# Классификация товаров по ОКПД 2 кодам

Сергей Андреевич Фирсов

Московский физико-технический институт

*Курс:* Автоматизация научных исследований (практика, В. В. Стрижов)/Группа Б05-105

*Эксперт:* В. М. Старожилец

*Консультант:* А. Е. Вознюк

2024

# Постановка задачи

Исследование направлено на решение задачи классификации товаров по кодам Общероссийского классификатора продукции по видам экономической деятельности (ОКПД 2) с использованием кратких текстовых описаний.
Выборка представлена парами "текстовое описание товара — код ОКПД2".

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m},\ \mathbf{x}_i = \{\text{token}\}_{j=1}^{n},\ y_i \in \mathbf{Y} = \{1, \ldots, k\}.$$

- Количество записей $m \approx 8$ миллионов, количество классов $k \approx 5000$.
- Структура классов несбалансирована: для некоторых классов доступно до 1000 записей, в то время как для других — более 200000.
- Текстовые описания часто содержат узкоспециализированную лексику, жаргонизмы, артикли и числовые значения, что усложняет задачу классификации.

# Определение модели

Используется модель логистической регрессии

$$P(y = 1|\mathbf{x}; \boldsymbol{\theta_k}) = \sigma(\mathbf{x}^\top \boldsymbol{\theta_k}),$$

где $\mathbf{x}$ обозначает вектор признаков наблюдения (с предварительно добавленной единицей для учета свободного члена), $\boldsymbol{\theta_k}$ — вектор параметров модели для класса $k$, а $\sigma(z) = \frac{1}{1+e^{-z}}$ — сигмоидная функция

Функция потерь и оптимизационная задача:

$$\mathcal{L}(\boldsymbol{\theta_k}) = -\frac{1}{N} \sum_{i=1}^{N} \left[ y_i \log(\sigma(\mathbf{x}_i^\top \boldsymbol{\theta_k})) + (1 - y_i) \log(1 - \sigma(\mathbf{x}_i^\top \boldsymbol{\theta_k})) \right],$$

$$\boldsymbol{\theta_k}^* = \arg\min_{\boldsymbol{\theta_k}} \mathcal{L}(\boldsymbol{\theta_k})$$

# Решение

Для анализа текстовых данных и классификации используется модель логистической регрессии, реализованная в библиотеке `scikit-learn`.
Для преобразования текстов в векторизованное представление используются эмбеддинги, полученные с помощью библиотеки `spaCy`.
Алгоритм решения:

1. Предварительная обработка данных: очистка текста от шума, нормализация и токенизация.
2. Фильтрация данных, для улучшения свойств выборки.
3. Преобразование текстов в векторное представление с использованием эмбеддингов `spaCy`.
4. Обучение модели логистической регрессии на обработанных данных.
5. Оценка качества модели с roc-auc и pr-auc.

# Вычислительный эксперимент

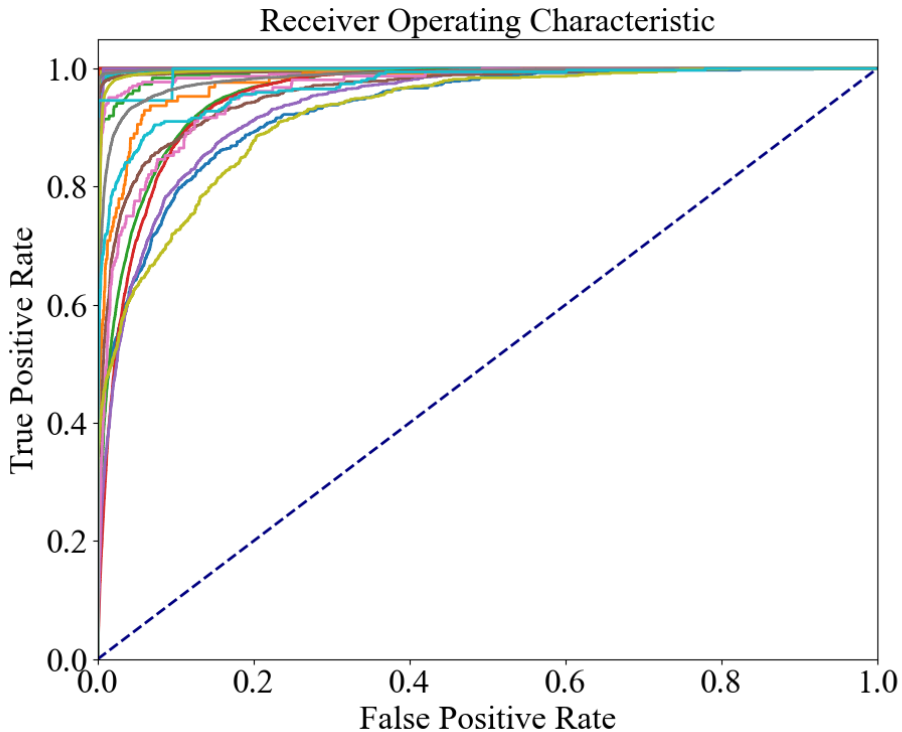
- Модель классификатора "sklearn.linear_model.LogisticRegression"
- Предобученная модель NLP "ru_core_news_lg"
- Проводились отдельные эксперименты с библиотекой "gensim", для получения другой вариации эмбеддингов. Но результаты классификации оказались значительно хуже.
- Размерность векторов эмбеддингов сильно влияет на классификацию. В экспериментах выбрана размерность 300, как оптимальная для использования "large" модели из "spaCy".
  Выбор был обоснован экспериментом со сравнением качества эмбеддингов полученных с помощью "large" и "small" моделей.

