Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №10
по дисциплине ««Основы систем ИИ»

Тема Характеристики набора данных

Студент Батуев А.Г.

Группа ИУ7-36Б

Преподаватели Строганов Ю.В.

Москва, 2024

Содержание

ВВЕДЕНИЕ **4**

1 Аналитическая часть **5**

1.1 Математическое ожидание 5

1.2 Дисперсия 5

1.3 Ковариация 5

1.4 Корреляция Пирсона 5

1.5 Корреляция Спирмена 6

2 Конструкторская часть **7**

2.1 Гипотезы 7

2.1.1 Математическое ожидание 7

2.1.2 Дисперсия 7

2.1.3 Ковариация 7

2.1.4 Корреляция Пирсона 7

2.1.5 Корреляция Спирмена 7

2.2 Получение результатов 8

3 Технологическая часть **9**

3.1 Средства написания программы 9

3.2 Загрузка векторов документов 9

3.3 Выбор признаков 9

3.4 Получение результатов 11

4 Исследовательская часть **14**

4.1 Оборудование 14

4.2 Результаты исследования 14

4.3 Вывод 21

ЗАКЛЮЧЕНИЕ **22**

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ **23**

ВВЕДЕНИЕ

Цель: построить вектора документов для черепашьего набора данных, а также провести анализ статистических характеристик данных

Задачи:

— построить вектора документов

— провести анализ статистических данных

1 Аналитическая часть

Метод "мешка слов"широко используется в обработке текстов и позволяет представить текстовый документ в виде набора количественных признаков, где каждый признак соответствует частоте или наличию слова в документе. Однако, с увеличением числа документов и уникальных слов (признаков) размерность векторов значительно возрастает, что может замедлять вычисления и усложнять анализ.

Для оптимизации анализа и снижения размерности векторов используются различные статистические параметры, такие как дисперсия, математическое ожидание, ковариация, корреляции Пирсона и Спирмена.

1.1 Математическое ожидание

Математическое ожидание (среднее значение) показывает центральную тенденцию распределения значений признака. Оно указывает на усредненное значение, вокруг которого располагаются все значения признака.

Для признака математическое ожидание определяется как:
$Mean = \frac{1}{N}\sum_{j}^{N} x_{ij}$ [4]

1.2 Дисперсия

Дисперсия показывает, насколько каждый признак (его частота) в наборе данных отклоняется от своего среднего значения. Высокая дисперсия указывает на значительное разбросанное распределение значений вокруг среднего, тогда как низкая — на более сгруппированные значения.

Для признака дисперсия определяется как:
$Var = \frac{1}{N}\sum_{j=1}^{N}(x_{ij} - \overline{x}_i)$ [5]

1.3 Ковариация

Ковариация измеряет, как два признака изменяются вместе. Положительная ковариация означает, что признаки увеличиваются или уменьшаются одновременно, тогда как отрицательная указывает на противоположные направления.

Для признаков ковариация определяется как:
$Cov(x_i, x_j) = \frac{1}{N}\sum_{k=1}^{N}(x_{i,k} - \overline{x}_i)(x_{j,k} - \overline{x}_j)$ [6]

1.4 Корреляция Пирсона

Корреляция Пирсона измеряет линейную зависимость между двумя признаками. Значение корреляции варьируется от -1 до 1, где 1 означает полную положительную зависимость, -1 — полную отрицательную, а 0 — отсутствие линейной зависимости.

Для признаков корреляция Пирсона определяется как:
$Corr_{Pearson} = \frac{Cov(x_i,x_j)}{\sigma_i \sigma_j}$ [7]

1.5 Корреляция Спирмена

Корреляция Спирмена оценивает монотонную зависимость между признаками. В отличие от корреляции Пирсона, она подходит для оценки нелинейных зависимостей.

Для признаков корреляция Спирмена определяется как:
$Corr_{Spearman} = 1 - \frac{6\sum_{k=1}^{N}(R_{x_i,k} - R_{x_j,k})^2}{N(N^2-1)}$ [8]

Вывод

Каждый из рассмотренных способов уменьшения количества параметров имеет свои ограничения, но, в целом, их использование предполагает выделение дублирующих или неинформативных признаков.

2 Конструкторская часть

Для анализа близости используется вектор документа, способ получения которого описывался в лабораторной работе №7. Метод получения векторов документа ничем не отличается от описанного в нём. Также из той же лабораторной работы используются метрики для получения матрицы близости документов. Поэтому в данной лабораторной работе они подробно описаны не будут.

2.1 Гипотезы

2.1.1 Математическое ожидание

По результатам расчета математического ожидания предполагается, что признаки со средним значением близким к нулю (в данном случае близким к нуля считается тот, что встречается в 2 документах из 50), не несут важной информации, поэтому могут быть удалены.

2.1.2 Дисперсия

Дисперсия позволяет выявить признаки с низкой изменчивостью. Признаки с низкой изменчивостью считаться те, что отличаются от среднего по всем признакам не более чем на одно значение среднего. Предполагается, что они не несут значимой информации для различения документов, так как мало отличаются друг от друга.

