Дипломная работа

Спектральный анализ текстов с использованием полиномов Лежандра

Студент: Имя Фамилия
Научный руководитель: д.ф.-м.н., профессор N. N.

Аннотация

В данной работе рассматривается подход к обработке и анализу текстовой информации, основанный на представлении текста в виде одномерного сигнала и последующем разложении этого сигнала по ортогональному базису полиномов Лежандра. Основная цель работы — показать, как можно извлекать интерпретируемые спектральные признаки, описывающие структуру текста по его длине, и использовать эти признаки в задачах классификации текстов и обучающих примерах для студентов.

Текст документа рассматривается как последовательность токенов (слов). Для каждого токена определяется числовая характеристика, отражающая интересующий признак: принадлежность к тематическому словарю (например, «политика», «спорт», «наука»), важность по TF–IDF, степень эмоциональной окраски и т. д. В результате возникает дискретный сигнал по позиции в тексте. Позиции нормируются в отрезок $[-1, 1]$, после чего полученный сигнал аппроксимируется рядом по полиномам Лежандра. Набор коэффициентов разложения $(a_0, a_1, a_2, \dots)$ служит компактным и интерпретируемым описанием распределения признака по длине документа: a_0 отвечает за средний уровень, a_1 — за линейный тренд (сдвиг активности к началу или концу текста), а a_2 — за наличие пиков в середине или по краям.

В работе подробно описан алгоритм построения спектральных признаков, приведены пошаговые разборы нескольких искусственных примеров, иллюстрирующих различные паттерны распределения тематических слов (только в начале, только в конце, пик в середине, несколько пиков по длине текста). Показано, как коэффициенты Лежандра позволяют различать такие структуры даже для коротких текстов. Кроме того, предлагается набор учебных заданий для студентов, позволяющих отработать на практике вычисление коэффициентов Лежандра, интерпретацию их значений и применение в задачах классификации.

Практическая часть выполнена на языке Python. В работе приведён полный исходный код на английском языке, реализующий вычисление полиномов Лежандра, построение текстовых сигналов, вычисление коэффициентов разложения, формирование гибридных признаков и использование их в простейшей задаче классификации текстов по тематике.

Содержание

1 Введение

Современные методы обработки текстовой информации (NLP) обычно опираются на представление текстов в виде наборов признаков. Наиболее распространённые подходы включают:

- мешок слов (bag-of-words) и TF–IDF;
- n-граммы;
- эмбеддинги слов и предложений.

Эти методы хорошо описывают, *какие слова* встречаются в документе и с какой частотой. Однако часто остаётся без внимания вопрос, *где именно* по длине текста располагаются важные элементы: ключевые слова, термины, эмоциональные фрагменты. Между тем структура текста по длине может играть важную роль:

- в новостях ключевой факт часто появляется в начале, детали — позже;
- в научных текстах введение и мотивация в начале, метод и результаты — ближе к середине и концу;
- в рецензиях и отзывах эмоциональный всплеск может располагаться в середине или в финальной части.

Цель данной работы — показать, как можно использовать полиномы Лежандра для извлечения *структурных* признаков текста, отражающих распределение выбранного признака (тематика, важность, эмоции) по длине документа. Идея состоит в том, чтобы:

1. рассмотреть текст как одномерный сигнал, заданный на последовательности позиций;
2. нормировать позиции в отрезок $[-1, 1]$;
3. разложить сигнал по базису полиномов Лежандра;
4. использовать полученные коэффициенты как компактное и интерпретируемое описание структуры текста.

В ходе работы:

- формально вводятся понятия текстового сигнала и его разложения по полиномам Лежандра;
- описывается алгоритм построения спектральных признаков;
- приводятся простые иллюстративные примеры и учебные задания;
- реализуется пример на Python, пригодный для использования студентами.

Все пояснения даны максимально подробно и «пошагово», чтобы студент бакалавриата или магистратуры мог самостоятельно воспроизвести вычисления и интерпретировать результаты.

2 Основные понятия

2.1 Текст, токены и тематический словарь

Под *текстом* мы будем понимать конечную последовательность токенов (слов или подслов) после предобработки (приведение к нижнему регистру, удаление знаков пунктуации и т. д.). Формально:

$$\text{Документ} = (t_1, t_2, \ldots, t_L),$$

где L — число токенов (длина текста), а каждый t_i — строка (слово).

Для тематического анализа вводится *тематический словарь* S, содержащий набор ключевых слов, соответствующих определённой теме. Например:

- тема «Политика»: $S_{\text{pol}} = \{\text{выборы},\ \text{министр},\ \text{парламент}\}$;
- тема «Спорт»: $S_{\text{sport}} = \{\text{матч},\ \text{гол},\ \text{тренер}\}$;
- тема «Кино»: $S_{\text{kino}} = \{\text{фильм},\ \text{актер},\ \text{премьера}\}$.

2.2 Текстовый сигнал

Определение 2.1. Пусть задан документ $(t_1, \ldots, t_L)$ и тематический словарь S. *Текстовым сигналом* по теме S называем последовательность чисел

$$f(i) = f_i, \quad i = 1, \ldots, L,$$

где f_i отражает наличие или степень проявления темы S на позиции i.

