ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ТЕЛЕКОММУНИКАЦИИ
INFORMATION TECHNOLOGY AND TELECOMMUNICATIONS

УДК 004.8

DOI: 10.21822/2073-6185-2025-52-1-162-172 Оригинальная статья /Original article

Модификация автоматического метода извлечения причинно-следственных связей, основанного на шаблонах и Байесовском классификаторе

Х.Б. Штанчаев[1,2], З.Т. Мугутдинов[1]
[1]Дагестанский государственный технический университет,
[1]367015, г. Махачкала проспект Имама Шамиля 70, Россия,
[2]ООО «Салаватстекло Каспий»,
[2]368080, Республика Дагестан, с. Коркмаскала, Заводская ул, зд.1 стр.1, Россия

Резюме. Цель. Целью исследования является модификация автоматического метода извлечения причинно-следственных связей. **Метод.** Исследование основано на оригинальном методе Антоние Сорженте с его последующей модификацией. **Результат.** Предложен метод извлечения причинно-следственных связей. Метод предполагает комбинированное использование статистических данных и машинных методов. Оригинальный метод был модифицирован переводом работы метода на современные библиотеки, такие как NLTK и Spacy. Правила, сформированные автором, были переработаны и добавлены в модуль Dependency Matcher библиотеки Spacy. Количество ключевых слов по каждому правилу были увеличены. Метод так же предполагает учет синонимов, подсчет Байесовской статистики и сглаживание Лапласа для нулевых вероятностей. Исходя из разности данных с ПСС и без, был введен коэффициент multiplier для компенсации перекоса классов в данных. **Вывод.** Разработанный метод был протестирован на исходных данных оригинального метода и показал улучшенные метрики относительно оригинального метода на тренировочных и тестовых данных.

Ключевые слова: причинно-следственные связи, обработка естественного языка, NLTK, Spacy, причинность, причина, следствие, статические методы, нестатические методы, машинные методы

Для цитирования: Х.Б. Штанчаев, З.Т. Мугутдинов. Модификация автоматического метода извлечения причинно-следственных связей, основанного на шаблонах и Байесовском классификаторе. Вестник Дагестанского государственного технического университета. Технические науки. 2025; 52(1):162-172. DOI:10.21822/2073-6185-2025-52-1-162-172

Modification of an automatic method for extracting causal relationships based on templates and a Bayesian classifier

H.B. Shtanchaev[1,2], Z.T. Mugutdinov[1]
[1]Daghestan State Technical University,
[1]70 I. Shamil Ave., Makhachkala 367015, Russia,
[2]OOO "Salavatsteklo Caspian",
[2]Zavodskaya St., Republic of Daghestan, s. Korkmaskala 368080, Russia

Abstract. **Objective.** The aim of the study is to modify the automatic method for extracting cause-and-effect relationships. **Method.** The study is based on the original method of Antonie Sorgente with its subsequent modification. **Result.** A method for extracting cause-and-effect relationships is proposed. The method involves the combined use of statistical data and machine methods. The original method was modified by translating the method to modern libraries such as NLTK and Spacy. The rules formed by the author were reworked and added to the Dependency Matcher module of the Spacy library. The number of keywords for each rule was increased. The method also takes into account synonyms, calculates Bayesian statistics and

smooths Laplace for zero probabilities. Based on the difference in data with and without PSS, a multiplier coefficient was introduced to compensate for the skew of classes in the data. **Conclusion.** The developed method was tested on the original data of the original method and showed improved metrics relative to the original method on training and test data.

Keywords: cause and effect, natural language processing, NLTK, Spacy, causality, cause, effect, static methods, non-static methods, machine methods

For citation: H.B. Shtanchaev, Z.T. Mugutdinov. Modification of an automatic method for extracting causal relationships based on templates and a Bayesian classifier. Herald of Daghestan State Technical University. Technical Sciences. 2025; 52(1):162-172. (In Russ) DOI:10.21822/2073-6185-2025-52-1-162-172.

Введение. Понимание причинно-следственных связей является фундаментальным для человеческого разума. Человек постоянно стремится разобраться, почему происходят события, чтобы предвидеть будущее и принимать обоснованные решения. В современном мире, где информация доступна в огромных объемах, а технологии развиваются стремительно, способность извлекать причинно-следственные связи из данных становится все более важной и имеет решающее значение для прогресса во множестве отраслей, от медицины и финансов до маркетинга и образования.

Постановка задачи. Важность и актуальность задачи извлечения причинно-следственных связей из данных обоснована несколькими причинами:

- Улучшение поиска информации и ответов на вопросы. Современные поисковые системы и системы вопросов-ответов могут использовать извлеченные причинно-следственные связи для предоставления более релевантных и точных ответов на запросы пользователей.

- Анализ больших данных и выявление скрытых шаблонов. В бизнесе, науке и других областях анализ больших объемов текстовых данных с целью выявления причинно-следственных связей помогает выявить скрытые паттерны и тенденции. Это может быть полезно для предсказания будущих событий, понимания основных факторов, влияющих на результаты, и принятия обоснованных решений.

- Поддержка принятия решений. В различных отраслях, таких как медицина, финансы, производство и управление, понимание причинно-следственных связей помогает руководителям и специалистам принимать более информированные решения. Например, врачи могут использовать результаты анализа медицинских текстов для выявления причин заболеваний и выбора оптимальных методов лечения.

