ОРИГИНАЛЬНАЯ СТАТЬЯ

DOI: 10.26794/3033-7097-2025-1-4-6-15
УДК 004.832.2:33(045)

Современные методы обработки документов для расчета биржевых индикаторов

Э.Ф. Болтачев, А.И. Тюляков
Финансовый университет при Правительстве Российской Федерации, Москва, Российская Федерация

АННОТАЦИЯ

В данной статье рассматриваются современные методы экстраполяции предобученных трансформеров, направленные на повышение их способности обрабатывать длинные, а также короткие текстовые последовательности на русском языке в финансовой сфере. Особое внимание уделяется задаче классификации текстов, отражающих ожидания брокерских аналитиков относительно движения рынка (ожидание роста, падения либо неопределенности изменения). Для решения данной задачи исследуется применение облегченных языковых моделей ruBERT-tiny1 и ruBERT-tiny2, адаптированных для эффективной работы с большим объемом входных данных при сохранении качества предсказаний. В работе анализируются различные подходы к расширению контекстного окна моделей, включая методы экстраполяции, а также рассматривается влияние стратегий токенизации, векторизации и эмбеддингов на итоговые результаты классификации. Дополнительно обсуждаются особенности применения трансформеров в условиях повышенной волатильности рынка и изменяющихся новостных потоков, что позволяет глубже оценить устойчивость предлагаемых решений. Кроме того, предлагается и обсуждается формула расчета опережающего индикатора для биржевых рынков, демонстрирующая практическую значимость использования трансформерных моделей в анализе финансовых текстов и формировании аналитических метрик. **Представленные результаты** подчеркивают перспективность применения компактных трансформеров в задачах предиктивной финансовой аналитики. Пул брокеров образует выборку мнений в виде текста с определенным смыслом, последовательность слов позволяет оценивать возможные ожидания на финансовом рынке совершенно нелинейным методом. Решение задачи обработки длинных последовательностей токенов актуально, конкретного универсального метода решения данной проблемы нет. Одним из вариантов решения задачи обработки естественного языка NLP на практике является ряд предобученных языковых моделей. Применение предобученных языковых моделей позволяет решать различные задачи классификации, исследуя тексты различных контекстов. В рамках исследования применяется метод экстраполяции предобученных трансформеров для изучения точности классификации и времени обучения, в зависимости от количества токенов в контекстном окне модели. Полученные данные могут быть использованы для дальнейших исследований и построения математической модели расчета опережающих индикаторов на рынке.

Ключевые слова: токенизация; токены; языковые модели; эскстраполяция; последовательность; векторизация; эмбеддинги

Для цитирования: **Болтачев Э.Ф., Тюляков А.И. Современные методы обработки документов для расчета биржевых индикаторов. *Цифровые решения и технологии искусственного интеллекта.* 2025;1(4):6-15. DOI: 10.26794/3033-7097-2025-1-4-6-15**

ORIGINAL PAPER

Modern Methods of Document Processing for Calculating Stock Market Indicators

E.F. Boltachev, A.I. Tyulyakov
Financial University under the Government of the Russian Federation, Moscow, Russian Federation

ABSTRACT

This article discusses modern methods of extrapolating pre-trained transformers aimed at improving their ability to process long and short text sequences in Russian in the financial sector. Particular attention is paid to the task of classifying texts that reflect broker analysts' expectations regarding market movements (expectations of growth, decline, or uncertainty of change). To solve this problem, the application of lightweight language models ruBERT-tiny1 and ruBERT-tiny2 is investigated, which are adapted to work effectively with large amounts of input data while maintaining prediction quality. The paper analyzes various approaches to expanding the contextual window of models, including extrapolation methods, and considers the impact of tokenization, vectorization, and embedding strategies on the final classification results. Additionally, the paper discusses the peculiarities of using transformers in conditions of

increased market volatility and changing news flows, which allows for a more in-depth assessment of the stability of the proposed solutions. Furthermore, a formula for calculating a leading indicator for stock markets is proposed and discussed, demonstrating the practical significance of using transformer models in the analysis of financial texts and the formation of analytical metrics. **The presented results** highlight the promising application of compact transformers in predictive financial analytics tasks.
Keywords: tokenization; tokens; language models; extrapolation; sequence; vectorization; embeddings

For citation: **Boltachev E.F., Tyulyakov A.I. Modern methods of document processing for calculating stock market indicators.** ***Digital Solutions and Artificial Intelligence Technologies.*** **2025;1(4):6-15. DOI: 10.26794/3033-7097-2025-1-4-6-15**

ВВЕДЕНИЕ

В последние годы технологии обработки естественного языка (Natural Language Processing, NLP) все более прочно интегрируются в процессы анализа и обработки данных в режиме реального времени. Данная технология применяется в различных сферах, где требуется анализ и классификация текста, а также регрессия на основе текстовых данных. Спектр применения достаточно широкий, начиная от образовательной сферы и заканчивая финансовой [1–4].