В простейшем случае можно использовать бинарный сигнал:

$$f_i = \begin{cases} 1, & \text{если } t_i \in S, \\ 0, & \text{иначе.} \end{cases}$$

Более сложные варианты:

- f_i равен TF–IDF-весу токена t_i;
- f_i равен оценке важности токена по модели внимания;
- f_i равен оценке эмоциональности данного фрагмента текста.

В дальнейшем для наглядности мы будем работать в основном с бинарным сигналом, так как его легко вычислять и интерпретировать.

2.3 Нормировка позиций в отрезок $[-1, 1]$

Чтобы применять полиномы Лежандра, аргумент функции должен меняться на отрезке $[-1, 1]$. Поэтому каждый номер позиции $i \in \{1, \ldots, L\}$ переводится в точку $x_i \in [-1, 1]$ по формуле:

$$x_i = -1 + 2 \cdot \frac{i-1}{L-1}, \qquad L \geq 2.$$

При $L = 1$ естественно положить $x_1 = 0$.

Таким образом:

$$x_1 = -1, \quad x_L = 1,$$

а промежуточные позиции равномерно распределены по отрезку $[-1, 1]$.

2.4 Полиномы Лежандра

Полиномы Лежандра $P_n(x)$, $n = 0, 1, 2, \ldots$, определяются как ортогональная система на отрезке $[-1, 1]$ с весом $w(x) = 1$. Первые полиномы имеют вид:

$$\begin{aligned} P_0(x) &= 1, \\ P_1(x) &= x, \\ P_2(x) &= \tfrac{1}{2}(3x^2 - 1), \\ P_3(x) &= \tfrac{1}{2}(5x^3 - 3x), \ \ldots \end{aligned}$$

Свойство ортогональности:

$$\int_{-1}^{1} P_n(x) P_m(x)\, dx = 0, \quad n \neq m.$$

Любую достаточно гладкую функцию $f(x)$ на $[-1, 1]$ можно разложить в ряд по полиномам Лежандра:

$$f(x) \approx \sum_{n=0}^{N-1} a_n P_n(x).$$

Коэффициенты ряда вычисляются по формуле:

$$a_n = \frac{2n+1}{2} \int_{-1}^{1} f(x) P_n(x)\, dx.$$

2.5 Аппроксимация интеграла суммой

В наших задачах функция f задана только в дискретных точках x_i. Поэтому интеграл заменим суммой:

$$\int_{-1}^{1} f(x) P_n(x)\, dx \approx \sum_{i=1}^{L} f_i P_n(x_i) w_i,$$

где w_i — веса квадратурной формулы. В простейшем варианте можно взять равномерную квадратуру:

$$w_i = \frac{2}{L}, \quad i = 1, \ldots, L.$$

Тогда приближённая формула для коэффициентов:

$$a_n \approx \frac{2n+1}{L} \sum_{i=1}^{L} f_i P_n(x_i).$$

Эту формулу мы будем использовать во всех последующих примерах.

3 Алгоритм построения спектральных признаков

В этом разделе формально сформулируем алгоритм, который превращает текстовый документ в набор коэффициентов Лежандра, описывающих структуру выбранного признака по длине текста.

3.1 Формулировка задачи

Пусть задан документ

$$(t_1, t_2, \ldots, t_L)$$

и интересующий нас признак (например, принадлежность токена тематическому словарю). Мы хотим построить набор чисел

$$(a_0, a_1, \ldots, a_{N-1}),$$

который:

- компактно описывает распределение признака по длине текста;
- легко интерпретируется;
- может использоваться как признак в задачах классификации текстов.

3.2 Пошаговый алгоритм

Шаг 1. Токенизация и предобработка текста.

1. Вход: сырой текст (строка).
2. Выполнить предобработку: привести к нижнему регистру, удалить пунктуацию, выполнить токенизацию.
3. Получить последовательность токенов $(t_1, \ldots, t_L)$.

Шаг 2. Построение текстового сигнала.

1. Задать тематический словарь S.
2. Для каждой позиции $i = 1, \ldots, L$ определить:

$$f_i = \begin{cases} 1, & \text{если } t_i \in S, \\ 0, & \text{иначе.} \end{cases}$$

3. Получить текстовый сигнал:

$$f = (f_1, f_2, \ldots, f_L).$$

Шаг 3. Нормировка позиций.

1. Для каждого $i = 1, \ldots, L$ вычислить

$$x_i = -1 + 2 \cdot \frac{i-1}{L-1},$$

если $L > 1$, и $x_1 = 0$, если $L = 1$.

2. Получить массив точек

$$x = (x_1, \ldots, x_L).$$

Шаг 4. Вычисление полиномов Лежандра.

1. Выбрать порядок разложения N (например, $N = 3$ или $N = 5$).
2. Вычислить значения полиномов Лежандра $P_n(x_i)$ для всех $i = 1, \ldots, L$ и $n = 0, \ldots, N-1$, используя рекуррентную формулу:

$$P_0(x) = 1, \quad P_1(x) = x, \quad (n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x).$$

Шаг 5. Вычисление коэффициентов. Для каждого $n = 0, \ldots, N-1$ вычислить:

$$a_n = \frac{2n+1}{L} \sum_{i=1}^{L} f_i P_n(x_i).$$

Шаг 6. Интерпретация и использование.

