Московский государственный университет имени М.В.Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра алгоритмических языков

Семак Владислав Викторович

Комбинирование методов извлечения научных терминов из текстового документа

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Научный руководитель:
доцент, к.ф.-м.н.
Большакова Елена Игоревна

Москва, 2021

Аннотация

Данная работа посвящена исследованию методов автоматического извлечения терминов из отдельного русскоязычного научно-технического текста. Результаты решения данной задачи могут применяться при построении глоссариев и предметных указателей, при поиске документов и т.д. Рассматриваются методы, основанные на лингвистических шаблонах, статистических мерах и алгоритмах графового ранжирования, а также возможности их комбинирования. Описываются эксперименты по применению комбинаций методов для извлечения терминов и оценивается их эффективность в рамках точности (precision), полноты (recall), F1-меры и средней точности (average precision).

Содержание

1 Введение

В настоящее время появляется всё больше и больше новых быстро развивающихся областей науки и производства. Этому способствует высокая цифровизация общества в целом и науки в частности. У исследователей есть возможность быстро, как никогда ранее, делиться результатами своей работы с коллегами по всему миру. При этом одним из самых распространенных способов передачи информации остаются текстовые данные: книги, публикации в журналах, статьи на специализированных Интернет-ресурсах и множество других.

Текстовый документ, относящийся к конкретной научной области, обычно включает слова и словосочетания, которые обозначают понятия данной области знаний – **термины**, например: *реликтовое излучение, созвездие, чёрная дыра*. Ряд терминов, которые содержатся в отдельном тексте или коллекции, может в дальнейшем использоваться для решения многих задач:

1. термины могут рассматриваться как набор ключевых слов текстовых документов, которые используются при поиске статей[1];

2. термины входят в предметный указатель, составляющегося для облегчения поиска нужной информации в тексте[2];

3. CAT-системы (Computer Assisted Translation) используют наборы терминов для подбора правильного перевода слов[3];

4. наборы терминов определенной предметной области, вместе с отношениями, которыми эти термины связаны между собой, используются при построении графов знаний и онтологий[4].

Активно развивающиеся области науки, как правило, не имеют устоявшейся терминологии, что вынуждает авторов публикаций вводить свои термины. Это введёт к ситуации, когда возможно использование нескольких различных слов или словосочетаний, обозначающих одно понятие. Подобные явления вместе с постоянно увеличивающимся объемом текстовой информации способствуют тому, что человеку сложно уследить за всеми нововведениями в предметной области, а следовательно, сложно вручную определять набор терминов, используемых в тексте.

Решение данных проблем является одной из задач такой области науки, как компьютерная лингвистика. В рамках этой области ведётся разработка методов автоматической обработки текста, в том числе предназначенных для автоматического выделения терминов из отдельного документа или коллекции текстовых документов заданной тематики[5]. При этом выделяется набор **терминов-кандидатов** – слов и словосочетаний, потенциально являющихся терминами, которые далее **ранжируются** (упорядо-

чиваются) так, чтобы как можно больше реальных терминов оказалось в топе ранжированного списка. Известны несколько основных подходов к решению задачи автоматического извлечения терминов[6], использующие:

- лингвистические критерии (например грамматические шаблоны);
- статистические характеристики текста (например частоту встречаемости слова);
- их комбинации (гибридные).

В последнее время начинают появляться статьи, в которых предлагаются использовать различные методы машинного обучения[7, 8, 9, 10] или нейронные сети.

Большинство работ в этой области рассматривают выделение терминов из коллекции документов, однако в ряде задач необходимо извлекать термины из отдельного документа, например в задачах 1–3 вышеприведенного списка. При этом методы, разработанные для текстовой коллекции, при работе с отдельным текстом обычно показывают худшие результаты [11]. Поэтому актуальным является исследование методов для работы с отдельным текстовым документом, тем более что довольно мало работ посвящены этой теме. Кроме того, выбор наиболее подходящего метода извлечение терминов для каждой конкретной предметной области является сложной задачей [12], а работа с отдельным документом лишь усложняет её [11].

В данной работе исследуется возможность комбинирования разнотипных методов для автоматического извлечения однословных и многословных терминов из научно-технических текстов на русском языке. Рассматривается извлечение терминологии из отдельного текста, при условии, что он имеет достаточный объем (не менее $\sim$10 тыс. слов) для применимости статистических критериев. Возможность обработки одного текста, а не коллекции, позволяет применять методы извлечения терминов для терминологического анализа отдельных публикаций или книг по заданной тематике.

Рассматриваемое комбинирование включает извлечение терминов-кандидатов средствами языка лексико-синтаксических шаблонов LSPL[13] с последующей фильтрацией на основе списков стоп-слов и статистики. Для ранжирования терминов-кандидатов исследуется возможность использования комбинации статистической меры терминологичности *C-value*[14], персонализированного PageRank[12] – метода графового ранжирования, изначально используемого в информационном поиске для ранжирования вебстраниц, а также их комбинации (проверяется гипотеза о том, что данная комбинация способна увеличить качество ранжировании). В рамках работы проводились эксперименты на текстах разных тематик и предметных областей. Качество кластеризации оценивалось экспертно. Проведённые эксперименты показали эффективность комбинирования разнотипных методов автоматического выделения терминов.

2 Постановка задачи

Целью работы является исследование возможностей комбинирования известных методов выделения терминов для повышения качества автоматического выявления значимых терминов из отдельного текста научной тематики.

Для достижения поставленной цели требуется решить следующие задачи:

1. провести обзор известных подходов и методов для автоматического выделения терминов и определить возможные способы комбинирования методов;
2. программно реализовать несколько комбинаций рассмотренных методов, включая:
   - шаблоны и правила для выявления терминов;
   - статистические меры;
   - графовые методы.
3. провести экспериментальное исследование их эффективности на текстах научно-технической тематики и выявить наиболее подходящие для решения поставленной задачи.

3 Методы извлечения терминов

3.1 Подходы к задаче извлечения терминов

Большинство методов автоматического извлечения терминов были разработаны для обработки коллекций текстов, методы можно разделить на лингвистические, статистические и гибридные [6].

Лингвистические методы используют лингвистические характеристики для выделения слов и словосочетаний, обозначающих термины предметной области. Для автоматического выделения и фильтрации кандидатов в термины из имеющегося текста они используют словари и **грамматические шаблоны** – образцы, описывающие порядок слов, составляющих искомую языковую конструкцию, и их характеристики, например, условия синтаксического согласования. Так, шаблон `A N1` $\texttt{N2}_{gen}$, описывает последовательности из прилагательного (`A`) и двух существительных (`N1` и `N2`), из которых второе существительное (`N2`) стоит в родительном падеже, в частности: *скалярное произведение векторов, декартово произведение множеств, глобальный экстремум функции и т.д.*. Данный класс методов скорее позволяет получить список фраз, обладающих заданной в шаблоне структурой, чем непосредственно реальных терминов, например, словосочетания *новый метод решения, фиксированное время работы и т.д.* обладают структурой, рассмотренного ранее шаблона, но очевидно не являются терминами какой-либо предметной области. Поэтому в рассматриваемом подходе может происходить фильтрация с помощью набора стоп-слов – выражений, которые априорно не могут являться терминами в рассматриваемой предметной области (например, *автор, метод* и т.д.).

Статистические методы используют различные меры для оценки и ранжирования слов и словосочетаний, с целью получения упорядоченного списка терминов-кандидатов, в начале которого расположено как можно больше кандидатов, являющихся настоящими терминами. Используемые меры можно разделить на те, которые пытаются оценить непосредственно **терминологичность** (termhood) возможного термина – степень его релевантности рассматриваемой предметной области, и меры, для многословных терминов, оценивающие **устойчивость** (unithood) – степень стабильности связи слов в составном термине, т.е. насколько неслучайной является совместная встречаемость слов, составляющих термин-кандидат. В работе [15] показано, что для разных предметных областей наилучшие результаты показывают различные статистические меры, из-за чего сложно на основе статистического подхода создать универсальный метод автоматического выделения терминов. Кроме того, в [15] показано, что взвешенное комбинирование нескольких мер может существенно улучшить итоговое качество, но этот приём требует правильного определения веса для каждой из используемых мер ассоциации, что представляет определенную сложность.

Примеры различных мер представлены в Таблице 1 (названия мер выделены **полу-**

Таблица 1: Меры устойчивости и терминологичности

Тип	**Формула**
На основе частоты	$\boldsymbol{MI} = \log_2 \frac{N * f(a,b)}{f(a) * f(b)}$, где N-количество токенов в коллекции, $f(a)$-частота токена a, $f(a,b)$-совместная частота токенов a и b
	$\boldsymbol{T\text{-}score} = \frac{f(a,b) - \frac{f(a)*f(b)}{N}}{\sqrt{f(a,b)}}$
	$\boldsymbol{C\text{-}value} = \begin{cases} \log(\lvert a\rvert) * f(a), & \{s : a \in s\} = \emptyset \\ \log(\lvert a\rvert) * \left(f(a) - \frac{\sum_{s:a\in s} f(s)}{\lvert s:a\in s\rvert}\right), & \textit{иначе} \end{cases}$, где $\lvert a\rvert$-количество слов в a $\{s : a \in s\}$-множество рассматриваемых словосочетаний, в которые «вложено» словосочетание a
На основе контекста	$\boldsymbol{NC\text{-}value} = 0.8 * C\text{-}value(a) + 0.2 * N\text{-}value(a)$, где $N\text{-}value(\text{a}) = \sum_{b\in C_a} f_a(b) * \frac{r(b)}{n}$ C_a-контекст a, $f_a(b)$-частота встречаемости b в контексте a, Ω-множество достоверных терминов, $\lvert\Omega\rvert = n$, $r(b)$-количество достоверных терминов, в контексте которых встречается b
	$\boldsymbol{DomainCoherence} = \frac{1}{n} \sum_{b\in\Omega} \frac{\log \frac{f_b a}{f(b)*f(a)}}{\log f_b(a)}$
На основе контрастной коллекции	$\boldsymbol{Weirdness} = \frac{\frac{f(a,b)}{N}}{\frac{f_r(a,b)}{N_r}}$, где N_r-количество токенов в контрастной коллекции, $f_r(a,b)$-совместная частота токенов a и b, в контрастной коллекции
	$\boldsymbol{CW} = \log f(a,b) * \left(\log \frac{N + N_r}{f(a,b) + f_r(a,b)}\right)$
	$\boldsymbol{Relevance} = 1 + \log_2 \left(2 + \frac{f(a,b) * df(a,b)}{f_r(a,b)}\right)^{-1}$, где $df(a,b)$-количество документов исследуемой коллекции, в которых встречается пара a, b

жирным шрифтом). Все описанные в таблице меры разделены на три типа согласно тому, на основе чего оценивается терминологичность и/или устойчивость кандидата:

- на основе частоты встречаемости термина-кандидата в тексте;
- на основе контекста, в котором встречается термин-кандидат;
- на основе сравнения частоты встречаемости термина кандидата в исследуемой и контрастной коллекциях.

Меры на основе частоты встречаемости в тексте, такие как меры ассоциации ***MI*** и ***T-score***[6], используются для оценки устойчивости многословных терминов-кандидатов.

