**Анализ тональности финансовых новостей на основе эмбеддингов и модели много классовой классификации**

Рыбалов Фёдор Андреевич

НИТУ МИСИС

Учебная группа: БИВТ-22 – ИСАД - 3

**Цели и задачи**

**Цель проекта**

Разработать систему анализа тональности новостей, способную оценивать информационный фон вокруг публичных компаний на фондовом рынке. Результатом должна быть автоматическая оценка позитивной или негативной направленности новостей.  
**Зачем нужна система**

Новости существенно влияют на восприятие компаний инвесторами, а значит — и на рыночную цену акций. Ручной анализ новостей требует времени и ресурсов, особенно при большом объёме информации. Автоматизированная система:

* позволяет быстро выявлять негативные или позитивные информационные всплески,
* предоставляет инвесторам инструмент для оценки рисков,
* может быть интегрирована в трейдинг-платформы как модуль новостной аналитики,
* подходит для корпоративного мониторинга репутационного фона.

В рамках проекта по анализу новостного фона фондового рынка были поставлены и решены следующие ключевые задачи:

1. Сбор новостей через парсинг RSS-лент

Для получения актуальной информации о событиях на фондовом рынке необходимо было настроить автоматический сбор новостей с популярных информационных ресурсов. Были выбраны следующие источники:

* **Finam.ru** — российский портал с фокусом на аналитику и новости рынка ценных бумаг,
* **Investing.com** — международный ресурс с широким охватом финансовых событий и мнений,
* **Ведомости** — деловая российская газета, публикующая материалы о компаниях и рынках.  
  Система парсинга построена на регулярных запросах к RSS-лентам этих источников. Каждая новость обрабатывается на уровне заголовка, содержимого и ссылки.

1. Анализ тональности новостей с использованием эмбеддингов и моделей машинного обучения

Для оценки тональности новостей были использованы предварительно обученные текстовые эмбеддинги модели deepvk/USER-bge-m3 из коллекции VK Sentence Encoders. После получения эмбеддингов их размерность была снижена с помощью метода Principal Component Analysis (PCA). На этом этапе была обучена классификационная модель (SVC), способная по векторному представлению определять, является ли заголовок позитивным, негативным или нейтральным.

3. Реализация Telegram-бота для взаимодействия с пользователем

Telegram-бот служит интерфейсом для конечного пользователя. Он реализует следующие функции:

* отправку пользователю новостей и результатов их анализа по команде /analyze или в автоматическом режиме,
* агрегацию общего новостного фона по компаниям за день или неделю (/today, /week, /last),
* скрытые ссылки на оригиналы статей и краткое содержание с результатом анализа тональности,
* фоновый режим обновления новостей каждые 5 минут с фильтрацией уже отправленных заголовков.

Вот развернутое описание модели, подготовленное с учётом кода, твоего описания и требований к проекту:

**Развёрнутое описание модели**

Модель анализа тональности новостей была построена с использованием эмбеддингов из открытой коллекции **VK Sentence Encoders**, в частности — модели deepvk/USER-bge-m3. Данный энкодер позволяет эффективно переводить текст заголовков новостных статей в числовое представление (вектор) размерностью **1×1024**, отражающее их смысловое содержание.