2.1.3 Ковариация

Так как важен лишь сам факт зависимости одного признака от другого, а не направление зависимости, было принять решения брать ковариацию по модулю. Предполагается, что высокая ковариация может говорить о том, что признаки дублируют друг друга. Поэтому один из них можно исключить, так как они отражают схожую информацию о документе.

2.1.4 Корреляция Пирсона

Как и при ковариации, важен лишь сам факт зависимости, поэтому корреляция будет браться по модулю. Предполагется, что признаки с высокой корреляцией Пирсона можно объединить или исключить один из них, так как они отражают схожую линейную информацию.

2.1.5 Корреляция Спирмена

Ситуация аналогична корреляции Пирсона.

2.2 Получение результатов

Используя выше описанные методы, получим список признаков, которые могут быть удалены. После обновления векторов можно будет приступить к построению матрицы корреляции и получению хитмапов.

Вывод

Гипотезы, выдвинутые в исследовании, касаются отбора признаков для улучшения качества представления документов. В результате, путем исключения нерелевантных признаков обновленные векторы документов позволят построить матрицу корреляции и визуализировать взаимосвязи с помощью тепловой карты, что улучшит точность анализа текстовых данных.

3 Технологическая часть

3.1 Средства написания программы

Для реализации программного обеспечения были использованы следующие средства:

— среда разработки PyCharm 2023 community edition [16]

— язык разработки Python 3.10 [17]

Используемые библиотеки:

— os — для доступа к файлам системы и создания папок/файлов [9]

— numpy — для быстрой работы с массивами, сохранения промежуточных данных и связи их с другими библиотеками [10]

— matplotlib.pyplot — для отображения графиков [11]

— seaborn — для построения графиков [12]

— sklearn.metrics.pairwise.cosine_similarity — косинусная мера близости [13]

— sklearn.metrics.jaccard_score — метод Жаккарда [13]

— pandas — все использованные статистические функции [14]

— pathlib.Path — доступ к файлам системы [15]

3.2 Загрузка векторов документов

Листинг 3.1 — Загрузка векторов документов

```
def load_vectors_from_directory(base_dir):
    vectors = []
    files = []
    for file_path in base_dir.rglob('vectors/*.txt'):
        files.append(file_path.name)
        vector = np.loadtxt(file_path)
        vectors.append(vector)
    return np.vstack(vectors), files
```

3.3 Выбор признаков

Листинг 3.2 — Выбор признаков по математическому ожиданию

```
def to_del_by_mean(df: pd.DataFrame):
    threshold = 0.04
    df_mean = df.mean()
    list_to_remove = df_mean[df_mean <= threshold].index.to_list()
    return list_to_remove
```

Листинг 3.3 — Выбор признаков по дисперсии

```
def to_del_by_dispersion(df: pd.DataFrame):
    threshold = 0.99
    df_var = df.var()
    mean_df_var = df_var.mean()
    lower_bound = mean_df_var * (1 - threshold)
    upper_bound = mean_df_var * (1 + threshold)
    list_to_remove = df_var[(df_var > lower_bound) & (df_var <
        upper_bound)].index.to_list()
    return list_to_remove
```

Листинг 3.4 — Выбор признаков по ковариации

```
def to_del_by_covariance(df: pd.DataFrame):
    threshold = 0.6
    df_cov = df.cov().abs()
    df_cov = df_cov.where(np.tril(df_cov, k=-1) > threshold)
    to_drop = set()
    for col in df_cov.columns:
        if col not in to_drop:
            correlated_features = df_cov[col].dropna().index
            to_drop.update(correlated_features.difference([col]))
    return to_drop
```

Листинг 3.5 — Выбор признаков по Пирсону

```
threshold = 0.99
df_pearson = df.corr(method='pearson').abs()
df_pearson = df_pearson.where(np.tril(df_pearson, k=-1) > threshold)
to_drop = set()
for col in df_pearson.columns:
    if col not in to_drop:
        correlated_features = df_pearson[col].dropna().index
        to_drop.update(correlated_features.difference([col]))
return to_drop
```

Листинг 3.6 — Выбор признаков по Спирмену

```
threshold = 0.99
df_pearson = df.corr(method='pearson').abs()
df_pearson = df_pearson.where(np.tril(df_pearson, k=-1) > threshold)
to_drop = set()
for col in df_pearson.columns:
    if col not in to_drop:
        correlated_features = df_pearson[col].dropna().index
        to_drop.update(correlated_features.difference([col]))
```

```
    return to_drop
```

3.4 Получение результатов

Листинг 3.7 — Построение тепловых карт

```
def build_heatmap(data, savepath, xlables, ylabels):
    plt.figure(figsize=(19.2, 10.8))
    sea.heatmap(data, xticklabels=xlables, yticklabels=ylabels, square
        =True, linewidths=.3, cmap="YlGnBu")
    plt.savefig(savepath, dpi=100)
    plt.clf()
```