Широко используемой является мера терминологичности ***C-value***[14]. Она учитывает количество слов в термине-кандидате ($|a|$), его частоту ($f(a)$), а также термины-кандидаты, в которые вложен данный термин-кандидат a ($\{s : a \in s\}$). Таким образом терминологичность оценивается через учёт частоты термина-кандидата, а его устойчивость через частоту термина-кандидата, как части других кандидатов в термины. Например, для терминов *сходимость функционального ряда* с частотой встречаемости равной 3 и *функциональный ряд*, встречающегося в тексте 6 раз (3 из который как часть термина *сходимость функционального ряда*), оценка первого по *C-value* будет выше, что согласуется с тем, что он является более специфичным термином.

Также существуют меры, которые используют информацию о частоте встречаемости термина-кандидата в различных контекстах. ***NC-value***[14] – одна из таких мер, основанная на *C-value*. Она оценивает терминологичность, учитывая то, насколько много терминов, встречается в контексте данного кандидата. Другой подобной мерой является ***DomainCoherence***[7], которая тоже оценивает терминологичность по количеству терминов в контексте кандидата ($f_b a$ – частота a в контексте b), но опирается на частоту встречаемости термина-кандидата в тексте, а не значение *C-value* для него.

Отдельно можно выделить группу мер, которые оценивают терминологичность термина-кандидата через сравнение частоты его встречаемости в текстах данной предметной области с частотой встречаемости в текстах контрастной тематики. К этой группу можно отнести такие меры как ***Weirdess, CW, Relevance***[16].

Современные методы извлечения терминов комбинируют оба подхода, объединяя в себе их сильные стороны. На первом шаге используется лингвистический подход для получения предварительного списка терминов-кандидатов, а на втором шаге данный список дополнительно фильтруется и ранжируются на основе различных статистик. В настоящее время, большинство работ исследуют именно такие гибридные методы. Общая схема такого подхода состоит из следующих шагов [6]:

1. выделение кандидатов в термины согласно грамматическим образцам, например:

N – углеводы, AN – скалярное произведение, NN – язык программирования, где A – прилагательное, N – существительное

2. фильтрация кандидатов
   - удаление служебных слов: сложных предлогов и т.д.
   - удаление общезначимых слов: *понятие, метод* и т.д.
3. оценка кандидатов по:
   - частоте
   - терминологичности
   - связности / устойчивости (только для многословных кандидатов)
4. ранжирование кандидатов, с учетом сделанных оценок

Лингвистический и статистический подходы к автоматическому извлечению терминов имеют свои недостатки. Трудности, возникающие при подборе наиболее эффективного метода для каждого конкретного случая, послужили появлению работ, предлагающих методы, которые используют машинное обучение и нейронные сети[7, 8]. Данные методы теоретически позволяют объединить преимущества, статистических мер и лингвистических характеристик, используемых в качестве признаков для распознавания терминов, однако обычно требует размеченного корпуса достаточно большого объема для обучения, что значительно уменьшает количество предметных областей, для которых они могут применяться. В частности, для русского языка существует крайне мало корпусов.

Чтобы оценить эффективность какого-либо метода автоматического извлечения терминов, необходимо выяснить, насколько набор выделенных терминов соответствует реальным термина текста. Для получения численной оценки результата в этой задаче обычно используется точность на первых k кандидатах (Precision at K, Precision@K) и средняя точность (AveragePrecision, AP), рассчитываемые по следующим формулам:

$$Precision@K = \frac{\sum_{i=1}^{k} rel(i)}{k} \tag{1}$$

$$AP = \frac{\sum_{k=1}^{n} Precision@k \times rel(k)}{R}, \tag{2}$$

где $rel(i)$ – индикатор, который равен единице, если кандидат под номером i действительно является термином и нулю в противном случае; R – общее количество кандидатов, которые действительно являются терминами. Также может использоваться F1-мера, которая оценивает качество, с которым термины-кандидаты были классифицированы на реальные термины и обычные слова и словосочетания. Обе оценки, через

среднюю точность (AP) и через F1-меру, требуют эталонного набора терминов, содержащихся в тексте. При этом создание подобного эталонного набора требует человека-эксперта, что делает эту задачу довольно сложной и затратной по времени.

3.2 Обзор современных работ

В работах последних лет, посвященных автоматическому извлечению терминов, рассматривается:

- нахождение синтаксических вариантов одного термина и кластеризация терминов, обозначающих схожие понятия;
- определение степени терминологичности слова или словосочетания – выделение терминов с классификацией по тому, насколько они специфичны для предметной области;
- методы фильтрации терминов-кандидатов;
- применение методов машинного обучения для автоматического выделения терминов из текстовых документов;
- графовые методы ранжирования терминов-кандидатов;

Ниже рассматриваются эти работы, а также работы, описывающие выделение ключевых слов и словосочетаний, поскольку данные задачи схожи между собой и методы их решения часто используют общие идеи. Однако извлечение ключевых слов производится из отдельного текстового документа и исследовано значительно лучше, что является важным в настоящей работе.

Распознавание терминологических вариантов и кластеризация терминов

Метод для выделения различных вариантов одного термина предлагается в работе [17], он предусматривает возможность работы с несколькими языками: с русским, с английским, с испанским, с немецким и с французским. На первом шаге используется набор шаблонов UIMA Tokens Regex для выделения однословных и многословных кандидатов в термины. Данные шаблоны учитывают само слово, лемму, псевдооснову и часть речи, что делает их достаточно гибким инструментом, позволяющим выделить самые разнообразные термины. Фильтрация кандидатов также происходит при помощи задания в шаблонах ограничений на наличие или отсутствие предлога в выражении, а также через явное указание слов, которые не могут являться частью кандидата. Ранжирование итогового списка возможных терминов происходит по метрике *Weirdness*. Для

группировки различных синтаксических и морфологических вариантов одного термина применяются специальные правила для сравнения кандидатов. Например, правило:

```
T1 = <существительное1> <предлог1> <существительное2>
T2 = <прилагательное1> <существительное1>
Условие: T1.<существительное1> == T2.<существительное1> И
         основа(T1.<существительное2>) == основа(T2.<прилагательное1>)
```

объединит вместе термины-кандидаты *effect of rotation (эффект вращения)* и *rotational effect (вращательный эффект)*.

В работе [18] рассматривается задача кластеризации терминов на основе морфологических характеристик и семантической информации из тезауруса WordNet. Авторы проводили исследования на коллекции документов из Википедии на польском языке по экономической тематике. В качестве терминов рассматриваются номинативные словосочетания, которые ранжируются по мере *C-value*, после чего словосочетания из конца отранжированного списка отбрасываются. Для кластеризации учитываются несколько различных факторов близости между терминами:

- соседние слова справа и слева;
- части речи слов в некотором окне вокруг термина;
- ближайшие глагол, существительное и предлог слева и справа;
- близость по тезаурусу WordNet, подсчитываемая на основе общих синсетов, к которым принадлежат исходные термины-кандидаты, их гиперонимы и гипонимы;
- количество общих слов в терминах-кандидатах;
- синтаксическая близость, определяемая вхождением кандидатов в одно перечисление (через запятую и/или союз «и»), а также употребляющиеся совместно в конструкциях вида *такой как, например, так же как* и другими.

Указанные факторы оценивались различными статистическими мерами, взвешенная сумма которых использовалась для проведения агломеративная иерархической кластеризации. Для оценки метода применялся набор из 400 терминов, размеченных экспертами. Эксперименты показали значение F1-меры в районе 0.75, а также то, что использование исключительно информации из WordNet или только морфологических и синтаксических признаков ведёт к ухудшению качества кластеризации.

Определение степени терминологичности

В работе [19] рассматривается выделение из текстов кулинарной тематики на немецком языке составных терминов, таких как *meeresfruchte (морепродукты),*

marzipanrohmasse (марципановая паста). Авторы пытаются расширить классификацию терминов-кандидатов и делить термины по их релевантности предметной области. В работе выделяют четыре класса терминологичности:

1. специфичный термин, понятный только экспертам:
   blausud – особый вид варки рыбы путём добавления кислоты;

2. релевантные предметной области термины, понятные не экспертам:
   tomatenpuree – томатная паста;

3. термины из другой предметной области, которые могут иметь косвенное отношение к рассматриваемой: *zimmertemparatur – комнатная температура*;

4. слова, не являющиеся терминами: *Deutschland – Германия*, как часть фразы *gericht aus Deutschland – блюдо из Германии*.

Предлагаемый метод состоит из двух основных этапов. На первом этапе все составные термины разбиваются на составляющие (например *tomatenpuree – томатная паста*, разбивается на *tomaten – томаты* и *puree – пюре*). На втором используется нейросетевая модель для предсказания класса терминологичности для данного кандидата, которая принимает на вход вектор исходного составного термина в дистрибутивном семантическом пространстве представления слов и по нему предсказывает класс терминологичности.

Оценка работы обученной нейросетевой модели производилась на наборе размеченных экспертами терминов. Результаты показывают, что учёт исключительно составного термина позволяет достичь F1-меры порядка 0.72 на скользящем контроле, а добавление в нейросетевую модель признаков, содержащих информацию о компонентах составного термина и их частотах, поднимает качество до 0.8. Представленные эксперименты показывали, что лучше всего классифицируются обычные термины и не термины, в то время как хуже всего модель работает на определении специфичных терминов и терминов из другой предметной области, которые могут иметь косвенное отношение к рассматриваемой. Авторы объясняют такое поведение модели ошибками в разделении составных терминов на составные части, а также сложностью отделения специфичных и обычных терминов.

Методы фильтрации кандидатов в термины

В исследовании [20] рассматривается задача исключения общезначимых словосочетаний из заранее автоматически полученного набора терминов-кандидатов из текстов предметной области на польском языке. Данная задача является важной, поскольку

большинство методов извлечения терминов опирается на различные статистические меры, которые дают достаточно высокую оценку общезначимым фразам из-за их высокой частоты встречаемости в текстах.

Под общезначимыми словосочетаниями понимаются слова и фразы, являющиеся малозначимыми или нерелевантными данной предметной области. Примерами таких фраз могут служить словосочетания: *низкий уровень, точка зрения, сложный вопрос* и другие. Предлагается два метода их выявления, первый состоит в оценке встречаемости термина в текстах различных предметных областей следующими способами:

- непосредственное сравнение частоты встречаемости термина-кандидата в различных предметных областях;
- анализ распределения некоторой статистической меры по различным текстам и выбрасывание кандидатов из концов распределения;
- штраф для тех терминов, которые имели высокое значение данной меры ассоциации в текстах нескольких предметных областей.

Второй метод состоит в анализе контекста рассматриваемых терминов: предполагается, что релевантный термин обычно встречается в небольшом количестве контекстов, тогда как общезначимые термины имеют более разнообразное множество контекстов.

Оценка методов показала, что наибольшее количество общезначимых терминов удаётся обнаружить, введя штраф за большое значение меры ранжирования (в статье использовалось *C-value*) сразу в нескольких различных предметных областях.