- a_0 отражает средний уровень признака (долю или среднюю величину) по тексту;
- знак и величина a_1 показывают, смещён ли признак к началу или к концу документа;
- знак и величина a_2 отражают, где сосредоточен основной пик: в центре или по краям;
- набор $(a_0, \ldots, a_{N-1})$ может использоваться как компонент вектора признаков для классифицирующей модели.

4 Иллюстративные примеры

В этом разделе приводятся подробные примеры для коротких текстов. Их можно использовать как учебные задачи для ручных расчётов.

4.1 Пример 1: ключевые слова в начале, середине и конце

Рассмотрим текст:

Текст 1: «Банк повысил процентные ставки по кредитам».

После токенизации:

$$t_1 = \text{банк},\ t_2 = \text{повысил},\ t_3 = \text{процентные},\ t_4 = \text{ставки},\ t_5 = \text{по},\ t_6 = \text{кредитам},$$

так что $L = 6$.

Выберем словарь:

$$S = \{\text{банк},\ \text{ставки},\ \text{кредитам}\}.$$

Шаг 1. Сигнал.

$$f = (1, 0, 0, 1, 0, 1),$$

так как слова на позициях 1, 4 и 6 входят в словарь.

Шаг 2. Нормировка позиций.

$$x_i = -1 + 2 \cdot \frac{i-1}{5}, \quad i = 1, \ldots, 6,$$

получаем:

$$x = (-1,\ -0.6,\ -0.2,\ 0.2,\ 0.6,\ 1).$$

Шаг 3. Полиномы Лежандра.

$$\begin{aligned} P_0(x_i) &= 1, \\ P_1(x_i) &= x_i, \\ P_2(x_i) &= \tfrac{1}{2}(3x_i^2 - 1). \end{aligned}$$

Подставляя значения x_i, получаем приближённо:

$$\begin{aligned} P_0 &= (1,\ 1,\ 1,\ 1,\ 1,\ 1), \\ P_1 &= (-1,\ -0.6,\ -0.2,\ 0.2,\ 0.6,\ 1), \\ P_2 &\approx (1,\ 0.04,\ -0.44,\ -0.44,\ 0.04,\ 1). \end{aligned}$$

Шаг 4. Коэффициенты. Используем формулу

$$a_n = \frac{2n+1}{L} \sum_{i=1}^{L} f_i P_n(x_i).$$

Коэффициент a_0.

$$a_0 = \frac{1}{6} \sum_{i=1}^{6} f_i = \frac{3}{6} = 0.5.$$

Коэффициент a_1.

$$a_1 = \frac{3}{6} \sum f_i x_i = \frac{1}{2}[(1)(-1) + (1)(0.2) + (1)(1)] = \frac{1}{2} \cdot 0.2 = 0.1.$$

Коэффициент a_2.

$$a_2 = \frac{5}{6} \sum f_i P_2(x_i) \approx \frac{5}{6}[1 + (-0.44) + 1] = \frac{5}{6} \cdot 1.56 \approx 1.3.$$

Интерпретация.

- $a_0 = 0.5$ показывает, что примерно половина позиций в тексте относится к теме.
- $a_1 > 0$, но небольшой: активность немного смещена к концу.
- a_2 положительный и довольно большой: это соответствует сильным пикам по краям текста и более слабой середине.

4.2 Пример 2: ключевые слова только в начале

Рассмотрим сигнал:

$$f^{(\text{нач})} = (1, 1, 1, 0, 0, 0).$$

Позиции x_i и значения полиномов $P_n(x_i)$ такие же, как в примере 1.

Коэффициент $a_0^{(\text{нач})}$.

$$a_0^{(\text{нач})} = \frac{1}{6}(1 + 1 + 1) = 0.5.$$

Коэффициент $a_1^{(\text{нач})}$.

$$a_1^{(\text{нач})} = \frac{1}{2}(-1 - 0.6 - 0.2) = \frac{1}{2} \cdot (-1.8) = -0.9.$$

Коэффициент $a_2^{(\text{нач})}$.

$$a_2^{(\text{нач})} = \frac{5}{6}[P_2(x_1) + P_2(x_2) + P_2(x_3)] \approx \frac{5}{6}(1 + 0.04 - 0.44) = \frac{5}{6} \cdot 0.6 \approx 0.5.$$

Интерпретация.

- $a_1^{(\text{нач})} < 0$: активность сильно смещена к началу текста.
- $a_2^{(\text{нач})} > 0$: структура без центрального пика, скорее «свалена» к одному краю.

4.3 Пример 3: ключевые слова только в конце

Сигнал:

$$f^{(\text{кон})} = (0, 0, 0, 1, 1, 1).$$

Повторяя расчёты, получаем:

$$a_0^{(\text{кон})} = 0.5, \quad a_1^{(\text{кон})} = 0.9, \quad a_2^{(\text{кон})} \approx 0.5.$$

Здесь $a_1 > 0$ большого модуля показывает смещение активности к концу текста. Значения a_0 и a_2 совпадают с примером 2, что отражает симметрию ситуации.