Листинг 3.8 — Внесение измений и построение тепловых карт

```
to_drop_by_dispersion = determining.to_del_by_dispersion(df)
vectors_updated_dispersion = np.delete(vectors, list(
    to_drop_by_dispersion), axis=1)
pearson_matrix, cosine_matrix, jac_matrix = get_corr_matrix(
    vectors_updated_dispersion)
build_heatmap(pearson_matrix, RES_DIR / "Heatmap_dispersion_pearson.
    png", files, files)
build_heatmap(cosine_matrix, RES_DIR / "Heatmap_dispersion_cosine.png"
    , files, files)
build_heatmap(jac_matrix, RES_DIR / "Heatmap_dispersion_jac.png",
    files, files)

to_drop_by_mean = determining.to_del_by_mean(df)
vectors_updated_mean = np.delete(vectors, list(to_drop_by_mean), axis
    =1)
pearson_matrix, cosine_matrix, jac_matrix = get_corr_matrix(
    vectors_updated_mean)
build_heatmap(pearson_matrix, RES_DIR / "Heatmap_mean_pearson.png",
    files, files)
build_heatmap(cosine_matrix, RES_DIR / "Heatmap_mean_cosine.png",
    files, files)
build_heatmap(jac_matrix, RES_DIR / "Heatmap_mean_jac.png", files,
    files)

to_drop_by_covariance = determining.to_del_by_covariance(df)
vectors_updated_covariance = np.delete(vectors, list(
    to_drop_by_covariance), axis=1)
pearson_matrix, cosine_matrix, jac_matrix = get_corr_matrix(
```

```
        vectors_updated_covariance)
    build_heatmap(pearson_matrix, RES_DIR / "Heatmap_covariance_pearson.
        png", files, files)
    build_heatmap(cosine_matrix, RES_DIR / "Heatmap_covariance_cosine.png"
        , files, files)
    build_heatmap(jac_matrix, RES_DIR / "Heatmap_covariance_jac.png",
        files, files)

    to_drop_by_pearson = determining.to_del_by_pearson(df)
    vectors_updated_pearson = np.delete(vectors, list(to_drop_by_pearson),
         axis=1)
    pearson_matrix, cosine_matrix, jac_matrix = get_corr_matrix(
        vectors_updated_pearson)
    build_heatmap(pearson_matrix, RES_DIR / "Heatmap_pearson_pearson.png",
         files, files)
    build_heatmap(cosine_matrix, RES_DIR / "Heatmap_pearson_cosine.png",
        files, files)
    build_heatmap(jac_matrix, RES_DIR / "Heatmap_pearson_jac.png", files,
        files)

    to_drop_by_spearman = determining.to_del_by_spearman(df)
    vectors_updated_spearman = np.delete(vectors, list(to_drop_by_spearman
        ), axis=1)
    pearson_matrix, cosine_matrix, jac_matrix = get_corr_matrix(
        vectors_updated_spearman)
    build_heatmap(pearson_matrix, RES_DIR / "Heatmap_spearman_pearson.png"
        , files, files)
    build_heatmap(cosine_matrix, RES_DIR / "Heatmap_spearman_cosine.png",
        files, files)
    build_heatmap(jac_matrix, RES_DIR / "Heatmap_spearman_jac.png", files,
         files)
```

Вывод

В технологической части были разработаны и реализованы программные методы для автоматизированного отбора признаков и анализа близости документов. Методы для отбора признаков включают вычисления на основе математического ожидания, дисперсии, ковариации, а также корреляций по Пирсону и Спирмену. Эти методы позволили выявить малозначимые и дублирующие признаки, которые затем исключались из векторов документов для повышения точности анализа.

На завершающем этапе была построена матрица корреляции и визуализированы тепло-

вые карты взаимосвязей между документами, что позволило наглядно представить результаты отбора признаков и оценить качество полученных данных.

4 Исследовательская часть

Цель исследования — разработка и реализация методов отбора значимых признаков для анализа и оценки близости документов, позволяющих уменьшить избыточность данных и повысить точность результатов.

4.1 Оборудование

Характеритстики ноутбука:

— процессор intel-core i5-12500H [19]

— ОЗУ 16 Гб DDR4

— ОС Windows 11 [18]

4.2 Результаты исследования

Длина до всегда состаляет 6917 признаков, поэтому в таблице показывается только длина после.

Метрика	**Длина после**
Дисперсия	558
Математическое ожидание	1322
Ковариация	6805
Пирсон	1600
Спирмен	1443

Таблица 4.1 — Результаты сокращения векторов документов

Метрика	**Длина после**
Дисперсия	6917
Математическое ожидание	1233
Ковариация	6849
Пирсон	5824
Спирмен	5816

Таблица 4.2 — Результаты сокращения векторов документов по тематике Анамнез

Метрика	**Длина после**
Дисперсия	5964
Математическое ожидание	1408
Ковариация	6736
Пирсон	5702
Спирмен	5693

Таблица 4.3 — Результаты сокращения векторов документов по тематике Биохимия крови

Метрика	**Длина после**
Дисперсия	6917
Математическое ожидание	1054
Ковариация	6823
Пирсон	5974
Спирмен	5972

Таблица 4.4 — Результаты сокращения векторов документов по тематике Комментаторы

Метрика	**Длина после**
Дисперсия	6917
Математическое ожидание	1439
Ковариация	6757
Пирсон	5621
Спирмен	5613