- Создание новых знаний и научные исследования. В научных исследованиях автоматическое извлечение причинно-следственных связей способствует обогащению знаний, позволяя исследователям быстрее находить релевантную информацию и строить новые гипотезы. Это особенно актуально в областях, где количество научных публикаций растет экспоненциально.

- Анализ социальных явлений и политики. Извлечение причинно-следственных связей из новостей, социальных медиа и других источников помогает аналитикам и исследователям лучше понимать социальные и политические процессы, их причины и последствия. Это может быть полезно для разработки политики и стратегий, направленных на решение социальных проблем.

- Интеллектуальные системы и автоматизация. Включение анализа причинно-следственных связей в интеллектуальные системы и роботов способствует созданию более автономных и умных систем, способных понимать и реагировать на окружающий мир более адекватно. Это открывает новые возможности для автоматизации различных процессов и задач.

Методы исследования. Для решения задачи автоматического извлечения причинно-следственных связей используются различные подходы и концепции. Практически все методы извлечения ПСС были построены в рамках двух больших концепций:

1. **Нестатистические методы.** Использовали построенные вручную лингвистические шаблоны. ПСС, не подходящие в выстроенные шаблоны, определялись некорректно, а зачастую просто игнорируются [1].
2. **Статистические и машинные методы**. Особенность методов этой концепции, в отличие от нестатистических методов, заключается в том, что методы сосредоточены на поиске шаблонов для извлечения ПСС [2].

Как показал анализ методов извлечения [1,2] нестатистические методы точны, но имеют сильную привязку к предметной области, машинные методы напротив хуже по точности, однако, не зависят от предметной области. Главным недостатком всех моделей является то, что они рассчитаны на один язык. Так же стоит упомянуть, что идея автоматических машинных методов извлечения ПСС предполагает наличия больших корпусов языка, таких как WordNet[3], VerbNet[4], TREC[5], Википедия и т.д.

В работе, опубликованной в 2013 году, Антонио Соргенте [6] описал один из методов автоматического извлечения ПСС. Идея автора заключается в извлечении из предложений набора S пар «причина-следствие»:

$$S = \{(C_1E_1), (C_2E_2), \dots, (C_iE_i)\} \quad (1)$$

где: (C_iE_i) – i-ая пара «причина-следствие» в предложении S.

Метод состоит из 4 шагов.

Проверка на наличие шаблонов. Предложение проверяется на заранее определенных лексико-синтаксических шаблонах, наиболее уместных в предложениях с ПСС. Для успешного выполнения данного шага автор определил языковые конструкции, которые определяют структуру ПСС в предложении. Наиболее распространенными являются следующие конструкции:

Простые причинные глаголы – одиночные глаголы, имеющие возможность «воздействия» на объект (изменять, запускать, увеличивать, уменьшать и т. д.).

Фразовые глаголы - словосочетания, состоящие из глагола, за которым следует частица (например, привести к).

Существительные + предлог - выражения, состоящие из существительного, за которым следует предлог (например, причина в..).

Страдательные причинные глаголы - глаголы в страдательном залоге, за которыми следует предлог.

Одиночные предлоги - предлоги, которые могут использоваться для связи «причины» и «следствия», (например, от, после и так далее).

Анализ(парсинг) предложения. Если языковая конструкция в предложении найдена, то предложение анализируется с помощью парсера, чтобы выявить структуру и зависимости между словами.

Применение правил к найденному предложению. Если в каком-либо предложении была найдена языковая конструкция, то к нему применяются правила. Для каждой из категорий языковых конструкций автор определил свой набор заранее определенных правил, согласно которым определяются пары причина-следствие.

Основным же правилом для определения ПСС является:

$$cause(S, P, C) \wedge effect(S, P, E) \rightarrow cRel(S, C, E) \quad (2)$$

где: cause(S, P, C) означает, что C является причиной в S в соответствии с шаблоном P, в то время как effect(S, P, E) означает, что E является следствием в C по отношению к P.

Применение классификатора. Для уменьшения числа ложных пар, полученных из неоднозначных шаблонов, используется Байесовский классификатор, который помогает отсеять некорректные пары. В свою очередь фильтрация байесовским классификатором проводится также в несколько этапов. На этапе подготовки данных собираются наборы пар «причина-следствие», которые были извлечены из текста с помощью правил. Такие пары могут быть как потенциально правильными, так и неправильными. Так же на этом этапе проводится разметка данных. На этапе извлечения признаков извлекается набор

признаков, который будет использоваться для обучения классификатора. Признаки, согласно автору, включают в себя: лексические признаки - слова находящиеся между C и E; семантические признаки - синонимы, гипонимы для C и E и т.д.; зависимости прямые зависимости, определенные правилами, указанными выше. Обучение классификатора проходит в несколько важных этапов. На основе размеченных пар и извлеченных признаков генерируется обучающая выборка для обучения классификатора. Важными признаками являются оцененные вероятности. Так для каждого признака f, связанного с парой r, оценивается вероятность P (f | c_i), где c_i – это класс [7]. В описываемом методе это два класса – причинный и не причинный.