Машинное обучение

В работе [7] предлагается метод автоматического выделения терминов из англоязычных текстов на основе обучения с частичным привлечением учителя и использованием различных статистических мер, а также Википедии в качестве дополнительного источника информации. После выделения терминов-кандидатов по грамматическим образцам применяются различные статистические меры для ранжирования. Сначала выбираются лучшие 50-200 терминов по мере *ComboBasic*, использующей информацию о частоте встречаемости термина-кандидата, его вложености в другие термины-канлдидаты и вложености других терминов-кандидатов в него. Данная мера, в отличие от *C-value*, даёт более высокую оценку терминам-кандидатам, являющимися частью других терминов. Затем производится обучение бинарного классификатора для распознавания терминов: выбранные лучшие 50-200 терминов-кандидатов используется в качестве положительных примеров, а остальные кандидаты – в качестве данных с неизвестной меткой. Признаками для классификации выступают значения следующих мер: *C-Value,*

DomainCoherence, Relevance(см. таблицу 1), *LinkProbability, KeyConceptRelatedness*. Последние две меры используют информацию из Википедии для оценки правдоподобности данного кандидата быть термином. Оценка предлагаемого метода производилась на 7 различных наборах англоязычных текстов различных тематик, на 4 из них объединение различных мер с использованием обучения показало улучшение результатов на 1%–2%.

Метод с использованием машинного обучения для автоматического выделения терминов из англоязычных текстов предлагается также в работе [8], в которой исследуется обучение при малом количестве размеченных данных. Предлагаемый метод состоит в итеративном обучении двух нейросетевых моделей с добавлением на каждой итерации примеров, в которых классификаторы более всего уверены, в обучающую выборку. Авторы использовали свёрточные и рекуррентные нейронные сети (LSTM), обосновывая свой выбор желанием учесть как локальную структуру, так и последовательную природу текстовых данных. На вход обе модели принимали векторные представления слов, составляющих кандидат в термины, в дистрибутивной модели word2vec (модель обучалась на тех же текстах, из которых потом извлекались итоговые списки терминов). Сравнение предложенного в [8] метода с ранжированием по C-value и сверточной нейронной сетью, обучаемой на большом объёме размеченных данных, показало, что предложенный метод значительно превосходит C-value (на 16% F1-меры) и лишь слегка уступает обучению с учителем (на 1% F1-меры), но требует намного меньше размеченных примеров (сотни против нескольких десятков тысяч).

3.3 Графовые методы ранжирования

Альтернативным к оценке и ранжированию на основе статистических мер является оценка и ранжирование терминов-кандидатов на основе графов.

В работе [21] предлагается метод ранжирования терминов, основанный на построении графа семантической близости между ними. Сначала известными методами, основанными на частотности слов в коллекции и грамматических образцах, извлекаются и ранжируются термины-кандидаты, из которых имеющие наименьший ранг рассматриваются в дальнейшем как кандидаты в термины. Для оценки их семантической близости используются векторные представления слов и словосочетаний в пространстве дистрибутивной модели семантики word2vec. Для улучшения работы с редко встречаемыми терминами (отсутствующими в модели семантики) используется информация из патентов. Каждая вершина соединяется рёбрами с 10 самыми близкими по косинусной мере вершинами. Вес ребра равен значению косинусной близости соединяемых вершин. После построения графа семантической близости авторы используют алгоритм TextRank[22] для ранжирования кандидатов в термины.

Проверка данного метода проводилась на корпусах англоязычных текстов по тематике информационных технологий и по тематике авиации. Эксперименты показали, что

использование векторных представлений слов для учёта семантической близости позволяет значительно улучшить результаты статистических методов. Кроме того, данный метод на 36% средней точности превосходит TextRank, использующий граф только на основе частоты совместной встречаемости терминов-кандидатов в тексте. Однако использование патентов, как дополнительного источника информации, является спорным, поскольку дало заметный прирост качества извлечения терминов лишь на текстах одной из двух рассматриваемых предметных областей.

В работе [12] для повышения средней точности извлечения терминов предлагается использовать информацию о семантической близости слов для переранжирования списка терминов, полученного каким-либо традиционным (базовым) методом. В качестве базовых методов могут выступать любые методы автоматического извлечения терминов, например, статистические методы, основанные на мерах *C-value, MI, Weirdness* и т.д. Семантическая близость слов оценивается на основе векторного пространства слов модели word2vec и используется для построения графа слов. Рёбрами соединяются только семантически близкие слова, косинусная близость векторов которых выше заданного порога. Для переранжирования используется персонализированный PageRank – модификация PageRank, в которой больший вес получают "важные" слова-вершины, отбираемые экспертом или полученные на основе оценок базового метода извлечения и ранжирования терминов. Это означает, что в результате ранжирования термины-кандидаты, состоящие из слов в "важных" и смежных с ними вершинах, при прочих равных получат большую оценку и окажутся ближе к началу отранжированного списка. Эксперименты показали, что такой подход позволяет значительно (до 10% средней точности) улучшить качество существующих методов извлечения терминов, но наилучший эффект наблюдается тогда, когда в качестве базового использовался не самый эффективный метод.

Таким образом, две рассмотренные работы[21, 12] показывают, что комбинирование графовых методов с другими методами ранжирования является перспективным.

3.4 Извлечение ключевых слов и словосочетаний

Задача выделения ключевых слов и словосочетаний близка к задаче выделения терминов, поскольку и те и другие имеют схожую грамматическую структуру. Ключевые слова описывают основные темы, содержащиеся в некотором тексте или текстовой коллекции, что роднит их с терминами, описывающими понятия предметной области, к которой принадлежит текст. Поэтому разработанные методы выделения ключевых слов и методы выделения терминов имеют много общего. Однако извлечение ключевых слов в большинстве работ производится для отдельного текстового документа небольшого размера, по этой причине традиционные статистические методы показывают худшее качество извлечения (10%–35% F1-меры), и во многих работах последних лет дополни-

тельно исследуются графовые методы[23, 24, 25].

В работе [26] предлагается статистический метод выделения ключевых слов англоязычного документа на основе матрицы совместной встречаемости слов в документе. Для каждого слова вычисляется несколько оценок: его частота в тексте; «степень слова» – количество слов, которые встречаются вместе с рассматриваемым словом (для которых значение в матрице совместной встречаемости отлично от нуля); отношение «степени слова» к его частоте. Итоговая оценка вычисляется как сумма оценок всех входящих в него слов. Экспериментальная проверка данного метода показала, что он позволяет достичь 37.2% F1-меры.

В [25] используется графовый метод выделения ключевых слов и словосочетаний. Сначала все кандидаты разбиваются на несколько групп алгоритмом иерархической кластеризации на основе того, сколько общих слов они имеют. На следующем шаге строится граф, где рёбрами соединены только кандидаты из разных групп, а вес ребра определяется на основе среднего расстояния между вхождениями рассматриваемых кандидатов в текст. Затем кандидаты ранжируются известным алгоритмом TextRank[22]. Эксперименты показали, что разбиение кандидатов на группы позволяет улучшить качество извлечения ключевых слов (до 26% F1-меры).

Метод для извлечения ключевых, использующий множество признаков каждого отдельного слова, рассматривается в статье [27]. Метод определяет, является ли слово ключевым, основываясь на его характеристиках:

– находится ли первая буква слова в верхнем регистре;

– какую позицию слово занимает в тексте;

– частоте встречаемости слова в тексте;

– насколько разнообразен контекст, в котором встречается слово;

– в скольких предложениях встречается слово.

На основе этих характеристик вычисляется оценка важности каждого слова, а по ним – оценка для кандидатов. Авторы сравнивали эффективность своего метода с 11 другими методами извлечения ключевых слов на 20 наборах англоязычных текстов. Эксперименты показали, что предлагаемый метод позволяет достичь значения F1-меры в 50%, и показывает лучшие результаты на 11 наборах текстовых данных из 20 рассмотренных.

В работе [28] предлагается комбинированный метод для выделения ключевых слов из новостного текста. Основной идеей метода является объединение нескольких стандартных статистических и графовых методов с информацией, извлекаемой нейросетевой моделью для генерации заголовков новостных текстов. Для каждого слова-кандидата вычисляется ранг при помощи статистических и графовых методов выделения ключевых слов, после чего эти значения вместе со значениями, взятыми из слоя нейросетевой модели дистрибутивной семантики BERT, обученной для генерации заголовков, подаются на вход алгоритму градиентного бустинга, предсказывающего, явля-

ется ли данный кандидат ключевым словом. Данный метод позволил достичь F1-меры 72% для выделения ключевых слов из новостных текстов. Заметим, что несмотря на высокое качество, этот метод напрямую не применим к задаче извлечения терминов из научного текста, поскольку новостные тексты существенно отличаются от научно-технических, а термины выражают понятия предметной области, а не основные темы текста, на которых строится заголовок.

*****

В представленном обзоре современных работ большинство рассмотренных методов извлечения терминов разрабатывались для работы с текстовыми коллекциями. Поэтому важными являются работы [2, 11], в которых исследуется эффективность таких методов при работе с отдельным текстом.

В работе [11] оценивалось 16 различных методов на 3 коллекциях документов, оценивая эффективность каждого метода на коллекции целиком и на каждом из документов в отдельности. Результаты показывают, что рассматриваемые методы достигают средней точности порядка 23%–65% и F-меры порядка 5%–38% при извлечении терминов из отдельного текста. При этом нет явной связи между оценками эффективности методов при работе с коллекцией и с отдельным текстом. Лучше всего при работе с отдельным текстом оказались *C-value*[14] и *KeyConceptRelatedness*[7], однако ни один из методов не оказался лучшим на всех рассмотренных текстах.

В работе [2] изучалась задача построения предметного указателя, в рамках которой производится извлечение и отбор терминов-кандидатов. Извлечение терминов-кандидатов производится с помощью набора лексико-синтаксических шаблонов языка LSPL[13], для фильтрации кандидатов используются списки стоп-слов, далее происходит отбор на основе факторов значимости терминов-кандидатов в обрабатываемых текстах. Эксперименты на учебно-научных текстах показали, что предлагаемые фильтрация и отбор хорошо справляются с удалением словосочетаний, не являющихся терминами, оставляя около 8% исходно выделенных терминов-кандидатов и позволяя достичь качества извлечения терминов в 70% F-меры.

Поскольку многие рассмотренные в обзоре работы показывают перспективность комбинирования разных методов извлечения терминов, а также действенность графовых методов при ранжировании терминов-кандидатов и при выявлении ключевых слов, целесообразно исследовать комбинации методов для извлечения терминов из отдельного русскоязычного научного текста, опираясь на методы фильтрации работы [2], методы графового ранжирования и ранжирования по мере *C-value*.

4 Комбинируемые методы и эксперименты

4.1 Этапы извлечения терминов

Исследуемые комбинации методов основаны на стратегии извлечения, фильтрации и отбора терминов, описанной в работе [2, 29] и включают следующие этапы:

1. извлечение терминов-кандидатов с использованием:
    - лингвистических шаблонов;
    - отбора по статистики встречаемости в тексте;
    - фильтрации по стоп-словам;
2. ранжирование терминов-кандидатов с использованием:
    - статистической меры терминологичности *C-value*;
    - персонализированного PageRank.