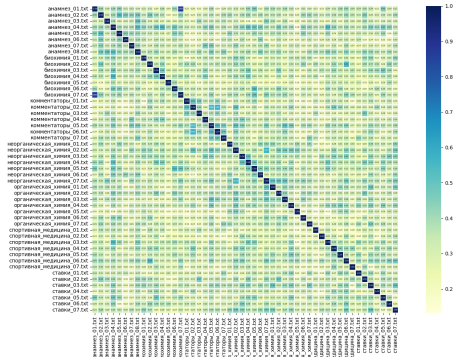
Таблица 4.5 — Результаты сокращения векторов документов по тематике Неорганическая химия

Метрика	**Длина после**
Дисперсия	5820
Математическое ожидание	1445
Ковариация	6796
Пирсон	5592
Спирмен	5579

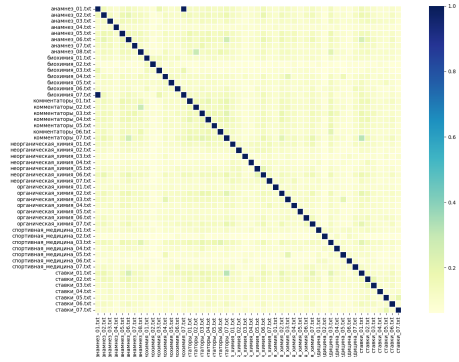
Таблица 4.6 — Результаты сокращения векторов документов по тематике Органическая химия

Метрика	**Длина после**
Дисперсия	5719
Математическое ожидание	1454
Ковариация	6804
Пирсон	5894
Спирмен	5881

Таблица 4.7 — Результаты сокращения векторов документов по тематике Спортивная медицина

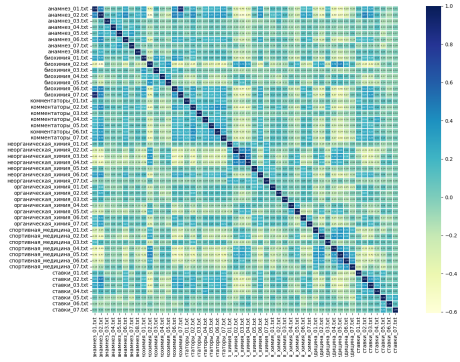


а) До сокращения вектора

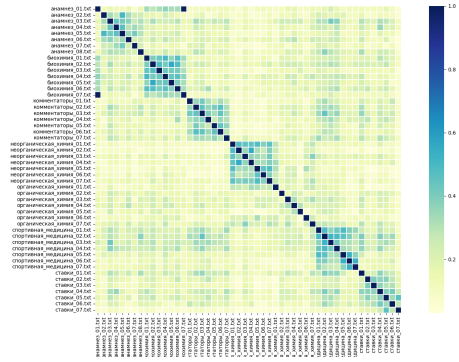


б) После сокращения вектора

Рисунок 4.1 — Сравнение методом Жаккарда уменьшенного вектора с помощью среднего значения

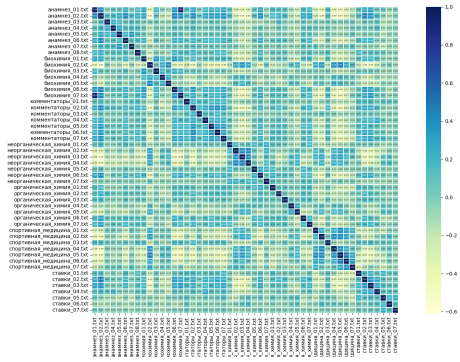


а) До сокращения вектора

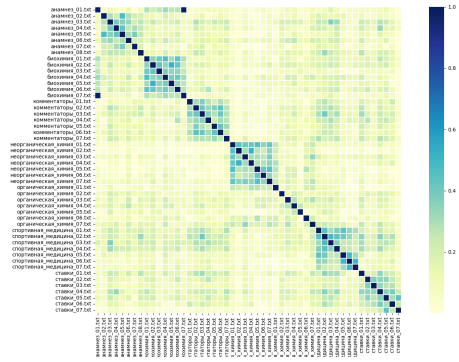


б) После сокращения вектора

Рисунок 4.2 — Сравнение косинусным методом уменьшенного вектора с помощью среднего значения