4.4 Пример 4: единый пик в середине

Рассмотрим сигнал:

$$f^{(\text{сер})} = (0, 0, 1, 1, 0, 0).$$

Тогда:

$$a_0^{(\text{сер})} = \frac{1}{6}(1 + 1) = \frac{2}{6} \approx 0.33,$$

$$a_1^{(\text{сер})} = \frac{1}{2}(x_3 + x_4) = \frac{1}{2}(-0.2 + 0.2) = 0,$$

$$a_2^{(\text{сер})} \approx \frac{5}{6}[-0.44 - 0.44] = \frac{5}{6}(-0.88) \approx -0.73.$$

Отрицательный a_2 указывает на то, что активность сосредоточена в центре, а по краям она меньше.

4.5 Сводное сравнение

Паттерн	a_0	a_1	a_2
Начало–середина–конец (пример 1)	0.50	0.10	1.30
Только начало (пример 2)	0.50	-0.90	0.50
Только конец (пример 3)	0.50	0.90	0.50
Центральный пик (пример 4)	0.33	0.00	-0.73

Из таблицы видно:

- a_0 описывает общий уровень активности;
- a_1 показывает смещение к началу ($a_1 < 0$) или к концу ($a_1 > 0$);
- $a_2 > 0$ соответствует усиленным краям, а $a_2 < 0$ — центральному пику.

5 Учебные задания для студентов

В этом разделе собраны задания, которые можно использовать в рамках практических занятий.

5.1 Базовые задания

1. Сигнал темы вдоль текста.

Дан текст ($L = 6$ токенов):

$$t = (\text{фильм},\ \text{новый},\ \text{актер},\ \text{интервью},\ \text{кино},\ \text{обзор}).$$

Словарь темы «Кино» = {фильм, актер, кино, обзор}.

Задание:

(a) Построить бинарный сигнал $f(i)$.

(b) Перевести позиции в $x_i \in [-1, 1]$.

(c) Найти a_0, a_1, a_2.

(d) Объяснить смысл знаков и величин коэффициентов.

2. Смещение ключевых слов к началу.

Дан текст ($L = 5$):

$$t = (\text{министр},\ \text{выборы},\ \text{банк},\ \text{новости},\ \text{обзор}).$$

Словарь темы «Политика» = {министр, выборы, банк}.

Задание: вычислить коэффициент a_1 и определить, смещён ли пик по теме к началу или к концу документа.

3. Большой всплеск в середине.

Дан текст ($L = 7$):

$t =$ (введение, пример, лаборатория, исследование, наука, обсуждение, вывод).

Словарь «Наука» = {лаборатория, исследование, наука}.

Задание:

(a) Построить сигнал $f(i)$.

(b) Вычислить a_2.

(c) Объяснить, почему при центральном пике часто $a_2 < 0$.

4. Сравнение двух текстов одинаковой длины.

Даны два текста ($L = 6$):

A) «кино новый фильм премьера актер обзор» B) «экономика рынок банк ставки кризис обзор»

Словарь «Кино» = {кино, фильм, премьера, актер}. Словарь «Экономика» = {банк, ставки, кризис}.

Задание:

(a) Для каждого текста построить сигналы по соответствующим словарям.

(b) Вычислить коэффициенты a_1.

(c) На основе знака и величины a_1 сравнить, где тема появляется раньше по длине текста.

5. Ранжирование по структуре признака.

Рассматривается сигнал:

$$f = (0, 0, 1, 0, 0), \quad L = 5.$$

Задание:

(a) Вычислить коэффициенты a_0 и a_1.

(b) Объяснить, почему одного a_0 недостаточно, чтобы понять, где именно по тексту находится активность.

6 Реализация алгоритма на Python

В этом разделе приводится реализация основных частей алгоритма на языке Python. Все комментарии и имена функций даны на английском языке, чтобы код было удобно использовать в международной среде.

6.1 Вычисление полиномов Лежандра и нормировка позиций

```python
import numpy as np

def normalized_positions(L: int) -> np.ndarray:
    """
    Map positions 0..L-1 to x in [-1, 1].

    Parameters
    ----------
    L : int
        Length of the text (number of tokens).

    Returns
    -------
    x : np.ndarray
        Array of shape (L,) with normalized positions in [-1, 1].
    """
    if L <= 1:
        return np.array([0.0], dtype=float)
    i = np.arange(L, dtype=float)
    x = -1.0 + 2.0 * i / (L - 1)
    return x


def legendre_polynomials(x: np.ndarray, N: int) -> np.ndarray:
    """
```

```
    Compute Legendre polynomials P_0,...,P_{N-1} at given points x.

    Parameters
    ----------
    x : np.ndarray
        1D array of shape (L,) with points in [-1, 1].
    N : int
        Number of polynomials to compute.

    Returns
    -------
    P : np.ndarray
        2D array of shape (L, N) where P[i, n] = P_n(x[i]).
    """
    L = len(x)
    P = np.zeros((L, N), dtype=float)

    if N <= 0:
        return P

    # P0(x) = 1
    P[:, 0] = 1.0

    if N == 1:
        return P

    # P1(x) = x
    P[:, 1] = x

    # Recurrence for higher orders:
    # (n+1) P_{n+1}(x) = (2n+1) x P_n(x) - n P_{n-1}(x)
    for n in range(1, N - 1):
        P[:, n + 1] = ((2 * n + 1) * x * P[:, n] - n * P[:, n - 1])
            / (n + 1)

    return P
```