На первом этапе происходит формирование списка терминов-кандидатов, содержащихся в тексте. Сначала из текста выделяется набора терминов-кандидатов согласно заранее составленному списку лексико-синтаксических шаблонов. Для этих целей использовался язык LSPL[1] (Lexico-Syntactic Pattern Language)[13], потому что он позволяет довольно гибко описывать синтаксические структуры и выделять самые разнообразные термины-кандидаты. После чего происходит фильтрация полученного набора по стоп-словам и отбор терминов-кандидатов на основе частоты их встречаемости в тексте и прочих факторов.

На втором этапе происходит ранжирование полученного списка терминов-кандидатов с целью их упорядочивания согласно релевантности предметной области. Рассматривались три способа ранжирования терминов-кандидатов: мера терминологичности *C-value*, метод графового ранжирования – персонализированный PageRank и их комбинация. Результатом решения задачи автоматического извлечения терминов считаются первые 90% терминов-кандидатов (но не более 200) отранжированного списка. Граница отсечения для отранжированного списка подбиралась экспериментально.

4.2 Извлечение терминов-кандидатов по шаблонам

Использовалось три группы лингвистических шаблонов и правил извлечения:

- выделения авторских терминов;
- выделения терминов-кандидатов по грамматическим образцам;

[1] http://lspl.ru

- выделения синонимов.

Шаблоны для авторских терминов нацелены на выделение терминов-кандидатов, определение которых содержится в обрабатываемом тексте. Примером может служить поиск в тексте фразы вида *Введём понятие <термин-кандидат>, под которым будем понимать ...*, которая выделяется следующим шаблоном:

```
TrustedAuth = "введем" "понятие" Term1 "," "под" "которым" "будем" "понимать"
```

При нахождении фрагмента текста, удовлетворяющего данному шаблону, выделяется слово или словосочетание, стоящее на месте *Term1*, и возвращается его нормальная форма (в случае словосочетания отдельно нормализуется каждое слово).

Шаблоны данной группы делятся на достоверные и недостоверные, что фиксируется в имени шаблона (Thrusted / Untrusted). Разделение происходит по тому, насколько высока точность работы конкретного лингвистического шаблона, менее достоверные шаблоны имеют более высокую вероятность выявить термин-кандидат, который в действительности не является терминов предметной области. Примером недостоверного шаблона может служить поиск выражения вида *<термин-кандидат> – это ...*. Очевидно, что точность этого шаблона ниже, чем у ранее приведённого примера, поэтому для включения в результат кандидатов, которые были выделены подобными шаблонами, следует использовать более строгие критерии.

Следующей группой шаблонов являются шаблоны для выделения грамматических образцов – существительных и номинативных групп, которые могут являться терминами. Язык LSPL позволяет указывать желаемые морфологические характеристики извлекаемых кандидатов: падеж, число, род, согласованность прилагательного с существительным и многое другое. Например:

- любое существительное:
  N – *матрица, функция, интеграл и т.д.*

- прилагательное и существительное, согласованные в роде, числе и падеже:
  A N <A=N> – *ортогональная матрицы, монотонная функция и т.д.*

- пара существительных, второе из которых стоит в родительном падеже:
  N1 N2<c=gen> – *определитель матрицы, производная функции и т.д.*

Эта группа шаблонов выделяет самое большое количество терминов-кандидатов, но очевидно обладает довольно низкой точностью, поскольку, например, выделяет все существительные, из которых лишь малая часть является реальными терминами. Несмотря на это, именно термины-кандидаты, полученные из этих шаблонов, позволяют найти самое большое количество терминов, поскольку далеко не все из них явно определяются в тексте и могут быть выделены группой шаблонов для авторских терминов.

Третий набор шаблонов служит для выделения синонимов. Данная группа позволяет расширить множество извлечённых терминов-кандидатов путём поиска и добавления слов и словосочетаний, которые описаны в тексте как синонимичные. Например при нахождении фрагмента текста удовлетворяющего шаблону:

```
Term "будем" "также" "называть" TermSyn
```

выделяется термин-кандидат *Term* и его синоним *TermSyn*, что помогает связать эти два термина-кандидата и учитывать их связь при дальнейшей фильтрации и отборе, что может быть важно в случае, если у одного из синонимов низкая частота встречаемости в исследуемом тексте, которая не позволяет сделать о нём верные статистические выводы.

Дополнительные примеры шаблонов всех видов приведены в приложении А.

В работе [2] проводилось исследование точности данных шаблонов. Результаты показали, что шаблоны авторских терминов имеют самую высокую точность от 90% до 95%, что объясняется использованием в данной группе специальных слов-маркеров, которые указывают на термины. Самую низкую точность показали лингвистические шаблоны второй группы, нацеленные на выделение грамматических образцов. Эксперименты показали, что их точность составляет 8%-10%, что объясняется тем, что они плохо отражают терминологичность и выделяют самые различные словосочетаний (например, *вариант, предлагаемый метод, задача интегрирования*). Набор шаблонов для извлечения синонимов показал точность около 65%, что объясняется опорой этих правил на определенные слова-маркеры и синтаксические структуры, которые не так сильно распространены в языке и более характерны для фраз, где употребляются термины.

4.3 Фильтрация терминов-кандидатов

В результате этапа извлечения терминов-кандидатов по шаблонам было извлечено три множества терминов-кандидатов:

- T_{auth} – выделенные шаблонами авторских терминов;
- T_{gram} – извлеченные шаблонами грамматических образцов;
- T_{syn} – полученные применением шаблонов для поиска синонимов.

Далее происходит фильтрация и отбор полученных терминов-кандидатов, формирующих итоговое множество терминов-кандидатов для ранжирования.

Сначала происходит фильтрация на основе заданных списков стоп-слов, которые были расширены в ходе данной работы. Списки включают общезначимые слова и словосочетания. Первый список содержит слова, которые сами по себе не могут являться терминами, такие как *данные, задача, метод* и другие. Второй список стоп-слов содержит слова, которые не могут быть частью термина, например *возможный, заданный,*

начальный. Из каждого множества T_{auth}, T_{gram} и T_{syn} удаляются все кандидаты, которые:

- входят в первый список;
- состоят исключительно из слов первого списка;
- содержат слова из второго списка.

На этом шаге происходит фильтрация большинства общезначимых фраз и служебных конструкций, которые часто встречаются в тексте, но при этом не могут являться терминами.

Поскольку в текстах научно-технической тематики содержится достаточно много буквенных обозначений, формул и примеров программ, на этом же шаге удаляются все термины-кандидаты, в которых нет ни одной буквы русского алфавита или которые состоят менее чем из 4 символов.

После этого итеративно строится итоговое множество терминов-кандидатов T. Изначально T считается пустым и на каждой итерации в него добавляется некоторая часть терминов-кандидатов из множеств T_{auth}, T_{gram} и T_{syn}. В процессе отбора терминов-кандидатов учитываются следующие факторы[2, 29]:

- достоверность шаблона, которым был извлечен термин-кандидат: кандидаты, извлеченные наиболее достоверными шаблонами, добавляются в T в первую очередь;
- частота встречаемости термина-кандидата: согласно закону Ципфа значимые термины должны принадлежать центральной части распределения кандидатов по частоте встречаемости в тексте (они не могут быть как слишком частотными, так и слишком редкими);
- лексическая близость терминов одной тематики: имеет смысл добавлять в T термины-кандидаты, имеющие общие слова (хотя бы одно) с уже отобранными терминами-кандидатами.

Самую высокую точность показывает набор лингвистических шаблонов для извлечения авторских терминов, поэтому логично взять именно термины-кандидаты из этого набора за основу, которая затем будет дополняться. Процесс построения итогового множества T, состоит из следующих шагов:

1. Вычисляются $p_1 = 40$ и $p_2 = 95$ перцентили частоты встречаемости в тексте для терминов-кандидатов из множества T_{auth}. Данные значения будут использоваться в качестве порогов, по которым будет происходить отсечение кандидатов на следующих шагах.

2. В T добавляются все термины-кандидаты из множества T_{auth}, которые были выделены достоверными шаблонами авторских терминов (помеченных как Trusted) и чья частота находится в диапазоне между перцентилями, вычисленными на предыдущем шаге.

3. В T добавляются термины-кандидаты из множества T_{gram}. Для добавления частота термина-кандидата должна принадлежать диапазону из первого правила и он должен иметь общие слова (хотя бы одно) с одним из терминов-кандидатов из множества T_{auth}. Например, если в T_{auth} есть термин-кандидат *и/или-граф*, то на этом шаге в T будет добавлен термин-кандидат *вершина графа*.

4. Оставшиеся термины-кандидаты из множества T_{auth}, имеющие общие слова с каким-либо терминов-кандидатов из T, также добавляются в T.

5. В T добавляются термины-кандидаты из оставшегося множества T_{auth}, являющиеся синонимами к какому-либо элементу T.

6. Термины-кандидаты из T_{gram}, являющиеся синонимом к терминам-кандидатом в T, также добавляются в T.

7. Все пары синонимов, чья суммарная частота встречаемости в тексте лежит между перцентилями из первого правила, добавляются в множество T.

8. В T добавляются термины-кандидаты из множество T_{gram}, если их частота принадлежит диапазону из первого правила и они имеют хотя бы одно общее слово с каким-либо из уже отобранных терминов-кандидатов (элементов T).

Значения перцентилей p_1 и p_2 из первого шага подбирались эмпирически.

4.4 Ранжирование терминов-кандидатов

C-value

В качестве одного из вариантов упорядочивания по релевантности терминов-кандидатов предлагается вычислять значение меры терминологичности *C-value*, которая была выбрана, поскольку показала хорошие результаты во многих работах, посвященных автоматическому извлечению терминов[15, 14, 18, 20], в том числе при работе с отдельным текстовым документом[11]. В данной работе используется модификация *C-value*, вычисляемая по формуле:

$$\textit{C-value(a)} = \begin{cases} \log\left(|a| + 0.01\right) * f(a), & \{s : a \in s\} = \emptyset \\ \log\left(|a| + 0.01\right) * \left(f(a) - \frac{\sum_{s:a \in s} f(s)}{|s:a \in s|}\right), & \textit{иначе} \end{cases} \tag{3}$$

Изначально данная мера создавалась исключительно для использования при обработке словосочетаний (см. таблицу 1), для возможности обработки однословных кандидатов в данной работе к длине кандидата в словах ($|a|$) добавляется небольшая аддитивная составляющая 0.01.

Персонализированный PageRank

PageRank оценивает важность каждой вершины графа на основе весов входящих в неё рёбер и вершин, из которых эти рёбра исходят. Чем больше вершин графа имеют ребро, идущее в оцениваемую вершину, тем больший вес она получит. При этом возникает проблема изолированных вершин, которые не связаны ребрами с другими вершинами графа и в связи с этим получают нулевую оценку. Для решения данной проблемы используется приём, называемый **телепортацией** – считается, что их каждой вершины графа кроме обычного перехода по выходящему из неё ребру можно «телепортироваться» в произвольную вершину с некоторой вероятностью.