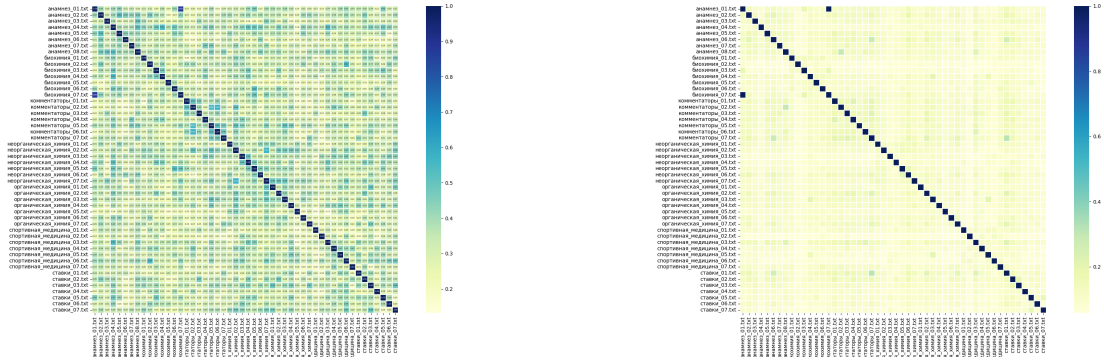


а) До сокращения вектора



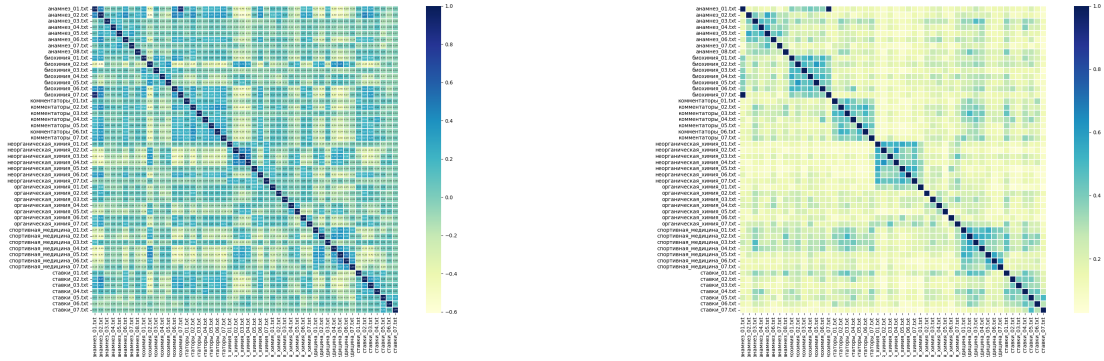
б) После сокращения вектора

Рисунок 4.3 — Сравнение методом Пирсона уменьшенного вектора с помощью среднего значения



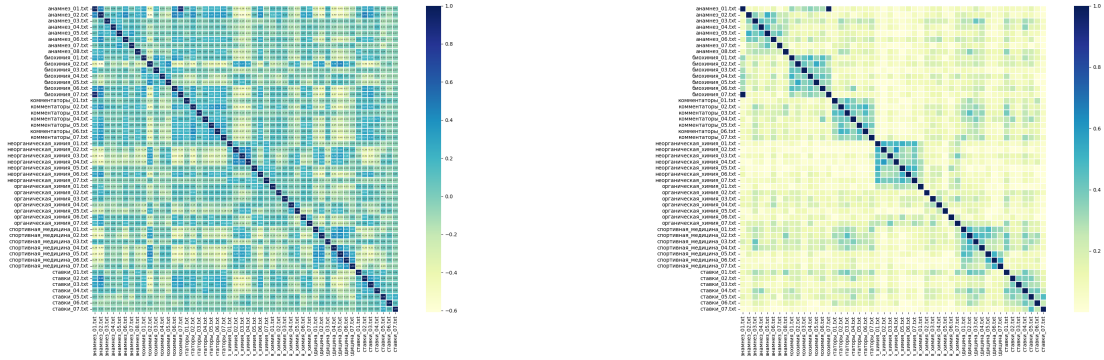
а) До сокращения вектора б) После сокращения вектора

Рисунок 4.4 — Сравнение методом Жаккарда уменьшенного вектора с помощью диспресии



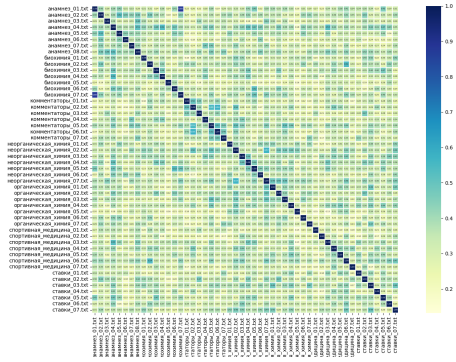
а) До сокращения вектора б) После сокращения вектора

Рисунок 4.5 — Сравнение косинусным методом уменьшенного вектора с помощью диспресии

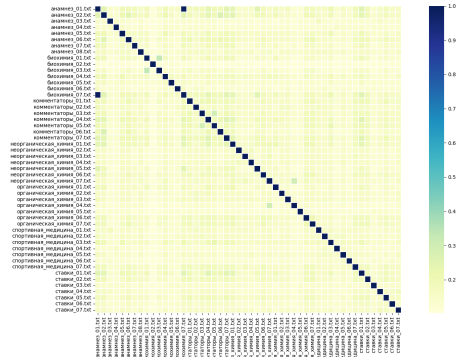


а) До сокращения вектора б) После сокращения вектора

Рисунок 4.6 — Сравнение методом Пирсона уменьшенного вектора с помощью диспресии

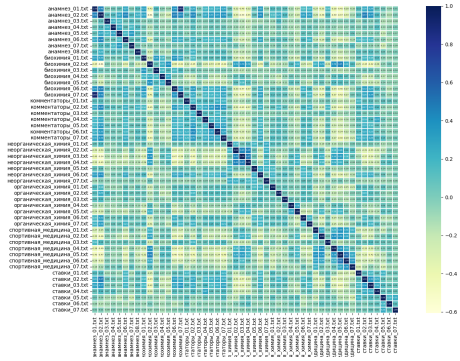


а) До сокращения вектора

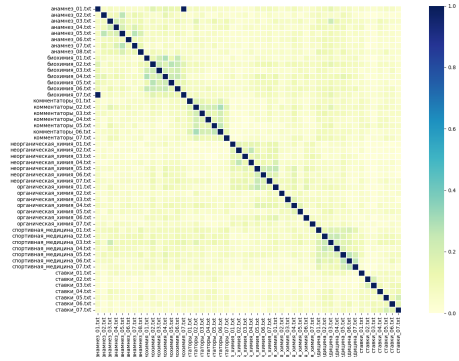


б) После сокращения вектора

Рисунок 4.7 — Сравнение методом Жаккарда уменьшенного вектора с помощью ковариации