Особенно предметом интереса выступает финансовая сфера применения ML в целом и NLP-методологии в отдельности. Однако в данной статье не стоит вопрос решения задачи регрессии на примере расчета возможной стоимости акции. Вместо этого есть возможность использовать индикаторы, которые будут выполнять задачу описания возможных ожиданий, связанных с изменениями стоимости акций.

Основной гипотезой выступает доверие брокерскому сообществу, которое в режиме реального времени пишет, анализирует и выкладывает свои оценки в открытом доступе. Данный пул брокеров образует выборку мнений. При этом каждый текст несет в себе определенный смысл, последовательность слов позволяет оценивать возможные ожидания на рынке совершенно нелинейным методом.

Решение задачи обработки длинных последовательностей токенов актуально, конкретного универсального метода решения данной проблемы нет. Одним из вариантов решения задачи NLP на практике является ряд предобученных языковых моделей. Предобученные трансформеры имеют возможность долговременного анализа ряда токенов, подающихся на вход модели. Применение предобученных языковых моделей позволяет решать различные задачи классификации, исследуя тексты различных контекстов. Языковые модели позволяют решать ряд сложных задач при использовании определенного количества ресурсов, зависящих от задачи.

Однако возникают ограничения, препятствующие реализации ряда задач, в которых число токенов имеет большую размерность, чем позволяет обрабатывать модель. Также бывает обратная ситуация, когда мощная модель принимает порядка 2048 входных последовательностей токенов, а в самих текстах число токенов меньше. Классификация данных оценок позволит сделать агрегацию значений по временным рядам. Полученные данные могут быть использованы для дальнейших исследований и построения математической модели расчета опережающих индикаторов на рынке.

Целью настоящей работы является анализ современных методологий токенизации и классификации русскоязычных текстов финансовых новостей, подготовленных биржевыми аналитиками, с использованием моделей ruBERT-tiny-1 и ruBERT-tiny-2. В рамках исследования применяется метод экстраполяции предобученных трансформеров для изучения точности классификации и времени обучения в зависимости от количества токенов в контекстном окне модели.

МЕТОДОЛОГИЯ ТОКЕНИЗАЦИИ ТЕКСТОВЫХ ДОКУМЕНТОВ

При решении задачи NLP общим алгоритмом действий является первоначальная обработка данных, а именно заключение текстовых данных в токены с последующей векторизацией для подачи в модель. Данный этап не имеет универсального решения, и зачастую проводятся эмпирические эксперименты, которые изучают эффективность тех или иных методов. Сам процесс сильно влияет на производительность обучения, как на точность метрик, так и на количество требуемых ресурсов для обучения.

Токенизация текста — первоначальная задача при реализации NLP. Она является фундаментальным этапом работы. В общем случае токенизация представляет собой процедуру разбиения текстовых данных на подмножества, или токены. От специфики задачи методы токенизации варьируются [2]. В качестве типичного примера токенизации можно привести разбиение текста на слова, подслова, символы. Данный метод является простейшим по реализации и во многих случаях с помощью него можно добиться высокой предиктивной способности модели. Однако избыточное количество сформиро-

ванных токенов приводит к повышенным вычислительным затратам и во многих случаях оказывается нерациональным с точки зрения эффективности обработки данных. К базовым методам токенизации также относится разбиение текстовых данных на n-граммы. Такой подход нередко рассматривается как эффективное средство сохранения семантического содержания последовательности слов.

После проведения ряда процессов по обработке текстов выбранными методами токенизации возникает проблема использования получившихся токенов в языковых моделях. Данную проблему решают методы векторизации токенов. Для решения задач обработки естественного языка (NLP) применяются современные методы токенизации, среди которых наиболее распространенными являются BPE (Byte-Pair Encoding), WordPiece и Unigram Tokenization.

Алгоритм BPE представляет собой последовательное обучение на символах корпуса текста [5]. Изначально токеном является символ, затем производится подсчет частот биграмм, наиболее часто встречаемая биграмма становится новым токеном. Таким образом, в процессе итеративного обучения модели словарный запас расширяется, тогда как количество токенов, необходимых для представления текста, уменьшается. Главный плюс такого подхода состоит в возможности работать с неизвестными словами. Данный алгоритм позволяет обрабатывать неизвестные слова разбиением на токены (символы) и повторением процедуры. Алгоритм BPE обеспечивает адаптивность и эффективность работы. Он предоставляет возможность обрабатывать редкие слова, поддерживая подсловные единицы (суффиксы, префиксы и т.д.).

Алгоритм WordPiece является основой метода BPE, имеющий аналогичный механизм действий [6]. Unigram отличается от приведенных выше алгоритмов токенизации [7] — он считает каждый токен независимым от токенов до него. Это самая простая языковая модель, в том смысле, что вероятность токена X, учитывая предыдущий контекст, является просто вероятностью токена X. На каждом этапе обучения алгоритм Unigram вычисляет потери по корпусу с учетом текущей лексики. Затем для каждого символа в словаре алгоритм вычисляет, насколько увеличится общая потеря, если символ будет удален, и ищет символы, которые увеличат ее меньше всего.