Итерационная формула вычисления оценок вершин графа для *PageRank* с использованием телепортации записывается так:

$$Pr = cMPr + (1 - c)v, \tag{4}$$

где Pr – вектор оценок вершин, M – матрица смежности графа, c – вероятность перехода по ребру, v – вектор вероятностей телепортации в вершины графа. Все начальные значения оценок в векторе Pr равны нулю, вычисление останавливается, когда изменения вектора Pr становятся достаточно малыми.

В данной работе предлагается использовать **персонализированный PageRank**[12] – модификацию *PageRank* с телепортацией, в которой вероятность телепортации в вершины не одинакова, а предпочтение отдаётся некоторому набору достоверных вершин. В качестве достоверных берутся 10% терминов-кандидатов с самым большим значением *C-value*. В экспериментах в качестве достоверных рассматривалось также использование терминов-кандидатов, выделенных шаблонами авторских терминов, однако этот вариант показал более плохое качество ранжирования в связи с малым числом подобных терминов-кандидатов.

Для применения персонализированного *PageRank* строится взвешенный граф с терминами-кандидатами в вершинах. Вес ребра определяется частотой совместной встречаемости соединяемых им терминов-кандидатов, подсчитанной по тексту с учётом контекстного окна в 14 слов. Вектор v в формуле (4) определяется следующим образом:

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

где t_i – i-ый термин-кандидат, S – множество достоверных терминов-кандидатов. Пример фрагмента построенного графа терминов-кандидатов представлен на рисунке 1.

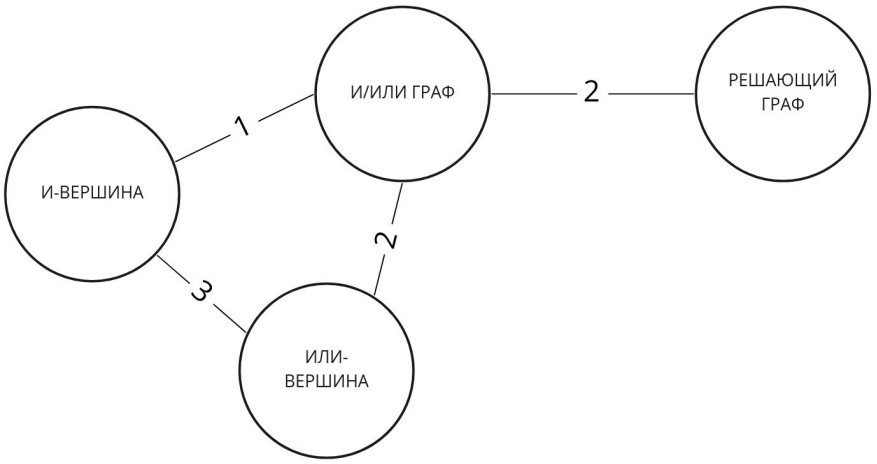


Рис. 1: Фрагмент графа терминов-кандидатов для расчёта *PageRank*

Комбинирование C-value и PageRank

При использовании комбинации меры терминологичности *C-value* и персонализированного *PageRank*, проверяется гипотеза о том, что данная комбинация способна увеличить качество извлечения терминов из отдельного текста. Комбинированная оценка термина-кандидата вычисляется по формуле, аналогичной в работе [12]:

$$score = n\_c - value \times (1 + n\_pagerank) \tag{5}$$

где n_c-calue, n_pagerank – нормализованные значения *C-value* и *персонализированного PageRank* соответственно. Нормализация производится так, чтобы максимальное значение оценки *score* термина-кандидата равнялось 1. Данная формула не симметрична отнотельно значений двух используемых мер и отдаёт большее предпочтение мере *C-value*, поскольку её эффективность показана во множестве работ[11, 15, 14].

4.5 Результаты экспериментов

Для проведения экспериментов было реализовано несколько программных модулей на языке Python3.7[2]:

- **lspl.py** – модуль для работы с утилитой языка LSPL, позволяющий обращаться к ней непосредственно средствами языка Python3.7. Для обработки результирующего xml-файла использовался стандартный модуль xml.etree.ElementTree из библиотеки языка.

[2]https://docs.python.org/3.7/

- **metrics.py** – модуль с набором функций, реализующих вычисления статистических мер, а также построение графа для алгоритма PageRank, использовалась реализация персонализованного PageRank из модуля fast_pagerank[3].

- **ate.py** – основной модуль, реализующий этапы извлечения терминов.

Последовательность шагов выполнения Python-программы:

1. На вход программе подаётся обрабатываемый файл с учебно-научным текстом в формате .txt или .pdf, а также файлы с шаблонами языка LSPL. Файл в формате .pdf преобразуется в формат .txt. Для работы с pdf-файлами использовался модуль textract[4].

2. Производится предобработка текста: в нём удаляются спецсимволы, остаются только буквы кириллицы и латиницы, а также цифры и знаки пунктуации.

3. Входной текст вместе с файлами шаблонов подаётся на вход утилите языка LSPL[5], которая производит поиск в тексте терминов-кандидатов по заданным шаблонам.

4. Из выходного xml-файла, полученного на предыдущем шаге, извлекаются выделенные термины-кандидаты, информация о шаблонах, которыми они были извлечены, и их позиции во входном тексте.

5. Производится фильтрация и отбор терминов-кандидатов множетсва T согласно этапу 1 извлечения терминов (см. 4.3).

6. Для каждого термина-кандидата подсчитывается его оценка по *C-value*.

7. Происходит построение графа терминов-кандидатов и вычисляются их оценки по алгоритму персонализированного *PageRank*, а также комбинации оценок *C-value* и *PageRank* по формуле (5).

8. Первые 90% терминов-кандидатов (но не более 200) отранжированного списка считаются терминами и записываются вместе со своими оценками в выходной файл в формате .csv.

Экспериментальная проверка рассматриваемых комбинаций методов проводилась на текстах семи русскоязычных учебно-научных текстов по темам: формальные грамматики (ФГ), дифференциальные уравнения (ДУ), дискретная математика (ДМ), искусственный интеллект (ИИ), язык Лисп (ЯЛ), системы программирования (СП) и математический анализ (МА). Каждый из текстов содержал главы, которые могли отличатся по тематике.

[3]https://pypi.org/project/fast-pagerank/
[4]https://textract.readthedocs.io/en/stable/
[5]https://github.com/cmc-msu-ai/lspl

Из каждого текста был выделен список терминов, для которых вручную были проставлены метки: термин, не термин. Полученная разметка использовалась для оценки качества извлечения терминов. Характеристики каждого текста приведены в таблице 2: число слов в текстовом документе, число извлеченных по шаблонам терминов-кандидатов, число глав в документе, а также итоговое число извлеченных терминов.

Таблица 2: Характеристики исследуемых текстов

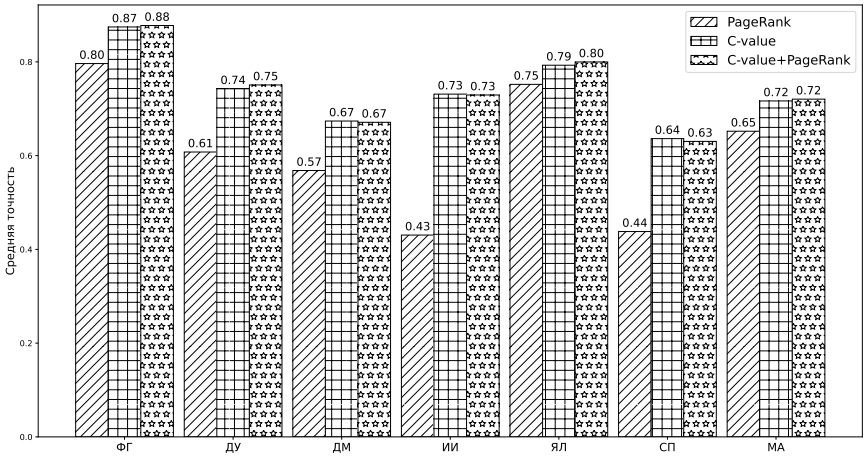
Текст	Число слов	Число кандидатов	Число глав	Извлечено терминов
ФГ	12127	9935	2	53
ДУ	14096	11044	3	80
ДМ	23770	17793	6	198
ИИ	26339	19858	4	168
ЯЛ	27281	21882	4	143
СП	43294	34374	6	200
МА	55306	46014	24	200
Среднее	28888	22986	7	149

Для оценки качества извлечения терминов использовалась средняя точность (AveragePrecision, AP), вычисляемая по формуле (2). Каждый текст обрабатывался в двух вариантах: целиком и по главам. При обработке по главам сначала извлекались термины из каждой главы, а потом полученные множества сливались и оценивались. В таблице 3 приведены значения средней точности для каждого из вариантов ранжирования (*персонализированный PageRank, C-value* и их комбинация) и обработки текста (целиком или по главам). Также результаты экспериментов для обработки текстов целиком и по главам представлены на рисунках 2(a) и 2(b) соответственно.

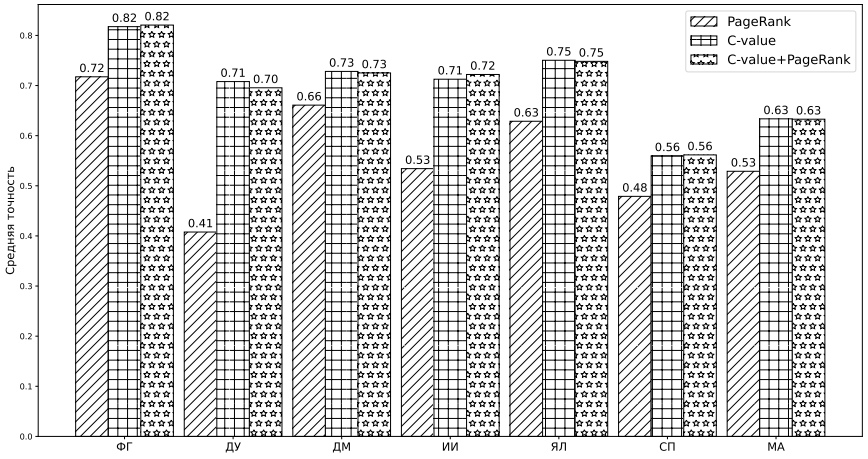
Таблица 3: Средняя точность извлечения терминов

метод / текст	PageRank		C-value		C-value+PageRank	
	целиком	по главам	целиком	по главам	целиком	по главам
ФГ	0.80	0.72	0.87	0.82	**0.88**	0.82
ДУ	0.61	0.41	0.74	0.71	**0.75**	0.70
ДМ	0.57	0.66	0.67	**0.73**	0.67	**0.73**
ИИ	0.43	0.53	**0.73**	0.71	0.72	0.72
ЯЛ	0.75	0.63	0.79	0.75	**0.80**	0.75
СП	0.44	0.48	**0.64**	0.56	0.63	0.56
МА	0.65	0.53	0.71	0.63	**0.72**	0.63
Среднее	0.61	0.57	0.73	0.70	**0.74**	0.70

Результаты проведённых экспериментов показывают, что из рассматриваемых способов ранжирования лучшего качества на четырёх текстах (ФГ, ДУ, ЯЛ, МА) достигает комбинация *C-value* и *PageRank*, на трёх остальных текстах (ДМ, ИИ, СП) качество



(a) Обработка текстов целиком



(b) Обработка текстов по главам

Рис. 2: Средняя точность извлечения терминов

не уступает (ДМ) или незначительно уступает (ИИ, СП) средней точности, полученной с использование *C-value* (падение менее 1%). Данный результат говорит о том, что ранжирование терминов-кандидатов с использованием комбинации *C-value* и *персонализированного PageRank* и с использованием одного *C-value* могут использоваться практически равноценно. *PageRank* (вне комбинации) показал самое плохое качество ранжирования из трёх рассмотренных способов, позволив получить сравнительно хорошие результаты лишь на нескольких текстах (ФГ, ДМ, МА), что, скорее всего, вызвано высокой плотностью терминов в них.