6.2 Построение текстового сигнала и коэффициентов Лежандра

```
def build_binary_signal(tokens, keyword_set) -> np.ndarray:
    """
    Build a binary signal for a given topic (keyword set).

    Parameters
    ----------
    tokens : list of str
        Tokenized text.
    keyword_set : set of str
        Set of keywords representing a topic.
```

```
    Returns
    -------
    signal : np.ndarray
        1D array of length L with values 0 or 1.
    """
    L = len(tokens)
    signal = np.zeros(L, dtype=float)
    for i, tok in enumerate(tokens):
        if tok in keyword_set:
            signal[i] = 1.0
    return signal


def legendre_coeffs_for_signal(signal: np.ndarray, N: int) -> np.ndarray:
    """
    Compute approximate Legendre coefficients a_0,...,a_{N-1}
    for a discrete signal defined on positions mapped to [-1, 1].

    Parameters
    ----------
    signal : np.ndarray
        1D array of shape (L,) with values of the text signal.
    N : int
        Number of coefficients.

    Returns
    -------
    coeffs : np.ndarray
        1D array of shape (N,) with Legendre coefficients.
    """
    L = len(signal)
    x = normalized_positions(L)
    P = legendre_polynomials(x, N)
    coeffs = np.zeros(N, dtype=float)

    # Approximate integral using simple rectangular quadrature:
    # a_n ~ (2n+1)/L * sum_{i} f_i P_n(x_i)
    for n in range(N):
        coeffs[n] = (2 * n + 1) / L * np.sum(signal * P[:, n])

    return coeffs
```

6.3 Формирование гибридных признаков

Здесь мы предполагаем, что у нас уже есть TF–IDF-признаки (или любые другие векторные признаки) и тематический словарь.

```
def hybrid_features(tokens, keyword_sets, base_vector, N_legendre=3):
```

```
    """
    Build a hybrid feature vector that combines base text features
    (e.g., TF-IDF) with Legendre coefficients for multiple topics.

    Parameters
    ----------
    tokens : list of str
        Tokenized text.
    keyword_sets : dict
        Dictionary mapping topic name to a set of keywords.
        Example: {"politics": {...}, "sport": {...}}.
    base_vector : np.ndarray
        Base feature vector (e.g., TF-IDF) of shape (D,).
    N_legendre : int
        Number of Legendre coefficients per topic.

    Returns
    -------
    features : np.ndarray
        Extended feature vector with concatenated Legendre
           coefficients.
    """
    legendre_parts = []

    for topic, kw_set in keyword_sets.items():
        signal = build_binary_signal(tokens, kw_set)
        coeffs = legendre_coeffs_for_signal(signal, N_legendre)
        legendre_parts.append(coeffs)

    if len(legendre_parts) > 0:
        legendre_concat = np.concatenate(legendre_parts)
        return np.concatenate([base_vector, legendre_concat])
    else:
        return base_vector
```

6.4 Пример использования в простой классификации

В следующем примере мы показываем, как построить матрицу признаков и обучить логистическую регрессию на небольшом наборе документов. Для краткости TF–IDF-вектора заменяются на фиктивные числовые векторы.

```
from sklearn.linear_model import LogisticRegression
import numpy as np

def example_classification():
    """
    Simple example of using hybrid features with Legendre
       coefficients
    for a toy text classification problem.
    """
    # Toy documents (tokenized)
```

```
    docs_tokens = [
        ["elections", "will", "be", "held", "in", "parliament"],
        ["team", "scored", "a", "goal", "in", "the", "match"],
        ["university", "opens", "a", "new", "program"],
        ["graphene", "research", "made", "a", "breakthrough"]
    ]

    # Topics (keyword sets)
    keyword_sets = {
        "politics": {"elections", "parliament", "minister"},
        "sport": {"team", "goal", "match"},
        "education": {"university", "program", "education"},
        "science": {"research", "graphene", "science"}
    }

    # Toy base vectors: for a real task, use TF-IDF or embeddings
    base_vectors = [
        np.array([1.0, 0.0, 0.0, 0.0]),  # politics document
        np.array([0.0, 1.0, 0.0, 0.0]),  # sport document
        np.array([0.0, 0.0, 1.0, 0.0]),  # education document
        np.array([0.0, 0.0, 0.0, 1.0])   # science document
    ]

    # Labels: 0=politics, 1=sport, 2=education, 3=science
    labels = np.array([0, 1, 2, 3], dtype=int)

    X = []
    for tokens, base_vec in zip(docs_tokens, base_vectors):
        feats = hybrid_features(tokens, keyword_sets, base_vec,
           N_legendre=3)
        X.append(feats)

    X = np.vstack(X)

    clf = LogisticRegression(max_iter=1000)
    clf.fit(X, labels)

    # Test on a new document
    new_tokens = ["new", "match", "of", "the", "team"]
    new_base_vec = np.array([0.0, 1.0, 0.0, 0.0])  # sport-like
       base vector
    new_feats = hybrid_features(new_tokens, keyword_sets,
       new_base_vec, N_legendre=3)
    pred = clf.predict(new_feats.reshape(1, -1))[0]
    print("Predicted class:", pred)
```

Этот пример иллюстрирует идею: к любым базовым признакам (TF–IDF, эмбеддинги) можно добавить набор коэффициентов Лежандра для нескольких тематических словарей и использовать полученный вектор как вход для стандартного классификатора.