В конкретных задачах Unigram в сравнении с BPE может оказаться лучше, однако его наибольшими минусами является проблема с пунктуацией и потеря семантического контекста. В целом данный алгоритм встречается редко.

Существуют основные подходы к токенизации, предусматривающие использование плотных векторных представлений, которые позволяют сохранять семантическую информацию о словах [8]. Использование эмбеддингов решает ряд задач, преобразуя категориальные признаки — текст в числовой формат для использования при обучении модели. Основным преимуществом эмбеддингов является возможность обучения модели пониманию смыслового содержания текста за счет захвата семантических отношений между токенами. При этом производительность модели увеличивается за счет снижения размерности. Однако часто возникает проблема несоответствия количества токенов, которые необходимо обработать, с максимальным числом токенов, которые подаются в языковую модель. Это связано со сложностью вычислений, поскольку стандартные архитектуры трансформеров имеют главную уязвимость — квадратичную сложность вычислений. Отсюда идея применения предобученных трансформеров в задачах с ограниченными ресурсами становится непрактичной.

Существует ряд известных методов по борьбе с данной проблемой. Примерами более универсальных методов являются чанкинг (chunking), иерархический подход с разными уровнями абстракции. Чанкинг (chunking) в NLP — это задача разделения последовательности слов (обычно предложения) на фрагменты (chunks), которые представляют собой синтаксически связанные группы слов, например именные группы (noun phrases, NP), глагольные группы (verb phrases, VP) и предложные группы (prepositional phrases, PP). Это промежуточный этап между токенизацией (разбиением текста на отдельные слова) и синтаксическим анализом (полным построением дерева разбора). Чанкинг обычно используется для извлечения информации, анализа настроений и других задач NLP.

Правильный чанкинг (rule-based chunking). Этот подход использует заранее определенные правила, основанные на грамматических знаниях и лексических признаках слов (часть речи, суффиксы и т.д.). Правила могут быть представлены в виде набора продукций (productions) в формализме контекстно-свободных грамматик (CFG).

Пример CFG правил для чанкинга:
NP — Det N (именная группа состоит из определителя и существительного);
NP — Adj N (именная группа состоит из прилагательного и существительного);
NP — N (именная группа состоит из одного существительного);
VP — V NP (глагольная группа состоит из глагола и именной группы);

PP — P NP (предложная группа состоит из предлога и именной группы),
где NP — именная группа; VP — глагольная группа; PP — предложная группа; Det — определитель (например, «the», «a»); Adj — прилагательное; N — существительное; V — глагол; P — предлог.

Иерархический подход к обработке токенов в NLP предполагает построение представлений на разных уровнях абстракции, начиная от отдельных токенов и заканчивая сложными семантическими единицами. Каждый уровень использует информацию с предыдущих уровней для создания более богатого и контекстуально-зависимого представления. Рассмотрим несколько уровней.

1. Уровень токенов (Word Embeddings).

На данном уровне каждый токен (слово) представляется вектором, как правило, сформированным с использованием моделей word2vec, GloVe или FastText.

2. Уровень n-грамм (N-gram embeddings).

Этот уровень объединяет последовательности из n токенов. Простейший подход — усреднение векторов токенов. Более сложные подходы могут использовать рекуррентные нейронные сети (RNN) или сверточные нейронные сети (CNN) для получения более контекстно-зависимых представлений n-грамм.

3. Уровень предложений (Sentence Embeddings).

Предложение представляется как последовательность токенов или n-грамм. Для получения векторного представления предложения можно использовать усреднение векторов токенов или n-грамм. Рекуррентные нейронные сети (RNN, LSTM, GRU) формируют представление предложения путем последовательной обработки токенов, при этом конечное скрытое состояние используется в качестве его векторного отображения. В архитектурах на основе трансформеров взаимосвязи между всеми токенами учитываются посредством механизма самовнимания, а итоговое представление предложения часто задается вектором специального токена [CLS].

4. Семантический уровень (Semantic Role Labeling, Relation Extraction).

Информация данного уровня ориентирована на извлечение семантических отношений между словами и предложениями. В частности, она включает идентификацию ролей участников ситуации — таких как субъект, объект и предикат (Semantic Role Labeling), а также выявление отношений между сущностями в тексте (Relation Extraction). Это часто включает использование графов знаний или обучение моделей классификации. Математическое описание на данном этапе определяется используемым методом, однако в типичном случае включает меры подобия или вероятностные модели. Этот иерархический подход позволяет эффективно использовать информацию на разных уровнях абстракции для решения различных задач NLP, таких как классификация текста, машинный перевод, извлечение информации и понимание естественного языка. Выбор конкретных методов на каждом уровне зависит от задачи и доступных ресурсов.