Обработка текстового документа по главам позволила достичь лучшего качества только на одном учебно-научном тексте – по дискретной математике. На наш взгляд, это произошло по причине большого тематического разнообразия глав в этом тексте, что может быть учтено при выборе способа обработки научно-технических текстов для извлечения из них терминов.

Дополнительно было проведено исследование точности, полноты и F1-меры извле-

чения терминов, оценка производилась для первых 50, 100, 150 терминов-кандидатов и для всего извлечённого набора (после шага 8 обработки текста). Результаты представлены в таблицах 4, 5, 6 и 7 соответственно. На рисунках 3(a) и 3(b) показаны диаграммы значений F1-меры для обработки текстов целиком и по главам, а диаграммы для точности и полноты приведены в приложении Б. Полученные оценки точности, полноты и F1-меры в целом подтверждают результаты, сделанные по оценке средней точности. Заметим, что результаты всех рассматриваемых в таблицах 4 – 7 мер практически одинаковы для *C-value* и комбинации *C-value* и *PageRank* (а для средней точности они отличаются), что объясняется тем, что эти меры не учитывают различия в упорядочивании терминов.

Таблица 4: Точность (P), полнота (R) и F1-мера для топ-50 терминов-кандидатов

метод / текст	PageRank						C-value						C-value+PageRank					
	целиком			по главам			целиком			по главам			целиком			по главам		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
ФГ	0.70	0.51	0.59	0.66	0.48	0.56	**0.72**	**0.52**	**0.61**	0.66	0.48	0.56	**0.72**	**0.52**	**0.61**	0.66	0.48	0.56
ДУ	0.46	0.52	0.49	0.40	0.46	0.43	**0.58**	**0.66**	**0.62**	0.48	0.55	0.51	**0.58**	**0.66**	**0.62**	0.48	0.55	0.51
ДМ	0.58	0.18	0.27	0.66	0.20	0.31	0.64	0.20	0.30	**0.80**	**0.25**	**0.38**	0.64	0.20	0.30	0.78	0.24	0.37
ИИ	0.38	0.18	0.24	0.50	0.24	0.32	**0.74**	**0.35**	**0.47**	0.72	0.34	0.46	0.72	0.34	0.46	0.72	0.34	0.46
ЯЛ	0.70	0.33	0.45	0.54	0.26	0.35	**0.78**	**0.37**	**0.50**	0.70	0.33	0.45	**0.78**	**0.37**	**0.50**	0.70	0.33	0.45
СП	0.50	0.09	0.15	0.46	0.08	0.13	**0.66**	**0.11**	**0.19**	0.54	0.09	0.16	**0.66**	**0.11**	**0.19**	0.54	0.09	0.16
МА	0.68	0.09	0.17	0.56	0.08	0.14	**0.80**	**0.11**	**0.20**	0.64	0.09	0.16	**0.80**	**0.11**	**0.20**	0.68	0.09	0.17
Среднее	0,57	0.27	0.34	0.54	0.26	0.32	**0.70**	**0.33**	**0.41**	0.65	0.30	0.38	**0.70**	**0.33**	**0.41**	0.65	0.30	0.38

Таблица 5: Точность (P), полнота (R) и F1-мера для топ-100 терминов-кандидатов

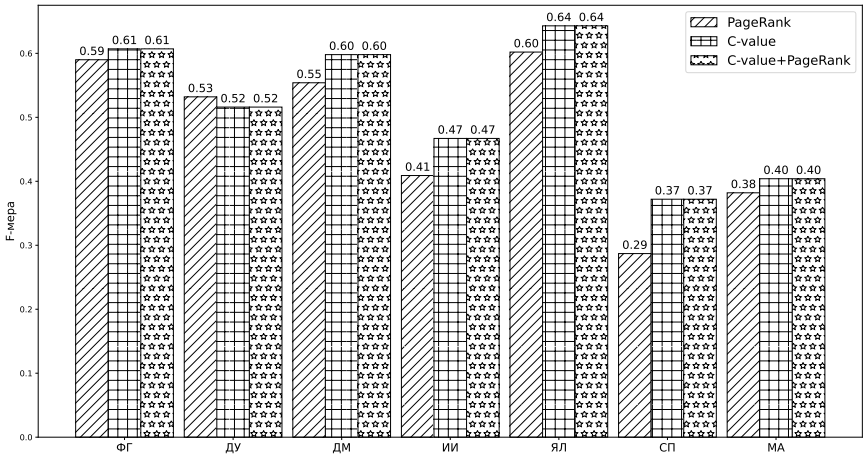
метод / текст	PageRank						C-value						C-value+PageRank					
	целиком			по главам			целиком			по главам			целиком			по главам		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
ФГ	**0.39**	**0.57**	**0.46**	0.37	0.54	0.44	**0.39**	**0.57**	**0.46**	0.37	0.54	0.44	**0.39**	**0.57**	**0.46**	0.37	0.54	0.44
ДУ	**0.33**	**0.75**	**0.46**	0.28	0.64	0.39	**0.33**	**0.75**	**0.46**	0.26	0.59	0.36	**0.33**	**0.75**	**0.46**	0.27	0.61	0.38
ДМ	0.57	0.35	0.43	0.58	0.36	0.44	0.63	0.39	0.48	**0.67**	**0.41**	**0.51**	0.63	0.39	0.48	**0.67**	**0.41**	**0.51**
ИИ	0.32	0.30	0.31	0.42	0.40	0.40	**0.54**	**0.51**	**0.52**	**0.54**	**0.51**	**0.52**	**0.54**	**0.51**	**0.52**	**0.54**	**0.51**	**0.52**
ЯЛ	0.61	0.58	0.59	0.48	0.45	0.47	**0.66**	**0.62**	**0.64**	0.57	0.54	0.55	**0.66**	**0.62**	**0.64**	0.58	0.55	0.56
СП	0.47	0.16	0.24	0.45	0.15	0.23	**0.58**	**0.20**	**0.29**	0.51	0.17	0.26	0.57	0.19	0.29	0.51	0.17	0.26
МА	0.65	0.18	0.28	0.49	0.14	0.21	**0.66**	**0.18**	**0.29**	0.63	0.18	0.27	**0.66**	**0.18**	**0.29**	0.63	0.18	0.27
Среднее	0.48	0.41	0.40	0.44	0.38	0.37	**0.54**	**0.46**	**0.45**	0.51	0.42	0.42	**0.54**	**0.46**	**0.45**	0.51	0.42	0.42

Таблица 6: Точность (P), полнота (R) и F1-мера для топ-150 терминов-кандидатов

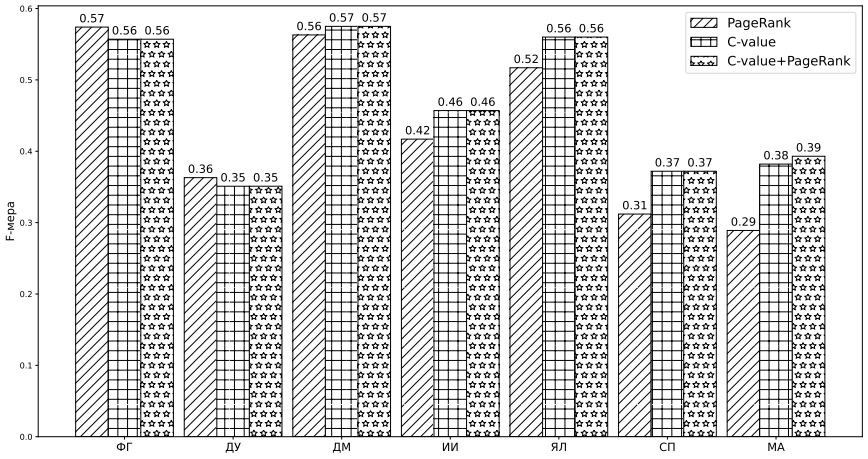
метод / текст	PageRank						C-value						C-value+PageRank					
	йеликом			по главам			целиком			по главам			целиком			по главам		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
ФГ	**0.26**	**0.57**	**0.36**	0.25	0.54	0.34	**0.26**	**0.57**	**0.36**	0.25	0.54	0.33	**0.26**	**0.57**	**0.36**	0.25	0.54	0.34
ДУ	**0.22**	**0.75**	**0.34**	0.21	0.71	0.32	**0.22**	**0.75**	**0.34**	0.21	0.71	0.32	**0.22**	**0.75**	**0.34**	0.21	0.71	0.32
ДМ	0.53	0.49	0.51	0.57	0.52	0.54	**0.61**	**0.56**	**0.58**	0.60	0.55	0.58	**0.61**	**0.56**	**0.58**	0.60	0.55	0.58
ИИ	0.33	0.47	0.39	0.35	0.50	0.41	**0.42**	**0.59**	**0.49**	0.39	0.55	0.45	**0.42**	**0.59**	**0.49**	0.39	0.55	0.45
ЯЛ	0.51	0.73	0.60	0.45	0.63	0.52	**0.54**	**0.76**	**0.63**	0.45	0.63	0.52	**0.54**	**0.76**	**0.63**	0.45	0.63	0.52
СП	0.37	0.19	0.25	0.39	0.20	0.27	**0.51**	**0.26**	**0.35**	0.48	0.25	0.32	0.51	0.26	0.34	0.47	0.24	0.32
МА	0.57	0.24	0.33	0.43	0.18	0.25	**0.61**	**0.25**	**0.36**	0.55	0.23	0.33	**0.61**	**0.25**	**0.36**	0.55	0.23	0.33
Среднее	0.40	0.49	0.40	0.38	0.47	0.38	**0.45**	**0.54**	**0.44**	0.42	0.49	0.41	**0.45**	**0.54**	**0.44**	0.42	0.49	0.41