Согласно Байесовскому классификатору, подсчитываются количества случаев, когда признак f встречается в причинных и не причинных парах затем делится на общее количество случаев. Важным моментом метода является возможность возникновения нулевых вероятностей для признаков и предотвращение их появления. Использование Лапласова сглаживания позволяет присвоить ненулевые вероятности даже тем признакам, которых нет в обучающей выборке. Классический классификатор выглядит следующим образом [8]:

$$P(ci \mid r) = \frac{P(r)P(r \mid c_i)}{P(c_i)} \quad (3)$$

где: $P(r \mid c_i)$ - вероятность наблюдать пару r, если она принадлежит классу c_i, $P(c_i)$- априорная вероятность класса c_i, P(r) - общая вероятность наблюдать пару r.

Лапласово же преобразование добавляет небольшую константу (обычно 1) к числителю и знаменателю при оценке вероятности признака [9]:

$$P(f|c_i) = \frac{N(c_i \mid f) + \alpha}{N(c_i) + \alpha|F|} \quad (4)$$

где: $N(c_i \mid f)$- количество случаев, когда признак *f* встречается в классе c_i; $N(c_i)$ - общее количество случаев в классе c_i; $|F|$ - общее количество уникальных признаков. α - сглаживающий параметр (обычно α=1) [9].

Предлагаемая модель извлечения ПСС и модификация оригинального метода основывается на оригинальном методе, описанном выше, с учетом следующих изменений:

- удаление 4-го правила из рассматриваемых автором; правило было исключено в связи с небольшим дублированием с 3-им.
- увеличение количества слов-шаблонов; слова шаблоны были подобраны вручную по аналогии для каждого правила.
- описание правил и определение зависимостей при помощи современной библиотеки для расширенной работы с естественным языком; для определения зависимостей описанных автором правила были декларированы в модуле DependencyMatcher[10] из библиотеки Spacy[11].
- добавление дополнительного коэффициента multiplier для компенсации перекоса классифицируемых классов совместно с Лапласовым преобразование smoothing. Коэффициент добавляется в зависимости от частоты слова при сопоставлении слова со словарем Байса причинно-следственных предложений.

Основываясь на оригинальном алгоритме и предполагаемых модификациях, приведем структурную схему модели извлечения ПСС. Для удобства разделим схему на два этапа: подготовка данных и непосредственно обучение модели (рис. 1, 2).

Рассмотрим работу со словами-шаблонами и правилами подробней. Всего в оригинальном методе использованы 4 правила для определения зависимостей.

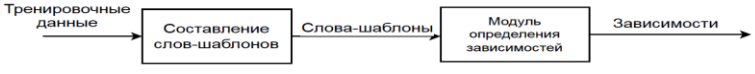

Рис. 1- Подготовка данных для модели извлечения
Fig. 1- Preparing data for the extraction model

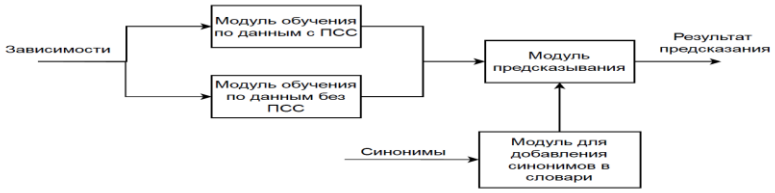

Рис. 2 - Структурная схема работы модели
Fig. 2- Structural diagram of the model operation

Для каждого такого правила у автора оригинального метода представлено небольшое количество слов-шаблонов. Для предлагаемого метода количество слов-шаблонов увеличено в несколько раз исходя из типа самих правил. Опишем правила, формализованные автором, соответствующие этим правилам слова-шаблоны и декларирование правил в модуле DependencyMatcher.

1. Причинные глаголы. Простые одиночные причинные глаголы, имеющие значение причинно-следственного действия (увеличивать, уменьшать, генерировать и т. д.). Согласно правилу, у слова-шаблона в зависимостях должно быть прямое дополнение и внешнее подлежащее [6]:

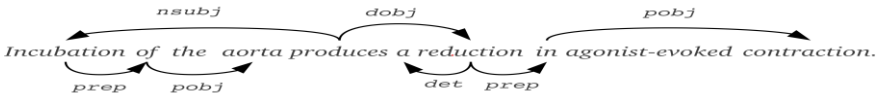

Рис. 3 - Зависимости между словами в предложении: «*Incubation of the aorta produces a specific reduction in agonist-evoked contraction*»
Fig. 3 - Relationships between words in the sentence: "Incubation of the aorta produces a specific reduction in agonist-evoked contraction"

Формализованное автором правило [6]:

$$verb(S,P,V) \wedge nsubj(S,V,C) \rightarrow cause(S,P,C) (3)$$
$$verb(S,P,V) \wedge dobj(S,V,E) \rightarrow effect(S,P,E) (4)$$

Список слов-шаблонов для первого правила выглядит следующим образом:

("cause","caused","causing","generate","generating","generated","triggers","trigger","triggered","make","makes","making","made","force","forces","forcing","require","requires","requiring","required","get","gets","got","getting","gotten","have","has","had","having","help","helping","helps","helped","hinder","hindered","hinders","afflict","afflicts","afflicted","permit","permits","permitted","permitting","let","letting","lets","allow","allowed","allows","enable","enables","enabled").

