.05
AFLT	0.06	0.00	0.07	0.16	0.15	0.05
ALRS	0.25	0.09	0.11	0.35	0.30	0.16
CBOM	0.05	0.10	0.06	0.08	0.06	0.06
CHMF	0.15	0.20	0.24	0.15	0.10	0.12
DSKY	-0.00	-0.09	0.05	0.30	0.00	-0.03
GAZP	0.10	0.13	0.12	0.15	0.06	0.10
GMKN	0.14	0.06	0.16	0.16	0.14	0.16
HYDR	-0.16	0.09	0.17	0.14	0.23	-0.12
IRAO	0.11	0.17	0.03	0.14	0.09	0.13
LKOH	0.07	0.20	0.21	0.24	0.19	0.13
MAGN	0.03	0.13	-0.00	0.12	0.16	0.10
MOEX	-0.00	0.03	0.03	0.14	0.19	0.06

	5 min	10 min	15 min	30 min	1 hour	1 day
MTSS	0.22	0.24	0.31	0.27	0.28	-0.01
NVTK	0.13	0.19	0.28	0.18	0.21	0.07
PHOR	0.20	0.16	0.07	0.12	0.22	0.01
PIKK	0.08	0.13	0.10	0.05	0.08	0.08
ROSN	0.14	0.22	0.23	0.25	0.22	0.11
RTKM	0.01	0.17	0.21	0.20	0.20	0.04
RUAL	0.18	0.23	0.18	0.16	0.14	0.06
SBER	0.11	0.18	0.20	0.24	0.23	0.17
SNGS	0.12	0.37	0.22	0.06	0.11	0.00
TATN	-0.08	-0.02	0.02	0.07	-0.11	-0.06
TCSG	0.15	0.21	0.16	0.28	0.16	0.21
TRNFP	0.06	0.12	0.19	0.04	0.08	0.01
VTBR	0.15	0.27	0.30	0.28	0.29	-0.10
YNDX	0.22	0.24	0.24	0.18	0.14	-0.06

Приложение 8

Полная таблица для результатов Accuracy классификации на 3 класса алгоритмом бустинга над деревьями для Классических новостных источников.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	-10.07	-10.45	-6.15	-7.39	-1.13	-2.09
AFLT	-4.73	-10.86	-1.93	-5.72	6.25	0.52
ALRS	-17.92	-16.83	-8.77	-14.82	-0.24	10.99
CBOM	-15.46	-9.77	1.63	-1.04	-10.99	-11.06
CHMF	-9.53	-2.67	1.90	-2.81	0.77	2.10
DSKY	18.48	-18.10	13.18	6.70	-9.14	-23.21
GAZP	3.06	1.39	4.85	3.39	6.80	3.99
GMKN	-0.93	2.77	-3.00	-2.23	1.21	-13.80
HYDR	-9.26	-10.16	-10.76	-12.16	-9.38	-15.95

	5 min	10 min	15 min	30 min	1 hour	1 day
IRAO	-2.22	-12.61	-16.29	-5.43	-7.89	-10.84
LKOH	-0.81	1.45	-0.62	-3.00	3.10	-1.46
MAGN	-20.10	0.34	-1.84	-4.81	-10.24	-0.25
MOEX	-6.76	-5.52	-6.33	1.22	3.40	-2.96
MTSS	6.61	-5.42	-0.68	4.91	-5.24	-15.01
NVTK	-2.37	-9.80	-5.62	-3.12	-7.16	-7.48
PHOR	-2.67	-10.93	-12.22	0.14	-5.47	0.66
PIKK	-6.14	-2.28	-1.64	-4.95	-6.26	-1.88
ROSN	-0.90	-5.16	-0.40	-0.10	-0.55	-6.22
RTKM	-11.79	-0.83	-7.58	1.99	-2.64	-3.76
RUAL	-7.59	-11.84	-12.58	1.65	-12.40	-5.43
SBER	-3.06	-1.49	0.93	3.40	3.22	-3.73
SNGS	-14.39	0.00	-5.60	-22.04	-1.23	-4.87
TATN	-13.52	-16.18	-15.89	-4.56	-13.77	-5.06
TCSG	-1.76	4.47	7.73	4.79	0.61	0.17
TRNFP	-10.44	-7.16	-8.08	-9.34	-3.39	-22.23
VTBR	-1.60	5.80	8.41	9.73	6.19	-8.49
YNDX	-0.32	6.43	8.03	7.54	0.23	-5.34