**Этапы построения модели:**

1. Сбор и предобработка данных:
   * Для обучения использован датасет от K.KHUBIEV  
     <https://www.kaggle.com/datasets/kkhubiev/russian-financial-news>   
     Датасет содержит свечные данные 176 компаний, торгуемых на московской бирже (MOEX), и 91,955 русскоязычных новостных статей на финансовую тематику, преимущественно про российский рынок ценных бумаг и российскую экономику.
   * Перед преобразованием в эмбеддинги заголовки очищались от HTML-разметки, ссылок и лишних символов.
   * Тексты без заголовков ("no title") отбрасывались.
2. Получение эмбеддингов:
   * Для каждого заголовка с помощью deepvk/USER-bge-m3 извлекались эмбеддинги (по первой скрытой репрезентации [CLS]-токена), затем нормализовались по L2-норме.
   * Обработка происходила батчами по 64 элемента, чтобы снизить нагрузку на видеопамять.
3. Снижение размерности:
   * Для уменьшения размерности пространства признаков применён метод главных компонент (PCA). Размерность была сокращена с 1024 до 100 признаков, что ускорило обучение модели и повысило её устойчивость к переобучению.
4. Преобразование целевой переменной:
   * Поскольку изначально задача стояла как регрессионная (предсказание численного значения тональности), но распределение таргета оказалось неравномерным, было принято решение перейти к многоклассовой классификации.
   * Оценки тональности были сгруппированы в три класса:
     + 0 — негативные новости (оценка < -0.1),
     + 1 — нейтральные (-0.1 ≤ оценка ≤ 0.1),
     + 2 — позитивные (> 0.1).
5. Обучение модели:
   * Были протестированы различные алгоритмы классификации, включая логистическую регрессию, случайный лес и градиентный бустинг.
   * Наилучшие результаты продемонстрировала модель SVC (Support Vector Classifier) с RBF-ядерной функцией.
   * Обучение производилось на тренировочном наборе с размером тестовой выборки 20%, стратифицированной по классам.
6. Оценка модели:
   * Отметим, что наилучшую точность и полноту модель показала в распознавании позитивных и негативных новостей. Нейтральные заголовки оказались наименее предсказуемыми, что согласуется с общей проблемой нечёткости их трактовки даже человеком.

|  |  |
| --- | --- |
| F1 (per class 0) | 0.716 |
| F1 (per class 1) | 0.541 |
| F1 (per class 2) | 0.773 |
| F1 (macro) | 0.677 |
| F1 (micro) | 0.693 |
| F1 (weighted) | 0.688 |

**Итоговая архитектура приложения**

Реализован телеграмм бот, который каждые 5 минут собирает свежие новости и отправляет пользователю в чат. Также реализован механизм сводки за текущий день и за неделю.  
  
Пайплайн обработки:

RSS Парсер

↓

Обработка текста (очистка, нормализация)

↓

Генерация эмбеддингов (модель Sentence Encoder от VK)

↓

Классификация (предсказание тональности — SVC)

↓

Telegram Bot (отправка новостей пользователю)

**Как работает:**

* При выполнении команды /analyze:
  + Запускается фоновое задание (job) с интервалом 5 минут.
  + Оно проверяет наличие новых новостей, которых ещё не было в CSV-файле news\_storage.csv.
  + Если такие есть — анализирует и отправляет их пользователю.
  + Добавляет отправленные новости в CSV, чтобы не присылать повторно

job\_queue.run\_repeating(

scheduled\_news\_analysis,

interval=300, # каждые 5 минут

first=10, # старт через 10 секунд

chat\_id=chat\_id,

name=str(chat\_id)

)

**Основные команды:**

1. **/start** — приветствие и вводная информация.
2. **/help** — список всех доступных команд.
3. **/analyze** — запуск анализа и включение автообновления каждые 5 минут.
4. **/today** — новости и общий тон за текущий день.
5. **/week** — новости и общий тон за последние 7 дней.
6. **/last** — последние 5 новостей

**Основные модули и логика**

* analyze\_news\_sentiment() — основной анализатор:
  + Загружает новости.
  + Предобрабатывает текст.
  + Получает эмбеддинги.
  + Делает предсказание (позитивная / нейтральная / негативная).
* format\_news(df) — форматирует DataFrame с новостями в удобный для Telegram вид (Markdown).
* sentiment\_summary(df) — считает количество новостей по каждой категории и формирует сводку

**Вывод**

В ходе проекта была создана автоматизированная система анализа тональности финансовых новостей с использованием эмбеддингов и машинного обучения. Реализован Telegram-бот, который каждые 5 минут отправляет пользователю актуальные новости с оценкой их тональности. Система показала хорошее качество классификации и может быть полезна для инвесторов и аналитиков при мониторинге информационного фона рынка.