Шаблон для DependencyMatcher первого правила:

```
pattern_one = [
      {
           "RIGHT_ID": "target",
           "RIGHT_ATTRS": {"ORTH": word}
       },
          {
           "LEFT_ID": "target",
            "REL_OP": ">",
           "RIGHT_ID": "cause",
           "RIGHT_ATTRS": {"DEP":"nsubj"}
          },
          {
           "LEFT_ID": "target",
           "REL_OP": ">",
           "RIGHT_ID": "effect",
           "RIGHT_ATTRS": {"DEP":"dobj"}
          }
   ]
```

Где: «ORTH» – это само слово, «DEP» – это каким членом предложения является слово. Чтобы задать шаблон в DependencyMatcher необходимо выбрать «слово-якорь», то есть по отношению к какому слову будут браться зависимости. Слово-якорь берётся слово из слов шаблонов, RIGHT_ID – это идентификатор слова внутри шаблона, LEFT_ID – это идентификатор родителя, по отношению к которому идёт поиск.

Как видно, у самого первого слова, которое имеет RIGHT_ID «target» нет LEFT_ID, так как это слово является «якорем», от него начинается поиск зависимостей, соответственно у него нет LEFT_ID.

REL_OP – означает какой тип зависимости должен быть между словами. «>» означает, что зависимость должна быть прямой, то есть в древе зависимостей между словами больше не должно быть слов.

2. **Фразовые глаголы.** Словосочетания, состоящие из глагола, за которым следует частица, а также существительные плюс предлог.

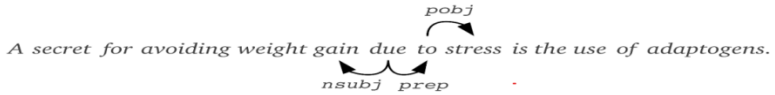

Рис. 4 - Зависимости между словами в предложении: “A secret for avoiding weight gain due to stress is the use of adaptogens”
Fig. 4 - Dependencies between words in a sentence: “A secret for avoiding weight gain due to stress is the use of adaptogens”

Формализованное автором правило выглядит следующим образом:

$$prep\_verb(S, P, due) \wedge prep(S, due, to) \wedge pobj(S, to, C) \rightarrow cause(S, P, C) (5)$$

Подобранный список слов-шаблонов:
("result","resulted","results","cause","caused","causing","lead","leads","leading","led","break","broke","broen","breaks","grow","grew","grown","growing,""act","acted","acts","acting","call","called","calls","calling","handed","hand","hands","handing","think","thinking","thought","one","ones","add","adds","adding","added","be","been","blow","blown","blew","blowing","blows","force","forced").

И частиц к ним:

("in","of","to","up","down","out","off","after","on","by","upon","for","with","around","over","out","away","about","along","into","through","forth").

Выведем шаблон для DependencyMatcher:

```
Pattern_two = [
        {
         "RIGHT_ID": "target",
         "RIGHT_ATTRS": {"ORTH": word}
        },
        {
         "LEFT_ID": "target",
         "REL_OP": ">",
         "RIGHT_ID": "particle",
         "RIGHT_ATTRS": {"DEP":"prep"}
        },
        {
         "LEFT_ID": "target",
         "REL_OP": ">",
         "RIGHT_ID": "cause",
         "RIGHT_ATTRS": {"DEP":"nsubj"}
        },
        {
         "LEFT_ID": "particle",
         "REL_OP": ">",
         "RIGHT_ID": "effect",
         "RIGHT_ATTRS": {"DEP":"pobj"}
        }
       ]
```

3. Пассивные причинные глаголы. Слово-шаблон в зависимостях должно иметь agent (какая-либо сущность, исполняющая действие) и существительное или местоимение, которое является субъектом действия, выполняемого над ним. Формализованное правило:

$$passive(S, P, V) \wedge agent(S, V, C) \rightarrow cause(S, P, C) (6)$$
$$passive(S, P, V) \wedge nsubjpass(S, V, E) \rightarrow effect(S, P, E) (7)$$

Слова-шаблоны:

("caused","generated","triggered","made","forced","required","helped","permited","let","allowed")

Шаблон для DependencyMatcher:

```
    pattern_three = [
  {
   "RIGHT_ID": "target",
   "RIGHT_ATTRS": {"ORTH": word}
  },
  {
   "LEFT_ID": "target",
   "REL_OP": ">",
   "RIGHT_ID": "cause",
   "RIGHT_ATTRS": {"DEP":"agent"}
  },
  {
   "LEFT_ID": "target",
   "REL_OP": ">",
   "RIGHT_ID": "effect",
   "RIGHT_ATTRS": {"DEP":"nsubjpass"}
  },
 ]
```

Обсуждение результатов. Для тестирования модели был взят набор из 655 предложений с причинно-следственной связью и 2389 предложений без причинно-следственной связи. Для возможности сравнения двух методов между собой тренировочные и тестовые данные были взяты из одного и того же источника, что и оригинальный метод. Метрики также были выбраны такие же как у оригинального метода. Основные метрики для проверки работы стали: accuracy, precision и recall. Для расчёта метрик разработанного метода модель прошла 105 циклов теста с различными параметрами multiplier (от 0 до 10 с шагом в 0.5) и smoothing (от 0 до 5):

	Accuracy	Precision	Recall	F-score	Multiplier	Smoothing
0	0.773325	0.093130	0.388535	0.150246	0.0	1.0
1	0.766440	0.134351	0.374468	0.197753	0.0	2.0
2	0.761314	0.149618	0.328859	0.205666	0.0	3.0
3	0.745072	0.164885	0.320475	0.217742	0.0	4.0
4	0.736202	0.183206	0.309278	0.230105	0.0	5.0
...	...	...	...	...	...	...
100	0.781537	0.774046	0.495117	0.603931	10.0	1.0
101	0.770368	0.784733	0.479478	0.595252	10.0	2.0
102	0.754271	0.790840	0.456813	0.580717	10.0	3.0
103	0.748029	0.792366	0.451304	0.575069	10.0	4.0
104	0.738173	0.792366	0.439831	0.565668	10.0	5.0