Полная таблица для F1 меры для предсказания 0 для алгоритма случайного леса для Классических новостных источников.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	0.05	0.09	0.12	0.11	0.19	0.16
AFLT	0.07	-0.01	0.08	0.08	0.21	0.10
ALRS	0.08	0.13	0.18	0.18	0.29	0.28
CBOM	-0.02	0.07	0.16	0.18	0.07	0.03
CHMF	0.12	0.20	0.28	0.20	0.17	0.21
DSKY	0.53	0.12	0.23	0.32	-0.00	0.02

	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	0.12	0.11	0.12	0.13	0.16	0.12
GMKN	0.13	0.20	0.17	0.15	0.15	0.07
HYDR	-0.07	0.12	0.14	0.16	0.19	-0.12
IRAO	0.19	0.09	-0.02	0.10	0.08	0.09
LKOH	0.09	0.23	0.20	0.15	0.24	0.20
MAGN	0.02	0.27	0.16	0.23	0.14	0.23
MOEX	0.05	0.07	0.06	0.17	0.21	0.09
MTSS	0.24	0.18	0.17	0.31	0.15	-0.03
NVTK	0.16	0.13	0.18	0.19	0.13	0.06
PHOR	0.17	0.12	0.06	0.21	0.11	0.10
PIKK	0.07	0.13	0.11	0.06	0.07	0.13
ROSN	0.14	0.20	0.25	0.24	0.21	0.04
RTKM	-0.02	0.08	0.09	0.22	0.15	0.14
RUAL	0.08	0.06	0.14	0.27	0.20	0.06
SBER	0.08	0.15	0.18	0.24	0.21	0.07
SNGS	0.12	0.23	0.17	0.02	0.19	0.11
TATN	0.03	0.02	-0.00	0.07	-0.10	0.13
TCSG	0.14	0.27	0.26	0.28	0.18	0.21
TRNFP	-0.03	0.11	0.04	-0.02	0.12	-0.09
VTBR	0.12	0.25	0.28	0.29	0.27	-0.07
YNDX	0.20	0.28	0.28	0.26	0.16	-0.07

Приложение 9

Результаты эффекта для нейросетевого подхода по F1 мере по сравнению с предсказанием нейтрального класса на для задачи 3-ех классов на новостях из Телеграмма.

	5 min	10 min	15 min	30 min	1 hour	1 day
AFKS	0.05	0.04	0.04	0.08	0.07	0.05

	5 min	10 min	15 min	30 min	1 hour	1 day
AFLT	-0.00	0.09	0.16	0.18	0.12	0.12
ALRS	0.08	0.15	0.12	0.16	0.19	0.13
CHMF	0.01	-0.03	-0.02	0.07	0.04	0.04
DSKY	0.02	0.04	0.04	0.08	0.11	0.04
GAZP	0.11	0.15	0.09	0.20	0.19	0.09
HYDR	0.05	0.06	0.07	0.15	0.07	0.01
LSRG	-0.08	-0.07	-0.03	-0.03	-0.07	-0.04
MAGN	0.15	0.10	0.16	0.20	0.20	0.01
MOEX	0.03	0.04	-0.04	-0.00	0.07	-0.05
MTSS	-0.01	0.06	0.07	0.09	0.02	-0.09
NVTK	0.00	0.19	0.11	0.13	0.13	0.02
PHOR	-0.14	0.02	0.01	-0.01	0.01	0.09
PIKK	-0.00	0.04	0.10	0.12	0.14	-0.01
ROSN	-0.00	0.15	0.16	0.18	0.16	-0.00
SNGS	0.03	0.08	0.09	0.14	0.19	0.13

Приложение 10

Результаты эффекта для нейросетевого подхода по F1 мере по сравнению с предсказанием нейтрального класса на для задачи 3-ех классов на новостях из Классчиеских источников.

	5 min	10 min	15 min	30 min	1 hour	1 day
GAZP	0.07	0.14	0.14	0.21	0.19	-0.00
MTSS	-0.01	0.02	0.01	-0.00	0.00	-0.05
RTKM	-0.17	-0.04	0.09	0.08	0.08	0.13
SBER	0.09	0.09	0.10	0.10	0.10	0.02
TCSG	-0.14	-0.06	-0.01	0.00	-0.02	0.03
VTBR	0.04	0.21	0.23	0.27	0.25	0.00
YNDX	0.09	0.05	0.05	0.01	0.09	-0.12