С другой стороны, решением данной задачи является методология экстраполяции предобученных трансформеров для обработки длинных последовательностей текстов [7]. Данный подход позволит наиболее корреляционным образом сократить количество токенов, не потеряв основные семантические связи. Реализация методологии экстраполяции осуществлена с использованием метода lsg converter [7].

ОПИСАНИЕ ДАТАСЕТА

Датасет обучения включает в себя 30 780 записей биржевых аналитиков об ожидании изменения индекса Мосбиржи, brent, золота. Однако в каждой статье аналитика встречается множество смежных областей биржи, дополняющих семантическое поле модели.

Таргетом является один из трех классов. В рассматриваемой задаче количество классов составляет три. Два из них отражают прогнозируемое направление изменения показателя: рост и снижение, что позволяет модели различать позитивные и негативные ожидания участников рынка. Третий класс обозначает неопределенность, учитывая случаи, когда прогноз не позволяет однозначно отнести событие к росту или снижению. Включение данного класса обеспечивает более точное моделирование реальных рыночных условий и позволяет анализировать ситуацию с учетом степени прогнозной неопределенности. Данные классы сигнализируют модели анализ текста для определения будущих изменений, а не текущих. Это позволяет агрегировать предсказанные таргеты во временных рядах с расчетом определенных лагов в прогнозировании во времени на этапе применения математической модели расчета опережающих индикаторов на бирже.

Записи датасета, собранные парсером из открытых источников, таких как investing.com, а также finam.ru, представлены в *табл. 1*. Разметка классов датасета проводилась вручную.

В датасете имеется дизбалансировка классов (см. *табл. 1*), класс понижения заключает в себе на 15–20% меньше количества записей от других

Таблица 1 / Table 1

Балансировка классов / Class Balancing

Класс / Class	Количество записей / Number of Data Points
0 (повышение)	10787
1 (неопределенность)	11263
2 (понижение)	8730

Источник / Source: составлено авторами / Complied by the authors.

классов. Также проблемой является большое количество токенов, поскольку записи с парсеров не являются детерминированными и включают в себя весь текст аналитика на определенную тему, а также на смежные сферы биржи.

Необходимо сформировать корпус текстов аналитических статей, опубликованных в открытых источниках, таких как Finam и Investing. Для каждой статьи предполагается последующее присвоение классовой метки, отражающей ожидаемое направление динамики рынка: рост, понижение или отсутствие существенных изменений.

В итоговом датасете, на котором будет проводиться обучение и анализ предиктивной способности модели, каждый текстовый документ имеет в среднем 265 слов, медианно 240 слов, количество токенов в среднем составляет 345.

УСЛОВИЯ ПРОВЕДЕНИЯ ЭКСПЕРИМЕНТА

Текст должен быть очищен от лишних символов, HTML-тегов и прочих артефактов. Можно использовать стандартные методы предобработки текста: токенизация, лемматизация, удаление стоп-слов. Однако для BERT-ru лемматизация необязательна, поскольку модель уже учитывает морфологические особенности слов.

Данные нужно разделить на три части: тренировочный набор (например, 70–80%); валидационный набор (10–15%) и тестовый набор (10–15%). Валидационный набор используется для настройки гиперпараметров модели и предотвращения переобучения.

Используем BERT-ru модель, например ‘ruBERT-tiny2’, ‘bert-base-russian-case-sensitive’ или ‘sberbank-ai/ruBert-base’. Выбор зависит от доступных ресурсов (памяти и вычислительной мощности) и ожидаемого качества. Обучение и анализ полученных метрик будет осуществлен путем использования моделей ‘ruBERT-tiny2’, а также ‘ruBERT-tiny1’. Необходимо подобрать оптимальные значения гиперпараметров, такие как размер батча, скорость обучения, количество эпох обучения. Это делается с помощью экспериментального поиска на валидационном наборе.

В данном случае это будет модель классификации текста на основе BERT. Выходной слой состоит из трех нейронов (по числу классов: рост; понижение; неизменность), с функцией активации softmax для получения вероятностей принадлежности к каждому классу. Процесс обучения заключается в подаче тренировочного набора данных в модель BERT и корректировке весов модели с целью минимизации функции потерь (например, кросс-энтропии). В процессе обучения необходимо отслеживать показатели на валидационном наборе, чтобы избежать переобучения. Метрики: accuracy, precision, recall, F1-score. В качестве основной метрики качества модели выступает F1-score.

ПРОВЕДЕНИЕ ЭКСПЕРИМЕНТОВ С ИСПОЛЬЗОВАНИЕМ МЕТОДА ЭКСТРАПОЛЯЦИИ ПРЕДОБУЧЕННОЙ МОДЕЛИ RUBERT-TINY2, RUBERT-TINY1

В ходе исследования метода экстраполяции предобученных трасформеров был проведен ряд экспериментов на модели ruBERT-tiny2 с различным количеством входной последовательности токенов модели. Эксперименты проводились с использованием облачных вычислений в среде разработки Google Colab, с использованием графического процессора T4. Сама модель ruBERT-tiny2 [10] имеет входную последовательность 2048 токенов, а также мощный семантический словарь в 83828 токенов.