105 rows × 6 columns

Рис. 5 - Расчет метрик для модифицированного метода извлечения ПСС в среде исполнения Jupyter Notebook

Fig. 5 - Calculation of metrics for the modified PSS extraction method in the Jupyter Notebook runtime environment

Ниже приведены расчеты всех трех метрик и их графиков при различных значениях smoothing и multiplier (рис. 6-20)

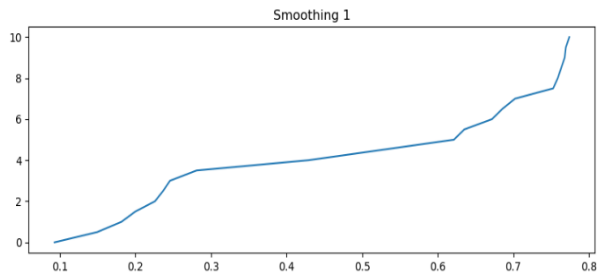

Рис. 6. - График precision для Smoothing 1

Fig. 6. - Precision graph for Smoothing 1

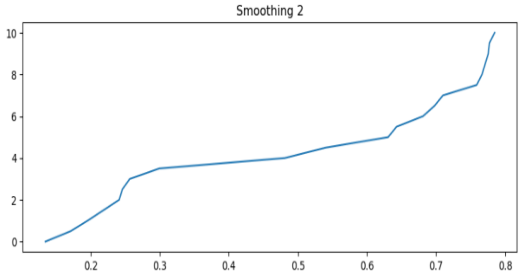

Рис.7- График precision для Smoothing 2

Fig.7- Precision graph for Smoothing 2

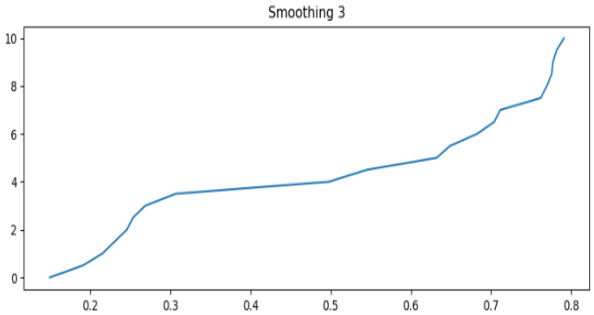

Рис. 8 - График precision для Smoothing 3
Fig. 8 - Precision graph for Smoothing 3

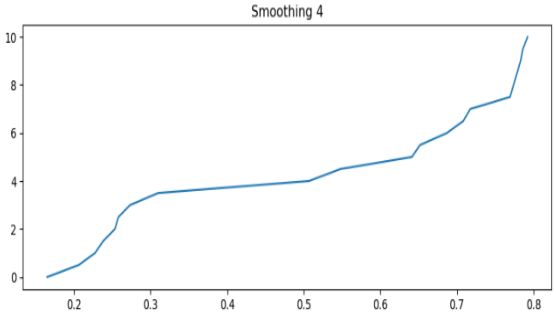

Рис. 9- График precision для Smoothing 4
Fig. 9 -Precision graph for Smoothing 4

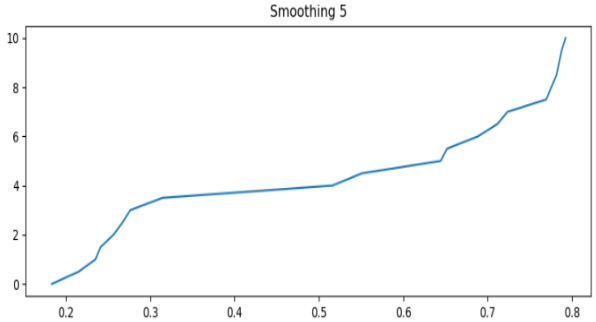

Рис. 10 - График precision для Smoothing 5
Fig. 10 - Precision graph for Smoothing 5

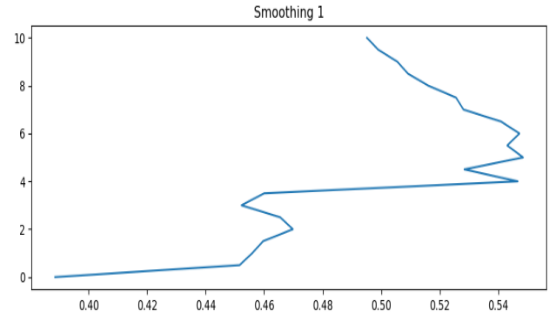

Рис. 11 - График recall для Smoothing 1
Fig. 11 - Recall schedule for Smoothing 1