Таблица 7: Точность (P), полнота (R) и F1-мера для всех извлечённых терминов

метод / текст	PageRank						C-value						C-value+PageRank					
	целиком			по главам			целиком			по главам			целиком			по главам		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
ФГ	0.68	0.52	0.59	0.66	0.51	0.57	**0.70**	**0.54**	**0.61**	0.64	0.49	0.56	**0.70**	**0.54**	**0.61**	0.64	0.49	0.56
ДУ	**0.41**	**0.75**	**0.53**	0.24	0.71	0.36	0.40	0.73	0.52	0.24	0.68	0.35	0.40	0.73	0.52	0.24	0.68	0.35
ДМ	0.51	0.61	0.55	0.56	0.56	0.56	0.55	0.66	0.60	**0.57**	**0.58**	**0.58**	0.55	0.66	0.60	**0.57**	**0.58**	**0.58**
ИИ	0.33	0.53	0.41	0.36	0.50	0.42	0.38	0.60	0.47	**0.39**	**0.55**	**0.46**	0.38	0.60	0.47	**0.39**	**0.55**	**0.46**
ЯЛ	0.52	0.71	0.60	0.48	0.57	0.52	**0.56**	**0.76**	**0.64**	0.52	0.61	0.56	**0.56**	**0.76**	**0.64**	0.52	0.61	0.56
СП	0.36	0.24	0.29	0.39	0.26	0.31	**0.46**	**0.31**	**0.37**	**0.46**	**0.31**	**0.37**	**0.46**	**0.31**	**0.37**	**0.46**	**0.31**	**0.37**
МА	0.54	0.30	0.38	0.41	0.23	0.29	**0.57**	**0.31**	**0.40**	0.54	0.30	0.38	**0.57**	**0.31**	**0.40**	0.55	0.31	0.39
Среднее	0.48	0.52	0.48	0.44	0.48	0.43	**0.52**	**0.56**	**0.52**	0.48	0.50	0.47	**0.52**	**0.56**	**0.52**	0.48	0.50	0.47



(a) Обработка текста целиком



(b) Обработка текста по главам

Рис. 3: F1-мера

Таким образом, рассмотренные в настоящей работе методы комбинирования, включающие ранжирование по *C-value* и комбинации *C-value* и *PageRank*, показали хорошее качество решения задачи автоматического извлечения терминов из отдельного текста (57–74% средней точности, 37–64% F1-меры), что в среднем выше качества решения данной задачи (средней точности 23–65%, F1-меры 5–38%), показанного в работе [11].

5 Заключение

Основные результаты данной магистерской диссертации:

- Проведён обзор современных методов автоматического извлечения терминов из текстов.
- Предложен способ построения графа терминов для ранжирования с помощью персонализированного PageRank, на основе близости терминов в контекстном окне текста.
- Реализованы комбинации методов извлечения и ранжирования терминов включая:
  - три вида лингвистических шаблонов;
  - фильтрацию по стоп-словам;
  - статистические меры (частота, C-value);
  - метод PageRank на основе предложенного способа построения графа.
- Проведена серия экспериментов с рассматриваемыми комбинациями и оценена их эффективность. Исследованные комбинации методов извлечения терминов показали достаточную эффективность.

По результатам работы были сделаны доклад на молодежном научном форуме «ЛОМОНОСОВ-2021» и публикация тезисов доклада[30].

6 Список литературы

[1] Loukachevitch N., Nokel M. An experimental study of term extraction for real information-retrieval thesauri. Proceedings of Terminology and Artificial Intelligence Conference TIA-2013, Citeseer, 2013, p. 69—78.

[2] Bolshakova E.I., Ivanov K.M. Term extraction for constructing subject index of educational scientific text. Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference "Dialogue", Moscow, 2018, p. 143–152.

[3] Arăn M., Turchi M., Tonelli S., Buitelaar P. Leveraging bilingual terminology to improve machine translation in a CAT environment. Natural Language Engineering, 05, 2017, p. 1–26.

[4] Anh Tuan Luu, Yi Tay, Siu Cheung Hui, See Kiong Ng. Learning Term Embeddings for Taxonomic Relation Identification Using Dynamic Weighting Neural Network. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2016, p. 403–413.

[5] Захаров В.П., Хохлова М.В. Автоматическое выявление терминологических словосочетаний // Структурная и прикладная лингвистика – 2014. № 3. – с. 182–200.

[6] Pazienza M.T., Pennacchiotti M., Zanzotto F.M. Terminology Extraction: An Analysis of Linguistic and Statistical Approaches. Knowledge Mining, Springer Berlin Heidelberg, 2005, p. 255–279.

[7] Astrakhantsev N. ATR4S: Toolkit with State-of-the-Art Automatic Terms Recognition Methods in Scala. Lang. Resour. Eval., Vol. 52, No. 3, 2018, p. 853–872.

[8] Wang R., Liu W., McDonald C. Featureless Domain-Specific Term Extraction with Minimal Labelled Data. Proceedings of the Australasian Language Technology Association Workshop, 2016, p. 103–112.

[9] Большакова Е.И., Лукашевич Н.В., Нокель М.А. Извлечение однословных терминов из текстовых коллекций на основе методов машинного обучения // Информационные технологии – 2013, № 7 – с. 31–37.

[10] Лукашевич Н.В., Логачев Ю.М. Комбинирование признаков для автоматического извлечения терминов // Вычислительные методы и программирование – 2010, № 11 – с. 108–116.

[11] Šajatović A., Buljan M., Šnajder J., Dalbelo Bašić B. Evaluating Automatic Term Extraction Methods on Individual Documents. Proceedings of the Joint Workshop on

Multiword Expressions and WordNet (MWE-WN 2019), Association for Computational Linguistics, 2019, p. 149–154.

[12] Zhang Z., Gao J., Ciravegna F. SemRe-Rank: Improving Automatic Term Extraction by Incorporating Semantic Relatedness with Personalised PageRank. Association for Computing Machinery, Vol. 12, No. 5, 2018.

[13] Большакова Е.И., Иванов К.М., Сапин А.С., Шариков Г.Ф. Система для извлечения информации из текстов на базе лексико-синтаксических шаблонов // Пятнадцатая национальная конференция по искусственному интеллекту с международным участием (КИИ-2016) – Труды конференции. Том 1 – Смоленск, Универсум, 2016.

[14] Frantzi K., Ananiadou S., Mima H. Automatic Recognition of Multi-word Terms: The C-value/ NC-value Method. Int. J. on Digital Libraries, Vol. 3, 2000, p. 115–130.

[15] Zhang Z., Iria J., Brewster C., Ciravegna F. A Comparative Evaluation of Term Recognition Algorithms. Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), 2008.

[16] Nokel M.A., Bolshakova E.I., Loukachevich N.V. Combining Multiple Features for Single-Word Term Extraction. Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference "Dialogue" (2012). Issue 11. Vol. 1 of 2. Main conference program. Moscow, RGGU, 2012, p. 490–501.

[17] Cram D., Daille B. Terminology Extraction with Term Variant Detection. Proceedings of ACL-2016 System Demonstrations, Association for Computational Linguistics, 2016, p. 13–18.

[18] Mykowiecka A., Marciniak M. Combining Wordnet and Morphosyntactic Information in Terminology Clustering. Proceedings of COLING 2012, 2012, p. 1951–1962.

[19] Hätty A., Schulte im Walde S. Fine-Grained Termhood Prediction for German Compound Terms Using Neural Networks. Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018), Association for Computational Linguistics, 2018, p. 62–73.

[20] Mykowiecka A., Marciniak M., Rychlik P. Recognition of non-domain phrases in automatically extracted lists of terms. Proceedings of the 5th International Workshop on Computational Terminology (Computerm2016), 2016, p. 12–20.

[21] Khan M.T., Yukun Ma, Kim J. Term Ranker: A Graph-Based Re-Ranking Approach. FLAIRS Conference, 2016.

[22] Mihalcea R., Tarau P. Textrank: Bringing order into text. Proceedings of the 2004 conference on empirical methods in natural language processing, 2004, p. 404–411.

[23] Wan X., Xiao J. CollabRank: Towards a Collaborative Approach to Single-Document Keyphrase Extraction. Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), 2008, p. 969–976.

[24] Florescu C., Caragea C. PositionRank: An Unsupervised Approach to Keyphrase Extraction from Scholarly Documents. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, 2017, p. 1105–1115.

[25] Boudin F. Unsupervised Keyphrase Extraction with Multipartite Graphs. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), 2018, p. 667–672.

[26] Stuart R., Engel D., Cramer N., Cowley W. Automatic Keyword Extraction from Individual Documents. Text Mining: Applications and Theory, 2010, p. 10–20.

[27] Campos R., Mangaravite V., Pasquali A., Jatowt A., Jorge A., Nunes C., Jatowt A. YAKE! Keyword extraction from single documents using multiple local features. Information Sciences Journal, Elsevier, Vol 509, 2020, p. 257-289.

[28] Selivanov A.A., Moloshnikov I.A., Rybka R.B., Sboev A.G. Keyword Extraction Approach Based on Probabilistic-Entropy, Graph, and Neural Network Methods. Artificial Intelligence, Springer International Publishing, 2020, p. 284–295.

[29] Bolshakova E., Efremova N., Ivanov K. Terminological Information Extraction from Russian Scientific Texts: Methods and Applications. Proceedings of Third Workshop "Computational linguistics and language science". EPiC Series in Language and Linguistics, V.4, 2019, EasyChair, p.95-106.

[30] Материалы Международного молодежного научного форума «ЛОМОНОСОВ-2021» / Отв. ред. И.А. Алешковский, А.В. Андриянов, Е.А. Антипов, Е.И. Зимакова. [Электронный ресурс] – М.: МАКС Пресс, 2021. – 1 электрон. опт. диск (DVD-ROM)

Приложение А

Примеры лексико-синтаксических шаблонов

Примеры шаблонов грамматических образцов

```
TERM =N1 (N1)
TERM =N1 N2<c=gen> (N1)
TERM =A1 N1 <A1=N1> (N1)
TERM =Pa1 N1 <Pa1=N1> (N1)
TERM =A1 A2 N1 <A1=A2=N1>
TERM =Pa1 Pa2 N1 <Pa1=Pa2=N1>
TERM =A1 N1 N2<c=gen> <A1=N1> (N1)
TERM =Pa1 N1 N2<c=gen> <Pa1=N1> (N1)
TERM =N1 N2<c=gen> N3<c=gen> (N1)
```

Примеры шаблонов авторских терминов

```
TrustedAUTH = "будем" "понимать" "под" Term1<c=ins>
TrustedAUTH = "введем" "понятие" Term1 "," "под" "которым" "будем"
"понимать"
TrustedAUTH = "назовем" Term1<c=ins>
TrustedAUTH = "под" {"словом"|"термином"}<1,1> Term1<c=nom>
{"понимаетсЯ"|"будем" "понимать"|V1<пониматьсЯ, t=pres, p=3, m=ind>}<1,1>
TrustedAUTH = Pn1<который> "можно" ["было" "бы"] "назвать" ["также"]
Term1<c=ins>
UntrustedAUTH = Term1<c=nom> ["-"] {"есть"|"это"}<1,1>
UntrustedAUTH = Pr1 Term1 {"\(" | "," }<1,1> {"т.е."|"то" "есть"|"т."
"е."}<1,1> Pr1
UntrustedAUTH = Term1 {"\(" | "," }<1,1> {"т.е."|"то" "есть"|"т." "е."}<1,1>
```