7 Заключение

В работе рассмотрен подход к анализу текстов, основанный на разложении по полиномам Лежандра. Основные идеи и результаты можно кратко сформулировать следующим образом:

- Текст можно рассматривать как одномерный сигнал по длине документа, где каждой позиции соответствует числовое значение, отражающее наличие или степень проявления интересующего признака (темы, важности, эмоциональности).
- Нормируя позиции токенов в отрезок $[-1, 1]$, можно применить аппарат полиномов Лежандра и получить компактное разложение сигнала.
- Коэффициенты разложения $a_0, a_1, a_2, \ldots$ имеют естественную интерпретацию: a_0 описывает средний уровень признака, a_1 отвечает за смещение активности к началу или концу текста, a_2 позволяет различать центральный пик и усиленные края.
- На простых примерах показано, как различные распределения ключевых слов (только в начале, только в конце, пики в середине и по краям) отражаются в наборе коэффициентов.
- Предложен набор учебных задач, которые позволяют студентам самостоятельно отработать вычисление коэффициентов Лежандра и их интерпретацию в контексте анализа текстов.
- Реализация на Python демонстрирует, как можно объединить базовые текстовые признаки (например, TF–IDF) и спектральные признаки, основанные на полиномах Лежандра, в единый гибридный вектор и использовать его для простой задачи классификации текстов.

Предложенный подход не претендует на замену современных глубоких моделей обработки текстов, но может служить:

- наглядным учебным инструментом,
- способом построения интерпретируемых признаков,
- дополнительным модулем в гибридных системах анализа текстов.

В дальнейшем подход можно развивать, используя:

- более сложные сигналы (на основе эмбеддингов, оценок внимания, синтаксических ролей);
- многоканальные разложения, где каждый канал соответствует своей теме;
- сравнение Лежандра-разложения с разложениями по другим ортогональным системам (например, системой Уолша или Фурье).

Список литературы (пример)

1. Рудин У. *Функциональный анализ.* М.: Мир, 1975.
2. Фихтенгольц Г. М. *Курс дифференциального и интегрального исчисления.* Т. 3. М.: Наука, 1966.
3. Jurafsky D., Martin J. H. *Speech and Language Processing.* 3rd ed., draft, 2023.
4. Manning C. D., Raghavan P., Schütze H. *Introduction to Information Retrieval.* Cambridge University Press, 2008.

8 Иллюстративные примеры работы спектрального анализа текста на основе полиномов Лежандра

В этом разделе приведены подробные пошаговые примеры применения описанного выше алгоритма к коротким текстам. Цель примеров — показать, как коэффициенты Лежандра (a_0, a_1, a_2) отражают разные паттерны расположения ключевых слов по длине документа:

- ключевые слова встречаются в начале, середине и в конце текста;
- ключевые слова сосредоточены только в начале;
- ключевые слова сосредоточены только в конце;
- один крупный пик активности в середине текста.

Во всех примерах мы будем использовать:

- длину текста $L = 6$ (шесть токенов);
- первые три полинома Лежандра:

$$P_0(x) = 1, \qquad P_1(x) = x, \qquad P_2(x) = \frac{1}{2}(3x^2 - 1);$$

- одинаковую нормировку позиций x_i в отрезок $[-1, 1]$;
- ту же формулу для коэффициентов:

$$a_n \approx \frac{2n+1}{L} \sum_{i=1}^{L} f_i P_n(x_i), \qquad n = 0, 1, 2.$$

8.1 Пример 1: текст с ключевыми словами в начале, середине и конце

Рассмотрим текст:

Текст 1: «Банк повысил процентные ставки по кредитам».

Пусть после предобработки и токенизации получаем последовательность:

$t_1 =$ банк, $t_2 =$ повысил, $t_3 =$ процентные, $t_4 =$ ставки, $t_5 =$ по, $t_6 =$ кредитам,

так что $L = 6$.

Шаг 1. Выбор ключевых слов и построение сигнала. Выберем словарь финансовых ключевых слов

$$S = \{\text{банк},\ \text{ставки},\ \text{кредитам}\}.$$

Строим бинарный индикаторный сигнал:

$$f(i) = \begin{cases} 1, & \text{если } t_i \in S, \\ 0, & \text{иначе.} \end{cases}$$

По нашему тексту получаем:

$$f = (f_1, \ldots, f_6) = (1, 0, 0, 1, 0, 1),$$

так как ключевые слова стоят на местах 1 («банк»), 4 («ставки») и 6 («кредитам»).