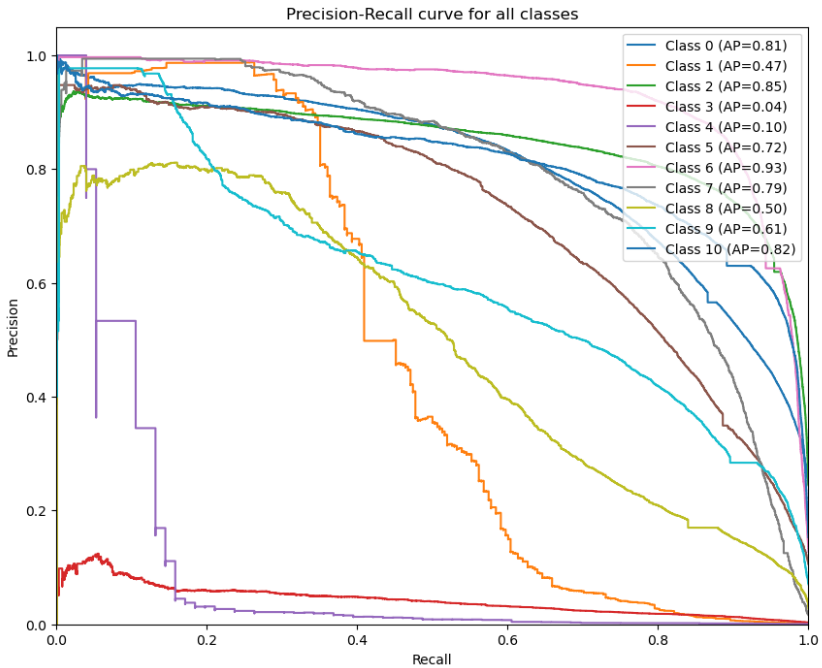
# Описание данных и их предобработка

Изначально данные — это пары значений: текстовое описание товара и его ОКПД2 код. Эти описания были составлены людьми, содержат орфографические ошибки, лишние символы, артикулы, цифры и много другое.

- Данные очистили и типизировали.
- В данных добавлены промежуточные ступени классификатора: полный код — это 9 цифр, пример 12.34.56.789. Разбиваем его на числовые значения по ступеням: 1 ступень — 12, ступень 1.5 — 123, ступень 2 — 1234, ступень 2.5 — 12345 и т.д..
- Из-за вычислительной сложности эксперимента было принято выделить несколько классов для анализа их классификации — 1,17,33,45,58,81,86 классы по первой ступени.
- Фильтрация для избавления от слишком мелких классов, чтобы избегать дизбаланса классов и улучшать качество классификации.

