**Инженерный вестник Дона**, №12 (2020)
ivdon.ru/ru/magazine/archive/n12y2020/6705

# Разработка веб-сервиса для поиска файлов по ключевым словам

*С.И. Нуриев, Ф.М. Бикмуратов, Н.П. Пашин, А.Ш. Хафизова*

*Казанский национальный исследовательский технический университет имени А. Н. Туполева – КАИ, Казань*

**Аннотация:** Предмет исследований – разработка сервиса для поиска по файлам пользователя по заданному набору ключевых слов с параметрами. Были изучены имеющиеся подходы к решению такой задачи и выбран наиболее релевантный. Сервис осуществляет поиск внутри файлов с текстовым содержимым с целью автоматизации процесса выделения нужных файлов среди всего множества. В основе его работы лежит алгоритм Портера и используется подход стемминга текста с целью получения более точных результатов. Выполняется поиск основы слова, учитывающий морфологию. Выполняя морфологический разбор слова, находим общую для всех его грамматических форм основу, отсекая суффиксы и окончания. В результате алгоритм работы сервиса позволяет искать не просто по заданным ключевым словам, но и учитывает их словоформы, а также ищет сразу по нескольким наборам ключевых слов, причём каждый набор анализируется отдельно. Помимо этого можно задавать диапазоны числовых значений для поиска. Особенность сервиса в том, что наборы ключевых слов ищутся совместно в ближних абзацах в интервале окрестности от -20 до +20 слов друг от друга, учитывая таким образом контекст их появления в тексте. Сервис ранжирует найденные документы по качеству соответствия критериям поиска. Обрабатываются файлы в основных форматах: doc, xls, pdf, txt. Сервис функционирует на платформе Linux под управлением веб-сервера Apache. Для разработки использованы бесплатные программные инструменты.

**Ключевые слова:** поисковая система, анализ документов, стемминг, алгоритм Портера, словоформы, морфология, среднее-арифметическое процентов, веб-сервис.

## Введение

Во многих компаниях сотрудникам приходится регулярно работать с большим количеством разнотипных текстовых документов: doc, pdf, xls и другие. В частности выполнять поиск нужных документов, как среди накопленных архивов, так и выделять интересующие документы в массиве поступающих новых. Как правило, для решения этой задачи сотрудник вручную открывает все документы и осуществляет просмотр контента для выделения интересующих документов. С учетом того, что объем документации растет лавинообразно процесс этот достаточно трудозатратен [1-2]. На сегодняшней день для автоматизации этой задачи есть несколько standalone descktop приложений [3]. К числу наиболее известных можно

отнести: поиск средствами проводника Windows (появился только в последней ОС Windows 10, установленной у малого количества сотрудников и требовательной к аппаратным ресурсам); файловый менеджер TotalCommander – удобный инструмент, но практически не используется офисными сотрудниками. Главный недостаток существующих программ в том, что они реализуют «простой поиск» среди файлов на предмет полного совпадения заданных ключевых слов без учета словоформ [4-5] и не ранжируют найденные документы по «качеству» соответствия.

Для решения рассматриваемой задачи предлагается разработать веб-сервис, осуществляющий поиск нужных документов среди множества загруженных на сервер файлов по заданному набору ключевых слов с параметрами [6-8]. При этом сервис учитывает слофовормы, используя подход стемминга и ранжирует в процентном соотношении найденные документы.

## Подходы для поиска текста в файле

Для работы поисковых алгоритмов применяют множество подходов. Можно выделить три подхода к реализации информационного поиска:

1. Прямой поиск – посимвольное сравнении строки с подстрокой.

2. Автоматический морфологический анализ – процедура, позволяющая из формы слова извлечь информацию о его грамматических признаках. Виды автоматического морфологического анализа: со словарем основ (лемматизация)[1], со словарем словоформ, методом логического умножения, без словаря (стемминг) [4].

3. Индексно-последовательный поиск – составляется структура данных индекса, позволяющая указать местоположение каждого слова в документе[3].

Для решения поставленных задач при разработке сервиса было решено использовать второй подход.