Шаг 2. Нормировка позиции в $[-1, 1]$. Нормируем позиции $i = 1, \ldots, 6$ по формуле

$$x_i = -1 + 2 \cdot \frac{i-1}{L-1}, \qquad L = 6.$$

Получаем:

$$x_1 = -1, \quad x_2 = -0.6, \quad x_3 = -0.2, \quad x_4 = 0.2, \quad x_5 = 0.6, \quad x_6 = 1.$$

Шаг 3. Значения полиномов Лежандра в точках x_i. Для P_0 и P_1 всё просто:

$$P_0(x_i) = 1, \quad P_1(x_i) = x_i.$$

Для $P_2(x) = \frac{1}{2}(3x^2 - 1)$ считаем по точкам (значения округлены до двух знаков):

$$P_2(x_1) = 1, \quad P_2(x_2) \approx 0.04, \quad P_2(x_3) \approx -0.44, \quad P_2(x_4) \approx -0.44, \quad P_2(x_5) \approx 0.04, \quad P_2(x_6) = 1.$$

Итак,

$$\begin{aligned} P_0 &= (1,\ 1,\ 1,\ 1,\ 1,\ 1), \\ P_1 &= (-1,\ -0.6,\ -0.2,\ 0.2,\ 0.6,\ 1), \\ P_2 &\approx (1,\ 0.04,\ -0.44,\ -0.44,\ 0.04,\ 1). \end{aligned}$$

Шаг 4. Вычисление коэффициентов a_0, a_1, a_2. Используем формулу

$$a_n \approx \frac{2n+1}{L} \sum_{i=1}^{L} f_i P_n(x_i).$$

Коэффициент a_0 (средний уровень).

$$a_0 = \frac{1}{6} \sum_{i=1}^{6} f_i P_0(x_i) = \frac{1}{6} \sum_{i=1}^{6} f_i = \frac{1}{6}(1+0+0+1+0+1) = \frac{3}{6} = 0.5.$$

Интерпретация: примерно половина позиций в тексте помечена как «финансовые».

Коэффициент a_1 (линейный тренд).

$$a_1 = \frac{3}{6}\sum_{i=1}^{6} f_i P_1(x_i) = \frac{1}{2}\sum_{i=1}^{6} f_i x_i.$$

Ненулевые f_i только при $i = 1, 4, 6$, поэтому

$$\sum f_i x_i = 1 \cdot (-1) + 1 \cdot 0.2 + 1 \cdot 1 = -1 + 0.2 + 1 = 0.2.$$

Отсюда

$$a_1 = \frac{1}{2} \cdot 0.2 = 0.1.$$

Интерпретация: лёгкий положительный тренд — чуть больше активности ближе к концу текста.

Коэффициент a_2 (форма, середина/края).

$$a_2 = \frac{5}{6}\sum_{i=1}^{6} f_i P_2(x_i).$$

Сумма по активным индексам:

$$\sum f_i P_2(x_i) = P_2(x_1) + P_2(x_4) + P_2(x_6) \approx 1 + (-0.44) + 1 = 1.56.$$

Тогда

$$a_2 \approx \frac{5}{6} \cdot 1.56 \approx 1.3.$$

Интерпретация: заметно положительный a_2 соответствует структуре с пиками в начале и в конце и спадом в середине — что согласуется с тем, что ключевые слова стоят в начале, середине и конце, причём по краям их особенно много.

8.2 Пример 2: ключевые слова только в начале текста

Теперь рассмотрим искусственный пример, где ключевые слова сосредоточены только в начале текста.

Шаг 1. Сигнал. Пусть снова $L = 6$ и для простоты используем ту же сетку x_i. Рассмотрим сигнал

$$f^{(\text{нач})} = (1, 1, 1, 0, 0, 0).$$

Это означает, что первые три токена — ключевые, а остальные — нет.

Шаг 2. Коэффициенты Лежандра. Поскольку x_i и $P_n(x_i)$ такие же, как в примере 1, переходим сразу к суммам.

Коэффициент $a_0^{(\text{нач})}$.

$$a_0^{(\text{нач})} = \frac{1}{6}\sum_{i=1}^{6} f_i^{(\text{нач})} = \frac{1}{6}(1 + 1 + 1 + 0 + 0 + 0) = \frac{3}{6} = 0.5.$$

Как и в первом примере, половина позиций активна.

Коэффициент $a_1^{(\text{нач})}$.

$$a_1^{(\text{нач})} = \frac{3}{6}\sum_{i=1}^{6} f_i^{(\text{нач})} x_i = \frac{1}{2}\big(x_1 + x_2 + x_3\big).$$

Подставляя $x_1 = -1, x_2 = -0.6, x_3 = -0.2$, получаем

$$x_1 + x_2 + x_3 = -1 - 0.6 - 0.2 = -1.8,$$

поэтому

$$a_1^{(\text{нач})} = \frac{1}{2} \cdot (-1.8) = -0.9.$$

Интерпретация: сильный отрицательный тренд — почти вся активность сосредоточена в начале текста.

Коэффициент $a_2^{(\text{нач})}$.

$$a_2^{(\text{нач})} = \frac{5}{6}\sum_{i=1}^{6} f_i^{(\text{нач})} P_2(x_i) = \frac{5}{6}\big(P_2(x_1) + P_2(x_2) + P_2(x_3)\big).$$

Подставляем численные значения:

$$P_2(x_1) \approx 1, \quad P_2(x_2) \approx 0.04, \quad P_2(x_3) \approx -0.44.$$

Тогда

$$P_2(x_1) + P_2(x_2) + P_2(x_3) \approx 1 + 0.04 - 0.44 = 0.6,$$

и

$$a_2^{(\text{нач})} \approx \frac{5}{6} \cdot 0.6 \approx 0.5.$$

Интерпретация: умеренно положительный a_2, структура скорее «скошена» к началу, без ярко выраженного центрального пика.