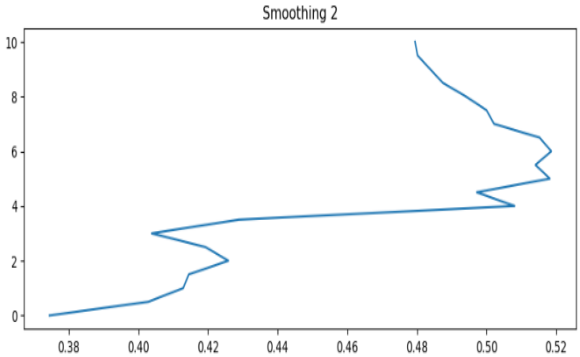

Рис. 12- График recall для Smoothing 2
Fig. 12- Recall chart for Smoothing 2

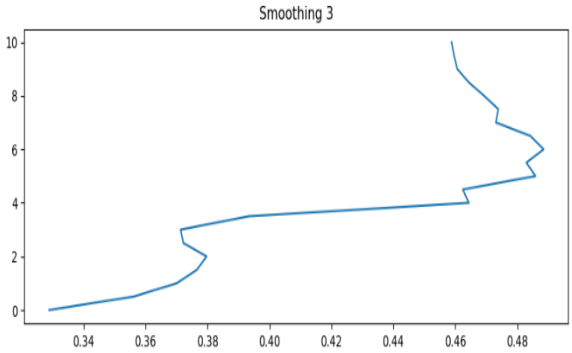

Рис. 13 - График recall для Smoothing 3
Fig. 13 - Recall chart for Smoothing 3

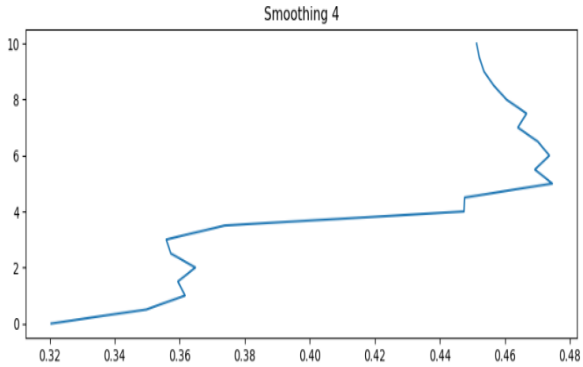

Рис. 14 - График recall для Smoothing 4
Fig. 14 - Recall chart for Smoothing 4

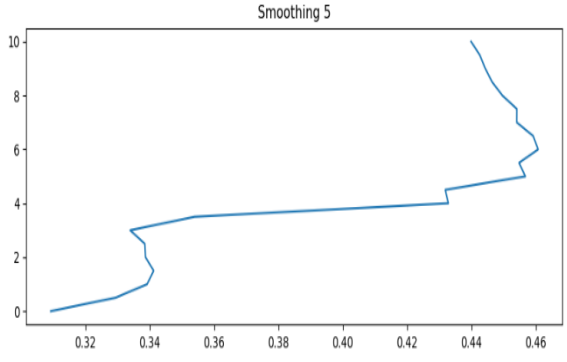

Рис. 15 -График recall для Smoothing 5
Fig. 15 - Recall chart for Smoothing 5

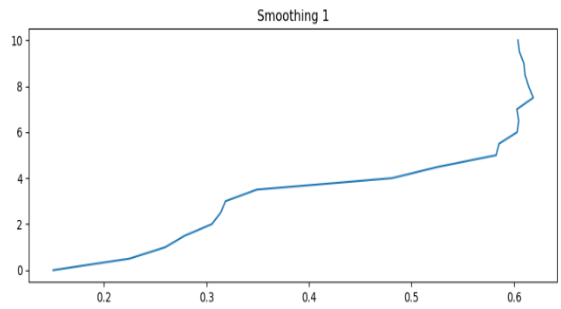

Рис. 16-График F-score для Smoothing 1
Fig. 16-F-score graph for Smoothing 1

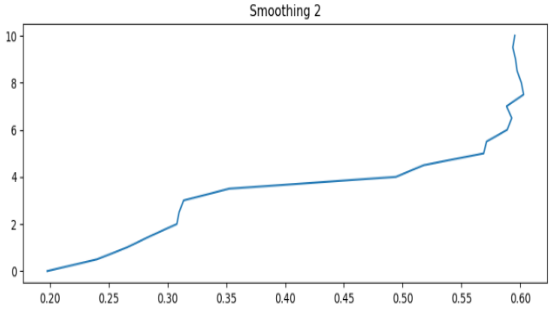

Рис.17-График F-score для Smoothing 2
Fig. 17-F-score graph for Smoothing 2

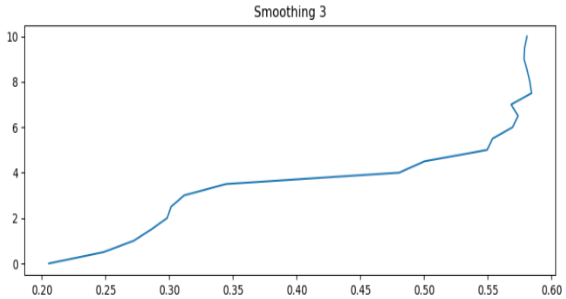

Рис. 18 - График F-score для Smoothing 3
Fig. 18 - F-score chart for Smoothing 3

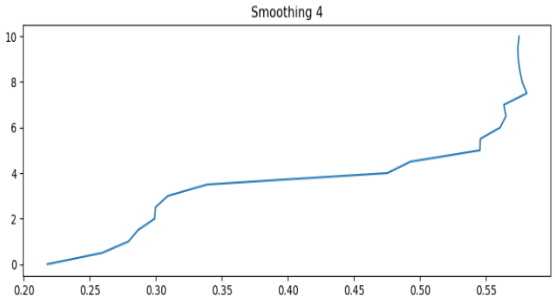

Рис.19- График F-score для Smoothing 4
Fig. 19 - F-score graph for Smoothing 4

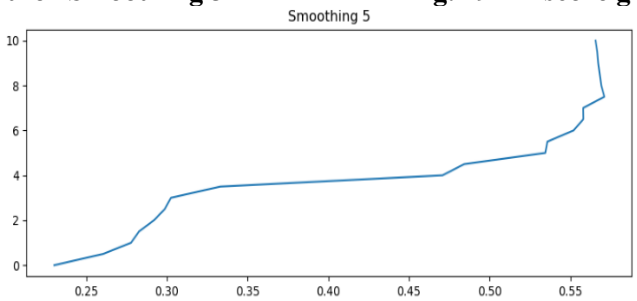

Рис. 20 - График F-score для Smoothing 5
Fig. 20 - F-score chart for Smoothing 5

Достигнутые результаты Антонио Сорренте приведены на скрине из оригинальной статьи:

	Precision	Recall	F-score	α
Global	49%	66%	56%	*no filter*
Per sentence	56%	67%	61%	
Global	55%	65%	59%	0
Per sentence	59%	66%	62%	
Global	**71%**	**58%**	**63%**	0.2
Per sentence	**72%**	**57%**	**64%**	
Global	70%	54%	61%	0.5
Per sentence	70%	53%	60%	
Global	70%	56%	63%	0.7
Per sentence	71%	56%	62%	
Global	71%	54%	62%	1
Per sentence	72%	54%	61%	

Рис. 21 - Скрин таблицы из оригинальной статьи с достигнутыми метриками
Fig. 21 - Screenshot of the table from the original article with achieved metrics

Как видно из рис. 21 максимально высокие результаты по метрикам это:

Precision – 71%; Recall – 58%; F-score – 63%

Наилучшие показатели по метрикам на тренировочных данных были достигнуты при использовании следующих коэффициентов: smoothing – 1, multiplier – 8:

F-score – 0.64%; Precision – 0.76%; Recall – 53%

Видно, что модифицированный метод просел немного по метрике recall. Исходя из того-что precision можно интерпретировать как долю объектов, названных классификатором положительными и при этом действительно являющимися положительными, а recall показывает, какую долю объектов положительного класса из всех объектов положительного класса нашел

алгоритм, в принципе работу классификатора можно считать хорошей. Небольшая просадка по recall не скажется на определении ПСС. Однако модель по другим метрикам превысила оригинальный метод. На тестовых данных достигнуты следующие показатели:

F-score – 0.65%; Precision – 0.76%; Recall – 59%

На тестовых данных метод показал лучшие результаты. Как видно просевшая метрика recall выровнялась. Можно сделать вывод, что метод успешно справляется с поставленной работой. К недостаткам можно отнести большое время обучения модели. Около трех часов на имеющихся тренировочных данных.

По результатам обучения и тестирования можно обозначить следующие шаги для улучшения работы модели в целом:

1. Для улучшения показателей и скорости обучения однозначно необходимо ликвидировать перекос классов в тренировочных данных и тестовых данных с одновременным увеличением самого набора данных. Для этого необходимо с помощью экспертов собрать предложения с наличием ПСС и их отсутствием. При этом синтетические данные не подойдут, так как после обучения система будет давать хорошие результаты на синтетических тестовых данных, а на реальных - метрики резко упадут.
2. Для уменьшения времени обучения уже на большем наборе данных необходимо продумать и внедрить систему накопления данных об обучении на внутренней памяти. Это позволит запускать обучение каждый раз не с 0.
3. Для более тщательного анализа необходимо уменьшить шаг коэффициента multiplier с 1 до 0,1. Возможно имеет смысл и меньше.
4. Разбить основные правила для извлечения на более мелкие.
5. Нужна проверка с использованием других видов сглаживаний, например сглаживание Котта или сглаживание на основе кросс-валидации.

Вывод. В данной статье описан метод извлечения причинно-следственных связей. Метод основан на предложенном Антонио Соргенте.

Метод предполагает комбинированное использование статистических данных и машинных методов. Оригинальный метод был модифицирован переводом работы метода на современные библиотеки, такие как NLTK и Spacy. Правила, сформированные автором, были переработаны и добавлены в модуль Dependency Matcher библиотеки Spacy. Количество ключевых слов по каждому правилу были увеличены.

Метод так же предполагает учет синонимов, подсчет Байесовской статистики и сглаживание Лапласа для нулевых вероятностей. Исходя из разности данных с ПСС и без, был введен коэффициент multiplier для компенсации перекоса классов в данных.

Разработанный метод был протестирован на исходных данных оригинального метода. Описываемый в статье метод показал улучшенные метрики относительно оригинального метода на тренировочных данных, хотя метрика recall была хуже относительно оригинального. Однако как показали тестовые данные описываемый метод по всем метрикам оказался лучше. В статье приведены пути дальнейшего развития метода.