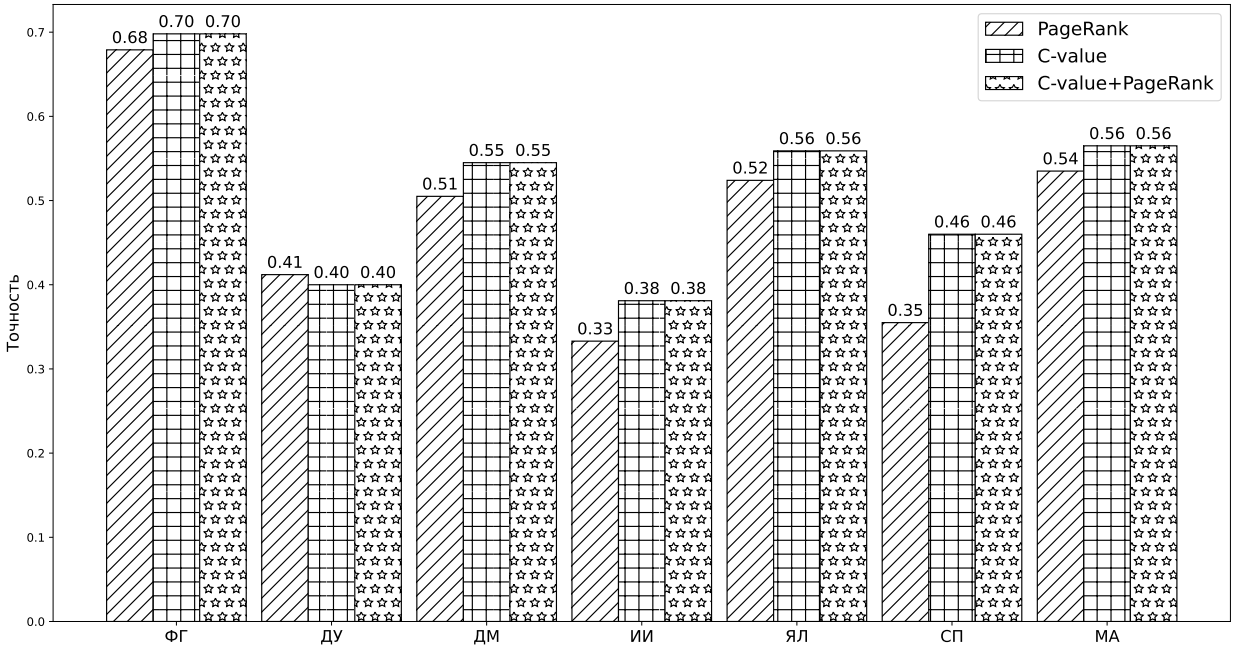
Примеры шаблонов для синонимов терминов

```
SYN-N-N = MSPN1 "\("MSPN2"\)" (MSPN1)
SYN-N-N = MSPN1 "\(" ["или"] MSPN2 "\)" <MSPN1.c=MSPN2.c>
SYN-N-N = MSPN1 "," "или" ["просто"] MSPN2
SYN-N-N = TermN1<c=acc> "будем" ["также"] "называть" TermN2<c=ins>
SYN-N-N = TermN1"\(" "далее" ["-"] TermN2<c=nom> "\)"
SYN-N-N = TermSynN1 "," "или" ["просто"] TermSynN2 <TermSynN1.c=TermSynN2.c>
(TermSynN1)
```

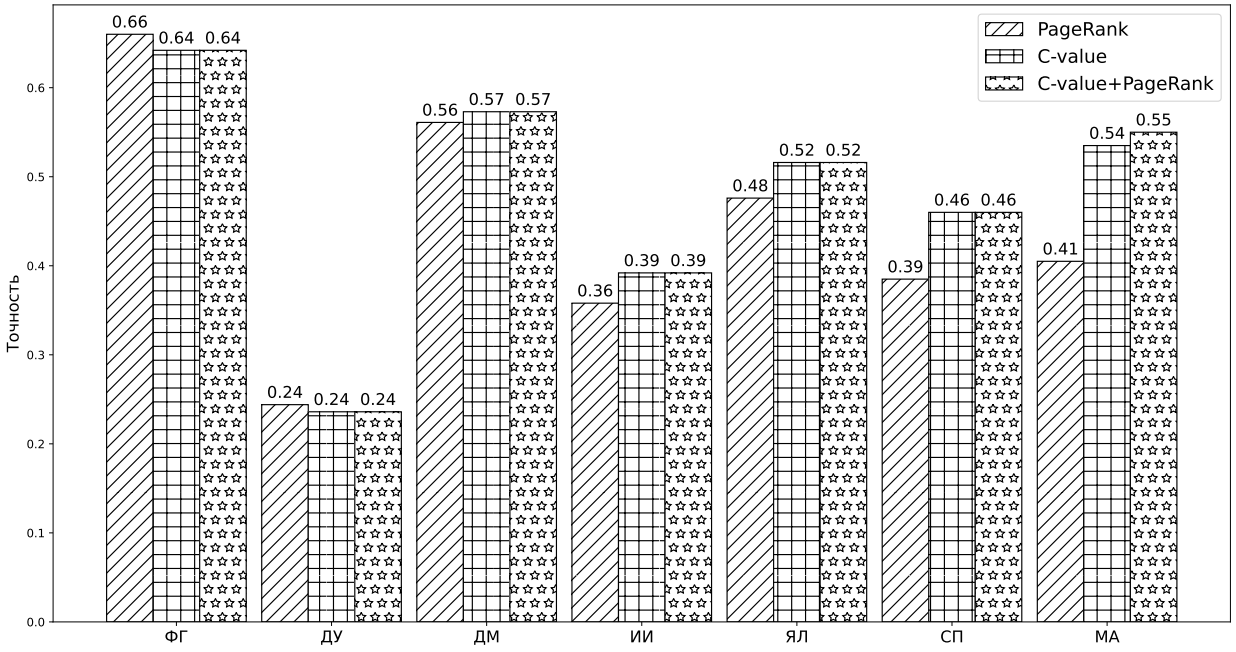
Приложение Б

Точность и полнота извлечения терминов

Точность

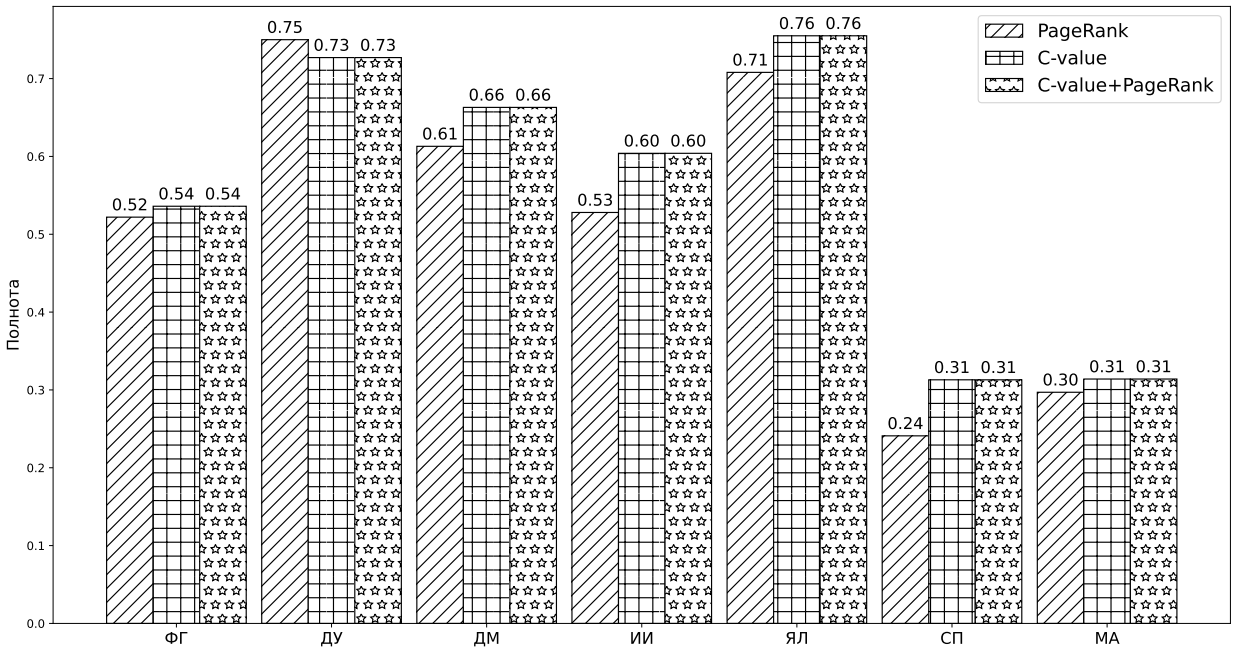


(a) Обработка текста целиком

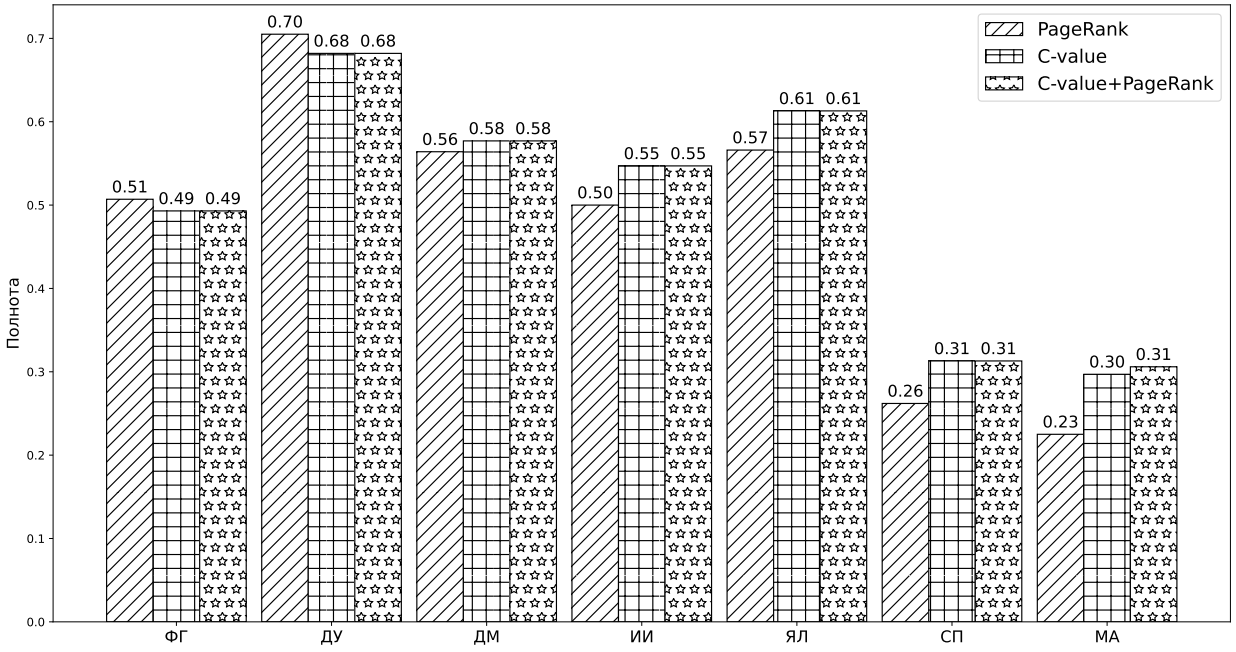


(b) Обработка текста по главам

Полнота



(c) Обработка текста целиком



(d) Обработка текста по главам