8.3 Пример 3: ключевые слова только в конце текста

Теперь рассмотрим противоположную ситуацию — ключевые слова только в конце.

Шаг 1. Сигнал. Берём

$$f^{(\text{кон})} = (0, 0, 0, 1, 1, 1).$$

Шаг 2. Коэффициенты. Коэффициент $a_0^{(\text{кон})}$.

Сумма компонент такая же:

$$a_0^{(\text{кон})} = \frac{1}{6}(0 + 0 + 0 + 1 + 1 + 1) = \frac{3}{6} = 0.5.$$

Коэффициент $a_1^{(\text{кон})}$.

$$a_1^{(\text{кон})} = \frac{3}{6}\sum_{i=1}^{6} f_i^{(\text{кон})} x_i = \frac{1}{2}\big(x_4 + x_5 + x_6\big).$$

Подставляя $x_4 = 0.2, x_5 = 0.6, x_6 = 1$, имеем

$$x_4 + x_5 + x_6 = 0.2 + 0.6 + 1 = 1.8,$$

следовательно

$$a_1^{(\text{кон})} = \frac{1}{2} \cdot 1.8 = 0.9.$$

Интерпретация: сильный положительный тренд — вся активность в конце документа.

Коэффициент $a_2^{(\text{кон})}$.

$$a_2^{(\text{кон})} = \frac{5}{6}\big(P_2(x_4) + P_2(x_5) + P_2(x_6)\big).$$

Подставляем значения:

$$P_2(x_4) \approx -0.44, \quad P_2(x_5) \approx 0.04, \quad P_2(x_6) \approx 1.$$

Сумма:

$$P_2(x_4) + P_2(x_5) + P_2(x_6) \approx -0.44 + 0.04 + 1 = 0.6,$$

поэтому

$$a_2^{(\text{кон})} \approx \frac{5}{6} \cdot 0.6 \approx 0.5.$$

Заметим, что $a_0^{(\text{кон})}$ и $a_2^{(\text{кон})}$ совпадают с соответствующими коэффициентами из примера 2, а $a_1^{(\text{кон})}$ имеет противоположный знак. Это отражает симметрию: там активность была в начале, здесь — в конце.

8.4 Пример 4: один крупный пик в середине текста

Рассмотрим ситуацию, когда ключевые слова сосредоточены в середине документа.

Шаг 1. Сигнал. Пусть

$$f^{(\text{сер})} = (0, 0, 1, 1, 0, 0).$$

Это означает, что только токены на позициях 3 и 4 считаются ключевыми.

Шаг 2. Коэффициенты. Коэффициент $a_0^{(\text{сер})}$.

$$a_0^{(\text{сер})} = \frac{1}{6}(0 + 0 + 1 + 1 + 0 + 0) = \frac{2}{6} \approx 0.33.$$

Активна примерно треть позиций.

Коэффициент $a_1^{(\text{сер})}$.

$$a_1^{(\text{сер})} = \frac{3}{6} \sum_{i=1}^{6} f_i^{(\text{сер})} x_i = \frac{1}{2}(x_3 + x_4).$$

Подставляя $x_3 = -0.2, x_4 = 0.2$, получаем

$$x_3 + x_4 = -0.2 + 0.2 = 0,$$

поэтому

$$a_1^{(\text{сер})} = 0.$$

Интерпретация: нет тренда в сторону начала или конца — активность симметрично сосредоточена в центре.

Коэффициент $a_2^{(\text{сер})}$.

$$a_2^{(\text{сер})} = \frac{5}{6}\big(P_2(x_3) + P_2(x_4)\big).$$

Так как $x_3 = -0.2$ и $x_4 = 0.2$ дают одинаковые значения P_2, имеем:

$$P_2(x_3) \approx -0.44, \quad P_2(x_4) \approx -0.44,$$

и

$$P_2(x_3) + P_2(x_4) \approx -0.88.$$

Тогда

$$a_2^{(\text{сер})} \approx \frac{5}{6} \cdot (-0.88) \approx -0.73.$$

Интерпретация: заметно отрицательный a_2 соответствует паттерну с пиком в середине и меньшей активностью по краям.

8.5 Сравнение коэффициентов для разных паттернов

Соберём полученные значения (a_0, a_1, a_2) в одну таблицу (числа округлены):

Паттерн	a_0	a_1	a_2
Пример 1: начало–середина–конец	0.50	0.10	1.30
Пример 2: только начало	0.50	−0.90	0.50
Пример 3: только конец	0.50	0.90	0.50
Пример 4: пик в середине	0.33	0.00	−0.73

Эта таблица хорошо иллюстрирует интерпретацию коэффициентов:

- a_0 отражает общий уровень активности (долю ключевых позиций);
- знак и модуль a_1 показывают, смещена ли активность к началу ($a_1 < 0$) или к концу ($a_1 > 0$) текста;
- знак a_2 различает паттерн «края ярче, чем середина» ($a_2 > 0$) и паттерн «яркая середина, слабые края» ($a_2 < 0$).

Таким образом, даже для очень простого бинарного сигнала по тексту небольшое число коэффициентов Лежандра позволяет компактно и интерпретируемо описывать разные типы структурных паттернов по длине документа.