Выбор данной модели обусловлен релевантным соотношением качества и затрачиваемых ресурсов для обучения. На примере описываемой задачи NLP в статье рассмотрены случаи с использованием 128, 512, 1024 входных токенов вместо 2048.

При использовании оригинального количества входных токенов были достигнуты следующие метрики (*табл. 2*).

Модель с 2048 входными токенами (см. *табл. 2*) достигла метрики 0,675, в рамках данного датасета это удовлетворительный результат, однако само обучение модели проводилось 55 мин 42 с. В контексте решения рассматриваемой задачи при ограниченном объеме ресурсов данная модель демонстрирует низкую эффективность. В связи с этим следующим этапом эксперимента стало сокращение числа входных токенов с использованием библиотеки LSG Converter до 1024 токенов. В результате полученная метрика составила 0,669, что ниже точности модели с 2048 входными токенами. Тем не менее время обучения модели уменьшилось с 55 мин 42 с до 28 мин 58 с.

Таблица 2 / Table 2

Результаты экспериментов на модели ruBERT-tiny2 с различным количеством входной последовательности токенов / Experimental Results for the RuBERT-tiny2 Model with Different Quantity of Input Length of the Sequence

Выборки / Splits*	Метрики / Metrics				
	Eval loss	Eval accuracy	Eval F1	Eval Precision	Eval Recall
Модель с 2048 входными токенами					
Train	0,559016	0,765646	0,758698	0,765958	0,756780
Val	0,714382	0,674119	0,667998	0,675303	0,666474
Test	0,704781	0,679361	0,675110	0,681473	0,674465
Модель с 1024 входными токенами					
Train	0,563955	0,764347	0,761114	0,764576	0,763576
Val	0,709973	0,673729	0,670612	0,673790	0,672578
Test	0,704570	0,672082	0,669810	0,674300	0,671864
Модель с 512 входными токенами					
Train	0,577549	0,763447	0,757664	0,762728	0,756614
Val	0,710775	0,671650	0,666917	0,671849	0,665786
Test	0,695135	0,681180	0,676933	0,683720	0,676570
Модель с 128 входными токенами					
Train	0,599955	0,739195	0,735903	0,736832	0,736963
Val	0,756285	0,640322	0,638574	0,639775	0,639736
Test	0,730899	0,661684	0,660281	0,660808	0,660871

Источник / Source: составлено авторами / Complied by the authors.

Примечание / Note: * Выборки: обучающая, валидационная и тестовая / Splits: train, validation and test.

Метод экстраполяции предобученных трансформеров позволяет варьировать значение количества входных токенов модели в зависимости от задачи.

Следующим экспериментом является сокращение до 512 входных токенов. Метрика качества равна 0,676 (см. *табл. 2*), время обучения модели 15 мин 19 с. В данном случае удалось добиться незначительно большей метрики, чем на модели с 2048 входными токенами, а также ускорить процесс обучения в 3,6 раза.

Также рассматривалась модель с 128 входными токенами. Для данной конфигурации значение метрики F1-score составило 0,66, что ниже по сравнению с предыдущими вариантами, тогда как время обучения модели сократилось до 2 мин 59 с (см. *табл. 2*).

Согласно полученным данным после проведения эксперимента экстраполяции предобученных трансформеров (*рис. 1, 2, табл. 3*) немодифицированная модель трансформеров ruBERT-tiny2 (2048 входных токенов) при решении данной задачи не является оптимальной. В данном случае наиболее удачным решением стала модель с 512 токенами входной последовательности. Она сочетает качество модели, а также удовлетворительное количество затраченных ресурсов при обучении.

В *табл. 4* представлены результаты экспериментов, проведенных на моделях ruBERT-tiny1 и ruBERT-tiny2. Модель ruBERT-tiny1 отличается меньшей сложностью, обладая сокращенным словарем и входной последовательностью длиной 512 токенов. В ходе эксперимента получены данные, свидетельствующие о возможности увеличения контекстного окна с использованием метода LSG Converter, что сопровождается ростом значения метрики F1-score за счет экстраполяции. Это указывает на то, что даже без применения метода чанкинга модели LLM с расширенным контекстным окном способны эффективно захватывать