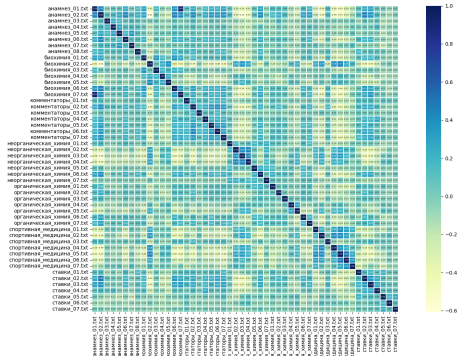


а) До сокращения вектора

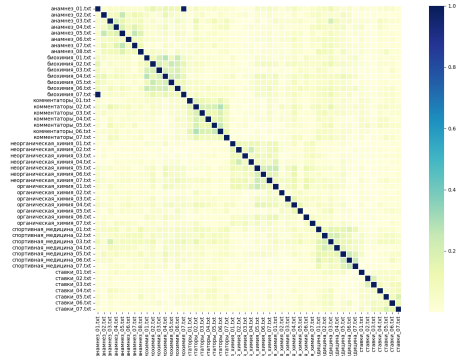


б) После сокращения вектора

Рисунок 4.8 — Сравнение косинусным методом уменьшенного вектора с помощью ковариации

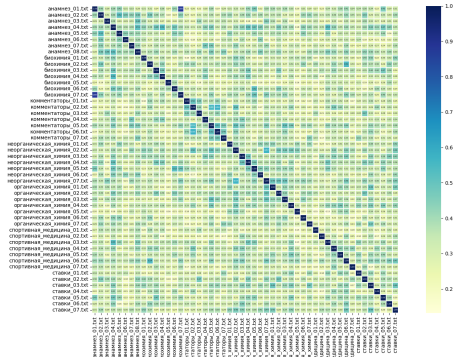


а) До сокращения вектора

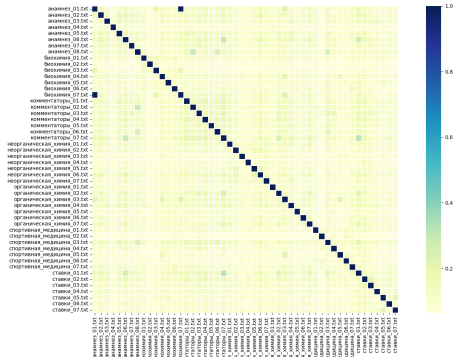


б) После сокращения вектора

Рисунок 4.9 — Сравнение методом Пирсона уменьшенного вектора с помощью ковариации

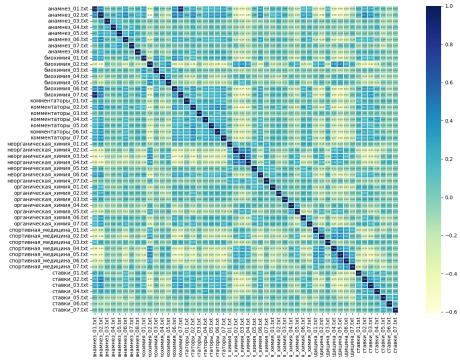


а) До сокращения вектора

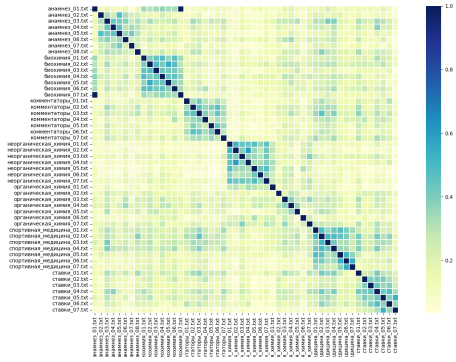


б) После сокращения вектора

Рисунок 4.10 — Сравнение методом Жаккарда уменьшенного вектора с помощью корреляции Пирсона