Библиографический список:

1. Штанчаев Х.Б. Нестатистические методы автоматического извлечения причинно-следственных связей из текста / Х.Б. Штанчаев // Известия ЮФУ. Технические науки. – 2023. – № 2(232). – С. 273-280. – DOI 10.18522/2311-3103-2023-2-273-280. – EDN JZUBSO.1
2. Штанчаев, Х.Б. Статистические и машинные методы автоматического извлечения причинно-следственных связей из текста (обзор) / Х.Б. Штанчаев // Известия ЮФУ. Технические науки. – 2023. – № 6(236). – С. 105-114. – DOI 10.18522/2311-3103-2023-6-105-114. – EDN TBHUWW.
3. Fellbaum, Christiane (2005). WordNet and wordnets. In: Brown, Keith et al. (eds.), Encyclopedia of Language and Linguistics, Second Edition, Oxford: Elsevier, 665-670.
4. VerbNet. A Computational Lexical Resourse for Verbs. Интернет-ресурс. Способ доступа - https://verbs.colorado.edu/verbnet/

5. Text Rettieval Conference(TREC). Интернет-ресурс. Способ доступа - https://en.wikipedia.org/wiki/Text_Retrieval_Conference
6. G.V. a. F.M. Antonio Sorgente, "Automatic extraction of cause-effect relations in," Institute of Cybernetics “Eduardo Caianiello” of the National Research Council, Январь 2013.
7. L.A. Dalton, E.R. Dougherty. Optimal Bayesian Classification, SPIE--The International Society for Optical Engineering, 2020, 363p.
8. Sayed, A.H. Inference and Learning from Data: Learning. Cambridge: Cambridge University Press. – 2022, Chapter 55, 2341-2356
9. Ananda P. Noto, Dewi R.S. Saputro; Classification data mining with Laplacian Smoothing on Naïve Bayes method. AIP Conf. Proc. 28 November 2022; 2566 (1): 030004.
10. Dependency Matcher Интернет-ресурс. Способ доступа - https://spacy.io/usage/rule-based-matching#dependencymatcher.
11. SpaCy 101. Все что нам нужно знать. Интернет-ресурс. Способ доступа - https://webdevblog.ru/spacy-101-vse-chto-vam-nuzhno-znat-chast-1
12. Метрики качества моделей бинарной классификации. Интернет-ресурс. Способ доступа - https://loginom.ru/blog/classification-quality.

References:

1. Shtanchaev H.B. Non-statistical methods of automatic extracting causal relationship from text// H.B. Shtanchaev. *Izvestiya YuFU. Tekhnicheskie nauki*. 2023;2(232):273-280. – DOI 10.18522/2311-3103-2023-2-273-280. – EDN JZUBSO.1. (In Russ)
2. Shtanchaev, H.B. Statistical methods of automatic extracting causal relationship from text(review)// H.B. Shtanchaev. *Izvestiya YuFU. Tekhnicheskie nauki.* 2023;6(236):105-114. – DOI 10.18522/2311-3103-2023-6-105-114. – EDN TBHUWW. (In Russ)
3. Fellbaum, Christiane (2005). WordNet and wordnets. In: Brown, Keith et al. (eds.), Encyclopedia of Language and Linguistics, Second Edition, Oxford: Elsevier, 665-670.
4. VerbNet. A Computational Lexical Resourse for Verbs. Internet-resurs. Sposob dostupa - https://verbs.colorado.edu/verbnet/
5. Text Rettieval Conference(TREC). Internet-resurs. Sposob dostupa - https://en.wikipedia.org/wiki/Text_Retrieval_Conference
6. G.V. a. F.M. Antonio Sorgente. Automatic extraction of cause-effect relations in. Institute of Cybernetics “Eduardo Caianiello” of the National Research Council, Yanvar' 2013.
7. L.A. Dalton, E.R. Dougherty. Optimal Bayesian Classification, SPIE--The International Society for Optical Engineering, 2020:363p.
8. Sayed, A.H. Inference and Learning from Data: Learning. Cambridge: Cambridge University Press. – Chapter 55, 2022:2341-2356
9. Ananda P. Noto, Dewi R.S. Saputro; Classification data mining with Laplacian Smoothing on Naïve Bayes method. AIP Conf. Proc. 28 November 2022; 2566 (1): 030004.
10. Dependency Matcher Internet-resurs. Sposob dostupa - https://spacy.io/usage/rule-based-matching#dependencymatcher.
11. SpaCy 101. Vse chto nam nuzhno znat'. Internet-resurs. Sposob dostupa - https://webdevblog.ru/spacy-101-vse-chto-vam-nuzhno-znat-chast-1
12. Metrics of binary classification. Internet.Link - https://loginom.ru/blog/classification-quality (In Russ)

Сведения об авторах:

Штанчаев Хайрутин Баширович, кандидат технических наук, доцент, кафедра программного обеспечения вычислительной техники и автоматизированных систем, старший инженер АСУТП; shtanchaev.h@gmail.com

Мугутдинов Залибек Темирланович, аспирант, кафедра программного обеспечения вычислительной техники и автоматизированных систем; zalik737@mail.ru

Information about authors:

Hairutin B. Shtanchaev, Cand. Sci. (Eng), Assoc. Prof., Department of Software for Computer Engineering and Automated Systems, Senior Engineer of APCS; shtanchaev.h@gmail.com

Zalibek T. Mugutdinov, Postgraduate Student, Department of Software for Computer Engineering and Automated Systems; zalik737@mail.ru

Конфликт интересов/Conflict of interest.

Авторы заявляют об отсутствии конфликта интересов/The authors declare no conflict of interest.

Поступила в редакцию/Received 31.10.2024.

Одобрена после/рецензирования Reviced 27.11.2024.

Принята в печать/Accepted for publication 15.01.2025.