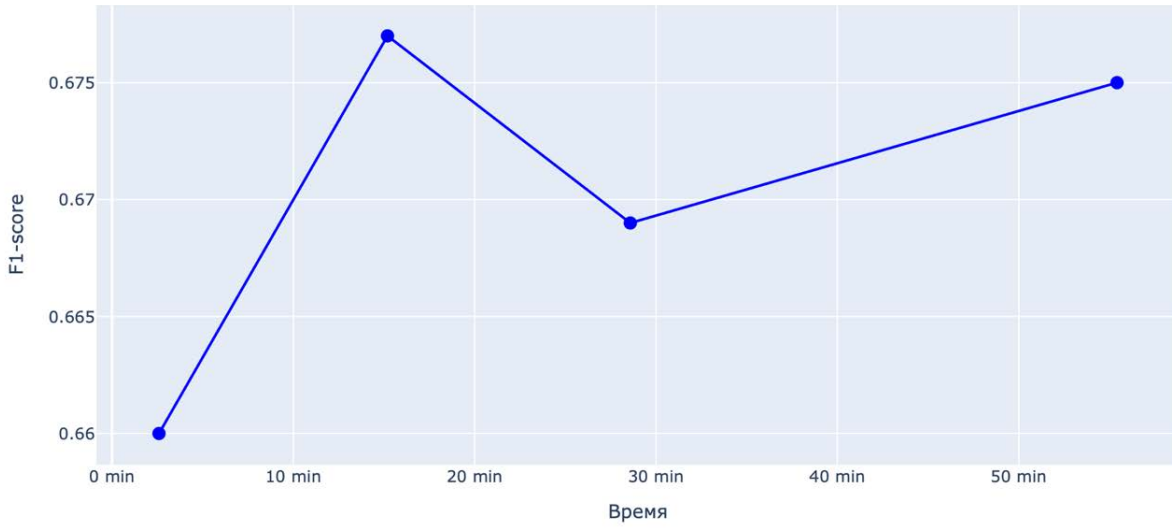

Рис. 1 / Fig. 1. **Зависимость метрик качества от времени обучения / The Relationship between Quality Metrics and Training Time**

Источник / Source: составлено авторами / Complied by the authors.

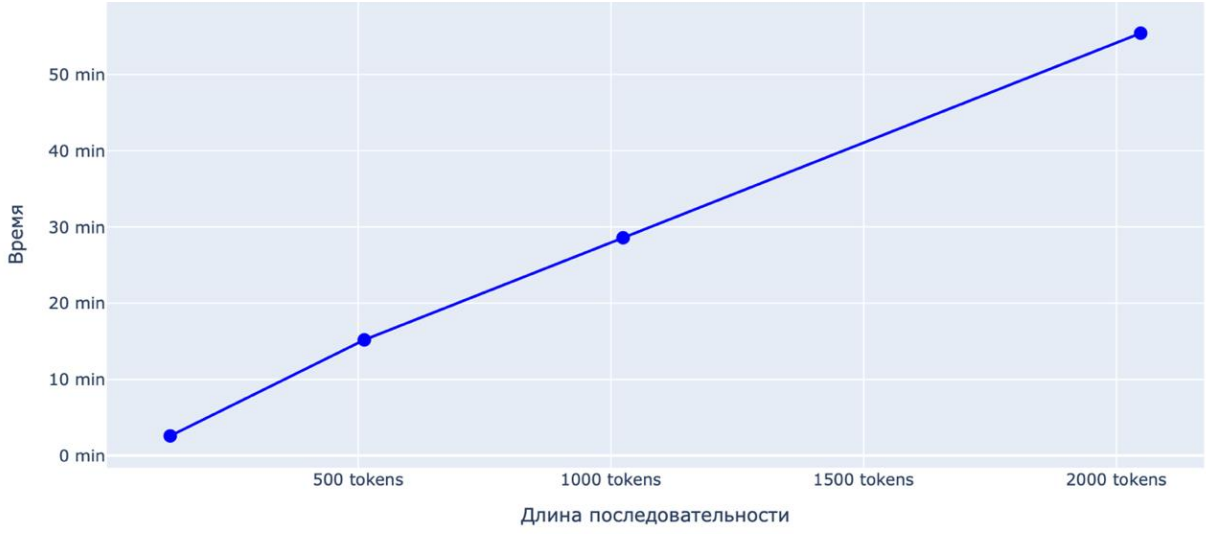

Рис. 2 / Fig. 2. **Зависимость времени обучения от входной длины последовательности / The Relationship between the Training Time and the Input Length of the Sequence**

Источник / Source: составлено авторами / Complied by the authors.

Таблица 3 / Table 3

Зависимость метрики качества F1-score и времени обучения модели RuBert-tiny2 от входной длины последовательности токенов / The Relationship between F1-score and the Training Time, and the Input Length of the Sequence for the RuBERT-tiny2 Model

Входная длина последовательности / Input length of the sequence	F1-score	Время обучения, мин / Training time, min
128	0,660	2,59
512	0,677	15,19
1024	0,669	28,58
2048	0,675	55,42

Источник / Source: составлено авторами / Complied by the authors.

Таблица 4 / Table 4

Данные экспериментов для моделей ruBERT-tiny1 и ruBERT-tiny2 при различной входной длине последовательности / Experimental Results for the RuBERT-tiny1 and RuBERT-tiny2 Models at Different Input Length of Sequence

Входная длина последовательности / Input length of the sequence	RuBERT-tiny1 (512)		RuBERT-tiny2 (2048)	
	F1-score	Время обучения, мин / Training time, min	F1-score	Время обучения, мин / Training time, min
128			0,6602	2,59
512	0,6486	13,48	0,6769	15,19
1024	0,6608	26,48	0,6698	28,58
2048	0,6681	50,53	0,6751	55,42

Источник / Source: составлено авторами / Complied by the authors.