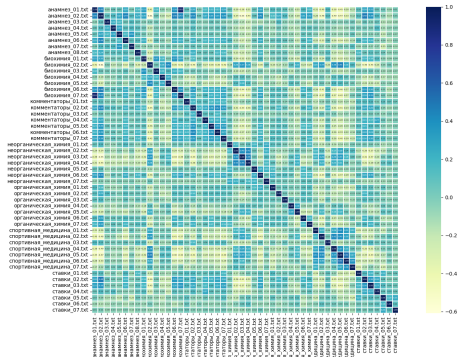


а) До сокращения вектора

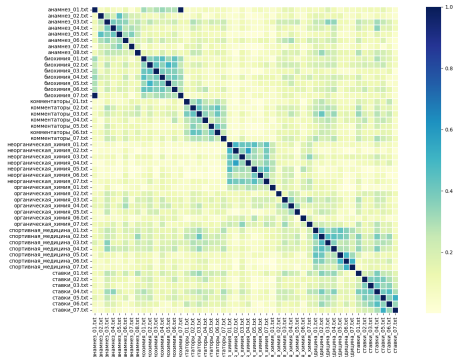


б) После сокращения вектора

Рисунок 4.11 — Сравнение косинусным методом уменьшенного вектора с помощью корреляции Пирсона

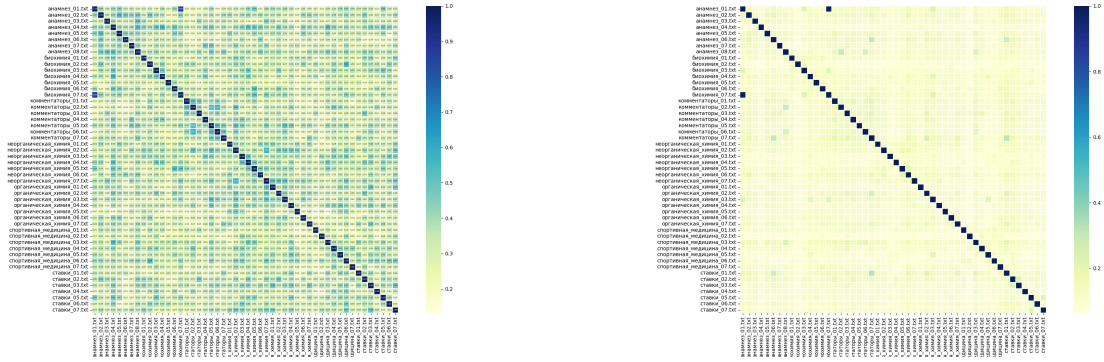


а) До сокращения вектора



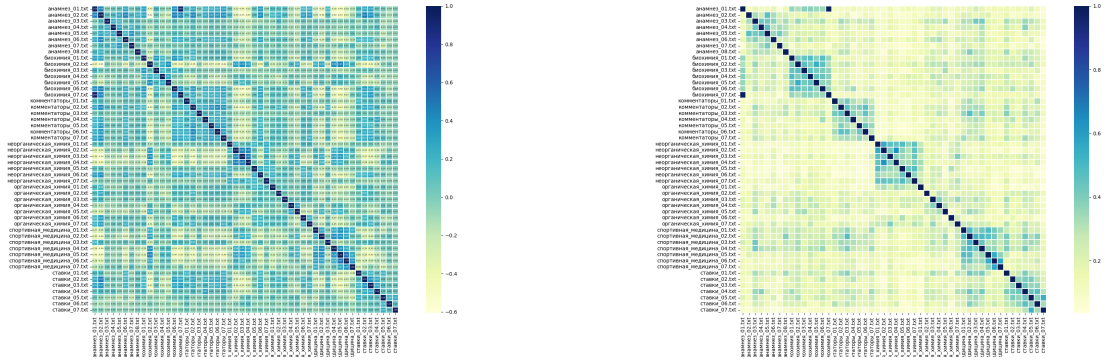
б) После сокращения вектора

Рисунок 4.12 — Сравнение методом Пиросона уменьшенного вектора с помощью корреляции Пирсона



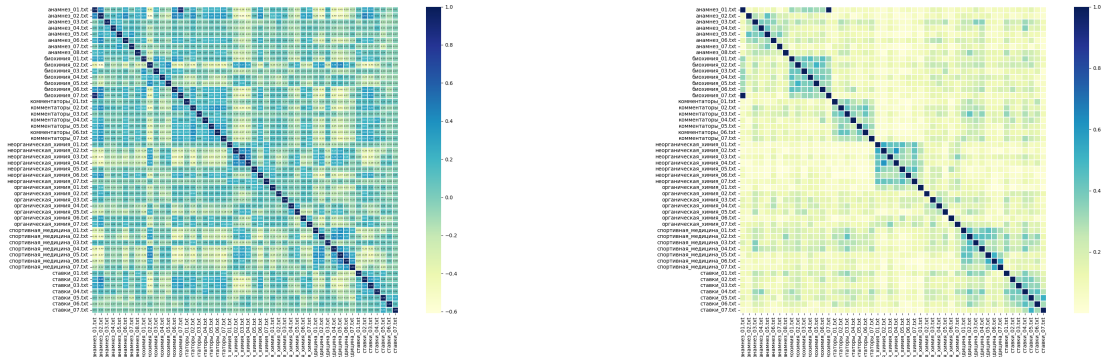
а) До сокращения вектора б) После сокращения вектора

Рисунок 4.13 — Сравнение методом Жаккарда уменьшенного вектора с помощью корреляции Спирмена



а) До сокращения вектора б) После сокращения вектора

Рисунок 4.14 — Сравнение косинусным методом уменьшенного вектора с помощью корреляции Спирмена



а) До сокращения вектора б) После сокращения вектора

Рисунок 4.15 — Сравнение методом Пирсона уменьшенного вектора с помощью корреляции Спирмена

4.3 Вывод

Метрика Жаккарда оказалась менее подходящей для анализа: тепловая карта получилась слишком бледной, что затрудняет визуальную оценку результатов. В то же время косинусная метрика и корреляция Пирсона дали почти одинаковые результаты, что позволяет считать их взаимозаменяемыми для этого исследования.