# Вычислительный эксперимент

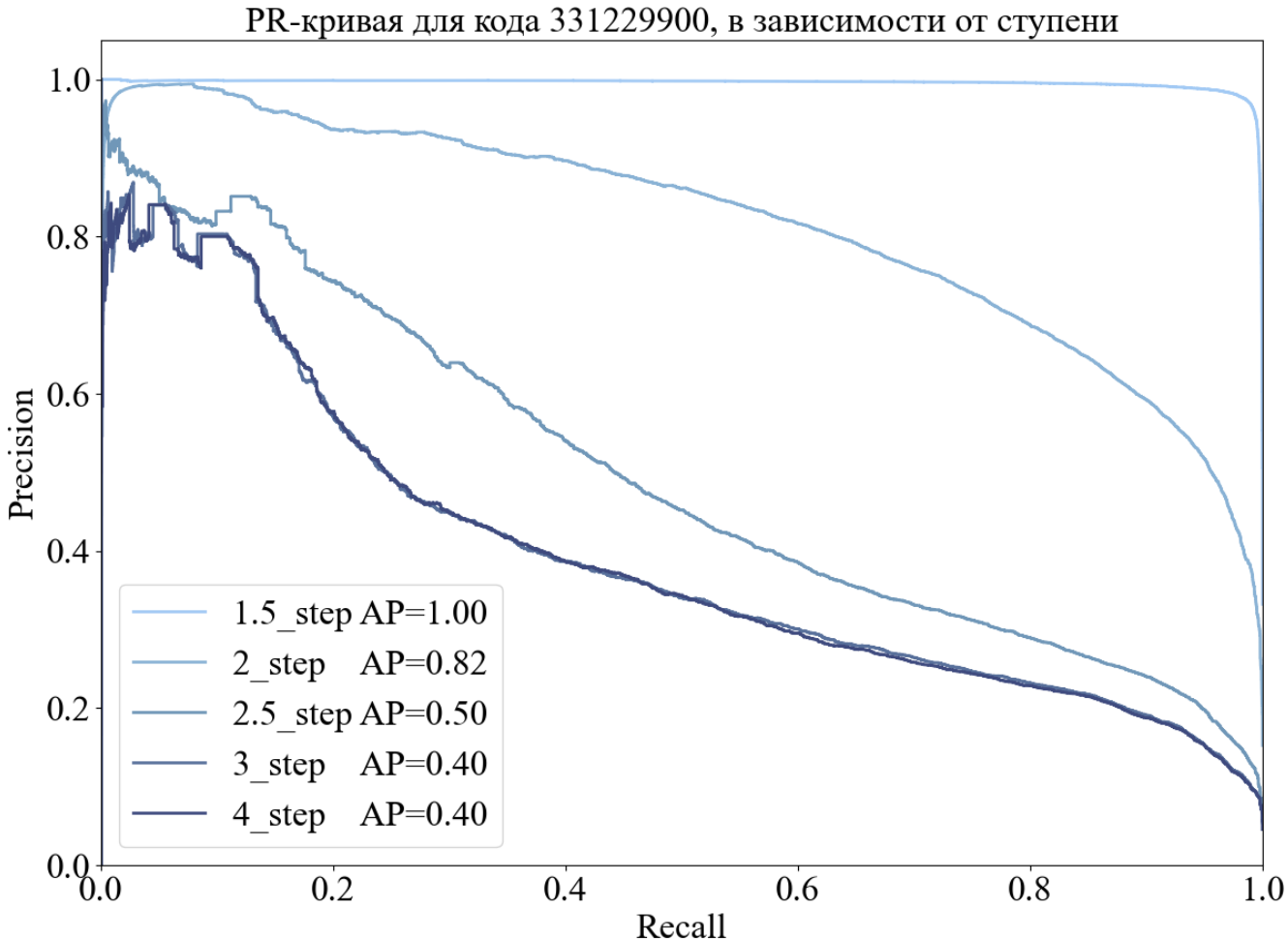



Здесь показаны ROC и PR кривые после классификации по ступени 1.5.

# Вычислительный эксперимент

| Text | 1 st | 1.5 st | 2 st | 2.5 st | 3 st | 4 st | Embedding |
|---|---|---|---|---|---|---|---|
| укроп свежий | 1 | 11 | 113 | 1131 | 11319 | 11319000 | [0.3,0.2..] |
| яблоки | 1 | 12 | 124 | 1241 | 12410 | 12410000 | [-0.2,0.3..] |



# Улучшенный эксперимент

Выборка одна и та же, всего 736 тысяч записей.
Так как всё же эксперименты чуть отличаются — не можем смотреть на графики roc и pr. Будем анализировать количество ошибок модели, в зависимости от ступени:

| **Method** | **Standart** | **Class split** | **Hierarchical** |
|---|---|---|---|
| step 2 | 90 | 86 | 87 |
| step 2.5 | 106 | 101 | 103 |
| step 3 | 120 | 112 | 113 |

В таблице представлено количество ошибок модели (в тысячах) в зависимости от метода.
**Вывод:** точность эксперимента повысилась лишь незначительно, примерно на 5%. При этом время и вычислительная сложность увеличились в разы.

# Заключение

- Предложен алгоритм для решения поставленной задачи классификации.
- Реализована модель выполняющая этот алгоритм
- Исследовано качестве модели в зависимости от глубины классификатора
- Исследовано влияние гиперпараметров на результаты модели
- Исследованы улучшенния модели и их качество

Пути улучшения:

- Улучшение качества эмбеддингов
- Объединение и усовершенствование предложенных улучшенных методов классификации
- Борьба с несбалансированностью классов

# Список литературы


- Lane, H., Howard, C., & Hapke, H. (2019). *Natural Language Processing in Action*. Manning Publications.
- Marra de Artiñano, I., Riottini Depetris, F., & Volpe Martincus, C. (2021). *Automatic Product Classification in International Trade: Machine Learning and Large Language Models*.
- Lewis, D. D., et al. (2004). *RCV1: A New Benchmark Collection for Text Categorization Research.* Journal of Machine Learning Research, 5.
- Haav, H.-M. (2021). *Assessment of HS Code Correctness*.
- Muñoz, E. (2020). *Introduction to Natural Language Processing: Word Embeddings & Sentiment Analysis with Python*.
- Montani, I. (2019). *Advanced NLP with spaCy: A Practical Guide to Advanced Natural Language Processing*. Independent.