семантический контекст текстов для задач классификации в NLP, извлекая релевантную информацию при меньших затратах ресурсов.

ФОРМУЛА РАСЧЕТА ИНДИКАТОРА ОЖИДАНИЙ НА БИРЖЕ

Для количественного анализа ожиданий участников финансового рынка в рамках настоящего исследования будет использован индикатор *MEI* (Market Estimations Indicator). Данный индикатор позволяет оценить прогнозные ожидания экономических агентов относительно ключевых макроэкономических переменных и, таким образом, выступает инструментом для исследования рыночной динамики и выявления потенциальных дисбалансов. Расчет *MEI* осуществляется на основе формулы, предложенной OECD, которая учитывает структурированные данные о прогнозах участников рынка и позволяет агрегировать индивидуальные оценки в единую интегральную величину. Применение данной методики обеспечивает сопоставимость результатов между различными временными периодами и экономическими условиями, а также повышает надежность интерпретации динамики ожиданий рынка в контексте экономического анализа.

Формально расчет *MEI* представлен следующей зависимостью:

$$MEI^{di} = 50 + \left(G + B * K_g\right) * \beta - \left(C + B * K_d\right) * \beta,$$

где MEI^{di} — диффузный индикатор мнений с распределенной инерцией; G — доля опрошенных респондентов, ответивших о повышении экономического параметра в будущем периоде; B — доля опрошенных респондентов, ответивших о неопределенности экономического параметра в будущем периоде; C — доля опрошенных респондентов, ответивших о понижении экономического параметра в будущем периоде; K_g — коэффициент веса факторов повышения экономического параметра в текущем периоде; K_d — коэффициент веса факторов понижения экономического параметра в текущем периоде; коэффициент $\beta = 0{,}5$.

Данная формула является примером того, как можно использовать получившиеся метки классификации для расчета индикатора будущих изменений на бирже.

ВЫВОДЫ

В настоящей работе представлены методы решения задач обработки естественного языка (NLP) в финансовой сфере с использованием предобученных трансформеров. Основная гипотеза исследования заключалась в том, что для эффективного обучения модели не всегда требуется использование длинного входного контекста. Применение мощных трансформеров совместно с уменьшенным контекстным окном, реализованным путем экстраполяции, позволяет оптимизировать объем необходимых вычислительных ресурсов при сохранении качества модели. В качестве практической рекомендации предлагается адаптировать количество входных токенов модели к медианному или среднему количеству токенов в текстах, на которых проводится обучение, с возможностью последующей корректировки без необходимости полного переобучения модели.

СПИСОК ИСТОЧНИКОВ

1. Липатова С.В., Бочкарева Ю.Е. Использование NLP для разработки электронных учебно-методических материалов. *Аллея науки*. 2023;4(79):926-931. URL: https://www.elibrary.ru/item.asp?id=54082726
2. Панкратова М.Д., Сковпель Т.Н. Модели NLP с использованием нейронных сетей в анализе тональности новостей. *Аналитические технологии в социальной сфере: Теория и Практика*. 2023;(15):97-107. URL: https://www.elibrary.ru/ctabku
3. Рыскин К.Э., Вечканова Ю.С., Федосин С.А. Обработка товарных номенклатур из отчетов дистрибьюторов с использованием NLP. Материалы XXV научно-практической конференции молодых ученых, аспирантов и студентов Национального исследовательского Мордовского государственного университета. Саранск: Национальный исследовательский Мордовский государственный университет им. Н.П. Огарёва; 2022:271–276; URL: https://elibrary.ru/item.asp?id=54051425
4. Дубровский В.В., Карманова Е.В. Проект разработки интеллектуального онлайн-сервиса для реферирования текстовых документов с использованием NLP. Управление проектами. Сборник статей по материалам II Всероссийской научной конференции, Магнитогорск, 01–03 декабря 2023 г. Магнитогорск: Магнитогорский государственный технический университет им. Г.И. Носова; 2024;37–45. URL: https://elibrary.ru/item.asp?id=60647866
5. Sennrich R., Haddow B., Birch A. Neural Machine Translation of Rare Words with Subword Units. *Proceedings of ACL*. 2016;1715–1725. DOI: 10.48550/arXiv.1508.07909
6. Song X., Salcianu A., Song Y., Dopson D., Zhou D. Fast WordPiece Tokenization. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2021:2089–2103. URL: https://aclanthology.org/2021.emnlp-main.160/
7. Vemula S.R., Sharma D.M., Krishnamurthy P. Rethinking Tokenization for Rich Morphology: The Dominance of Unigram over BPE and Morphological Alignment.*arXiv preprint*; 2025:arXiv:2508.08424. DOI: 10.48550/arXiv.2508.08424
8. Condevaux C., Harispe S. LSG Attention: Extrapolation of Pretrained Transformers to Long Sequences. In: Kashima, H., Ide, T., Peng, WC., eds. Advances in Knowledge Discovery and Data Mining. PAKDD 2023. Lecture Notes in Computer Science. 2023;13935:443–454. DOI: 10.1007/978-3-031-33374-3_35
9. Марков А.К., Семеночкин Д.О., Кравец А.Г., Яновский Т.А. Сравнительный анализ применяемых технологий обработки естественного языка для улучшения качества классификации цифровых документов. *International Jornal of Information Technologies*. 2024;12(3):66–77. URL: https://www.elibrary.ru/tubosi