ЗАКЛЮЧЕНИЕ

В ходе исследования удалось значительно сократить размерность вектора признаков — с 6917 до примерно 558–1454 признаков в зависимости от метода. Однако дальнейшее сокращение оказалось невозможным из-за проблем с визуализацией, так как центральные квадраты на тепловых картах перестают отображаться.

Все методы уменьшения размерности показали свою эффективность, за исключением метода, основанного на ковариации. При отборе признаков по ковариации даже уменьшение до около 6736 признаков привело к значительным потерям информации, что сделало этот метод непригодным для данных, с которыми проводилось исследование.

Таким образом, исследование показало, что предложенные методы позволяют уменьшить размерность вектора, но при этом сокращение недостаточно значительное, а возможные потери данных могут быть ощутимыми. Для достижения лучшего баланса между точностью и размерностью вектора стоит рассмотреть другие методы уменьшения размерности.

Из анализа сокращения размерности вектора признаков для различных тематических групп делается вывод, что эффективность методов отбора признаков варьируется в зависимости от тематики данных. Наибольшее сокращение размерности в большинстве тематик достигается при использовании метода отбора по Спирмену, что свидетельствует о его способности эффективно отбирать значимые признаки для конкретных типов данных. В то же время метод дисперсии, несмотря на свою эффективность в ряде случаев, часто оставляет значительно больше признаков, что указывает на его ограниченную применимость для некоторых тематических групп.

Метод ковариации, как и было отмечено ранее, показал наименьшую эффективность во всех случаях, так как оставил практически максимальное количество признаков, что приводит к значительным потерям информации при попытке более агрессивного сокращения. Это подтверждает, что для данных, используемых в исследовании, метод ковариации является наименее подходящим для уменьшения размерности.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Seaborn: statistical data visualization URL: https://seaborn.pydata.org/ (Дата обращения 10.11.2024)

2. Principal component analysis (PCA). URL: https://scikit-learn.org/dev/modules/generated/sklearn.decomposition.PCA.html (Дата обращения 10.11.2024)

3. Метод эффективного расчета матрицы ближайших соседей для полнотекстовых документов. URL: https://cyberleninka.ru/article/n/metod-effektivnogo-rascheta-matritsy-blizhayshih-sosedey-dlya-polnotekstovyh-dokumentov (Дата обращения 10.11.2024)

4. Математическое ожидание функции URL: https://cyberleninka.ru/article/n/matematicheskoe-ozhidanie-funktsii (Дата обращения 10.11.2024)

5. Во славу дисперсии URL: https://cyberleninka.ru/article/n/vo-slavu-dispersii (Дата обращения 10.11.2024)

6. Оценивание средних и ковариаций нечетко-случайных величин URL: https://cyberleninka.ru/article/n/otsenivanie-srednih-i-kovariatsiy-nechetko-sluchaynyh-velichin (Дата обращения 10.11.2024)

7. Note on Regression and Inheritance in the Case of Two Parents (Карл Пиросон) - Лондон: Royal Society, 1904 - №75. - стр. 347–352

8. Анализ значений коэффициента ранговой корреляции Спирмена URL: https://cyberleninka.ru/article/n/analiz-znacheniy-koeffitsienta-rangovoy-korrelyatsii-spirmena (Дата обращения 10.11.2024)

9. Модуль os в Python, доступ к функциям ОС URL: https://docs-python.ru/standart-library/modul-os-python/ (Дата обращения 10.11.2024)

10. NumPy Documentation URL: https://numpy.org/doc/ (Дата обращения 10.11.2024)

11. Matplotlib 3.9.2 documentation URL: https://matplotlib.org/stable/index.html (Дата обращения 10.11.2024)

12. seaborn: statistical data visualization URL: https://seaborn.pydata.org/ (Дата обращения 10.11.2024)

13. Документация sklearn URL: https://scikit-learn.org/stable/ (Дата обращения 10.11.2024)

14. Документация Pandas https://pandas.pydata.org/pandas-docs/stable/index.html (Дата обращения 10.11.2024)

15. Документация Pathlib https://docs.python.org/3/library/pathlib.html (Дата обращения 10.11.2024)

16. Среда разработки Pycharm 2023 Community edition URL: https://www.jetbrains.com/ru-ru/pycharm/download/other.html

17. Язык разработки Python 3.10 URL: https://www.python.org/downloads/release/python-3100/

18. ОС Windows 11 URL: https://www.microsoft.com/ru-ru/software-download/windows11

19. Intel i5-12500H URL: https://www.intel.com/content/www/us/en/products/sku/96141/intel-core-i512500h-processor-18m-cache-up-to-4-50-ghz/specifications.html?wapkw=12500h