REFERENCES

1. Lipatova S.V., Bochkareva Yu.E. Using NLP for the development of electronic teaching and methodological materials. *Alley of Science*. 2023;4(79):926-931. URL: https://www.elibrary.ru/item.asp?id=54082726
2. Pankratova M.D., Skovpel T.N. NLP models using neural networks in news sentiment analysis. *Analytical technologies in the social sphere: Theory and Practice*. 2023;(15):97-107. URL: https://www.elibrary.ru/ctabku
3. Ryskin K.E., Vechkanova Y.S., Fedosin S.A. Processing of product items from distributors' reports using NLP. Proceedings of the XXV Scientific and Practical Conference of Young Scientists, Postgraduate Students and Students of the National Research Mordovian State University. Saransk: National Research Mordovian State University named after N.P. Ogarev, 2022;271-276; URL: https://elibrary.ru/item.asp?id=54051425
4. Dubrovsky V.V., Karmanova E.V. Project for the Development of an Intelligent Online Service for Abstracting Text Documents Using NLP. Project Management. Proceedings of the II All-Russian Scientific Conference, Magnitogorsk, December 01–03, 2023. Magnitogorsk: Magnitogorsk State Technical University named after G.I. Nosov; 2024:37-45. URL: https://elibrary.ru/item.asp?id=60647866
5. Sennrich R., Haddow B., Birch A. Neural Machine Translation of Rare Words with Subword Units. *Proceedings of ACL*. 2016;1715-1725. DOI: 10.48550/arXiv.1508.07909
6. Song X., Salcianu A., Song Y., Dopson D., Zhou D. Fast WordPiece Tokenization. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2021;2089-2103. URL: https://aclanthology.org/2021.emnlp-main.160/
7. Vemula S.R., Sharma D.M., Krishnamurthy P. Rethinking Tokenization for Rich Morphology: The Dominance of Unigram over BPE and Morphological Alignment. 2025; URL: https://arxiv.org/abs/2508.08424
8. Condevaux C., Harispe S. LSG Attention: Extrapolation of Pretrained Transformers to Long Sequences. In: Kashima, H., Ide, T., Peng, WC., eds. Advances in Knowledge Discovery and Data Mining. PAKDD 2023. Lecture Notes in Computer Science. 2023;13935:443–454. DOI: 10.1007/978-3-031-33374-3_35

9. Markov A.K., Semenochkin D.O., Kravets A.G., Yanovsky T.A. Comparative analysis of applied natural language processing technologies to improve the quality of digital document classification. *International Journal of Information Technologies*. 2024;12(3):66-77. URL: https://www.elibrary.ru/tubosi

ИНФОРМАЦИЯ ОБ АВТОРАХ / ABOUT THE AUTHORS

Эльдар Филаридович Болтачев — кандидат технических наук, доцент кафедры искусственного интеллекта факультета информационных технологий и анализа больших данных, Финансовый университет при Правительстве Российской Федерации, Москва, Российская Федерация
Eldar F. Boltachev — Cand. Sci. (Tech.), Assoc. Prof. of Artificial Intelligence Department of the Faculty of Information Technology and Big Data Analysis, Financial University under the Government of the Russian Federation, Moscow, Russian Federation
https://orcid.org/0000-0002-6375-0365
Автор для корреспонденции / Corresponding author:
efboltachev@fa.ru

Александр Игоревич Тюляков — студент программы магистратуры кафедры искусственного интеллекта факультета информационных технологий и анализа больших данных, Финансовый университет при Правительстве Российской Федерации, Москва, Российская Федерация
Alexander I. Tyulyakov — Master Programme Student of Artificial Intellegence Department of the Faculty of Information Technologies and Big Data Analysis, Financial University under the Government of the Russian Federation, Moscow, Russian Federation
https://orcid.org/0009-0008-0534-0342
244447@edu.fa.ru

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.
Conflicts of Interest Statement: The authors have no conflicts of interest to declare.

Статья поступила в редакцию 13.10.2025; принята к публикации 24.11.2025.
Авторы прочитали и одобрили окончательный вариант рукописи.
The article was submitted on 13.10.2025; accepted for publication on 24.11.2025.
The authors read and approved the final version of the manuscript.