В лемматизации используется готовая база данных с лексемами слов. Эти базы существуют только для английского языка. Поэтому в разработке системы данный подход не рассматривается, так как отсутствует решение для реализации поискового алгоритма. Была использована идея стемминга. В ее основе лежит стеммер Портера. Стеммер Портера – алгоритм стемминга, опубликованный Мартином Портером в 1980 году [9]. Оригинальная версия стеммера была предназначена для английского языка и была написана на языке BCPL. Впоследствии Мартин создал проект «Snowball» и, используя основную идею алгоритма, написал стеммеры для распространенных индоевропейских языков, в том числе для русского. Алгоритм не использует баз основ слов, а лишь, применяя последовательно ряд правил, отсекает окончания и суффиксы, основываясь на особенностях языка, в связи с чем работает быстро, но не всегда безошибочно. В разработке использовались следующие библиотеки стемминга: php-lingua-stem-ru, php-stemmer, Stemmer. Данные стеммеры представлены на сайте github.com и публикуются с лицензией MIT [10].

## Алгоритм работы разрабатываемого сервиса

Сервис имеет веб-интерфейс (рис.1), в котором пользователь загружает свои файлы на сервер и указывает ключевые слова и их параметры для поиска. Эти данные сохраняются во временной таблице, создаваемой под каждый очередной поисковый запрос пользователя. Содержимое всех загруженных файлов приводится к текстовому виду с помощью готовых специализированных утилит.

Входными данными для работы алгоритма поиска разработанного веб-сервиса являются:

1. Поисковые параметры, вносимые в таблицу пользователем.

2. Прикладываемые файлы различных форматов, содержащие интересующую текстовую информацию, загруженные на сервер.

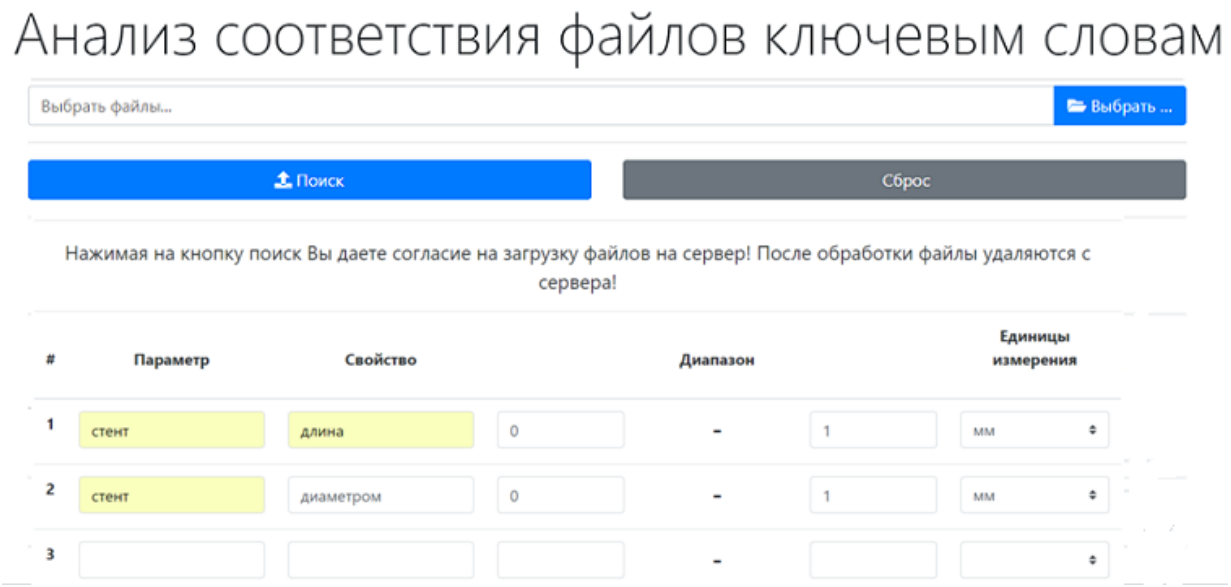
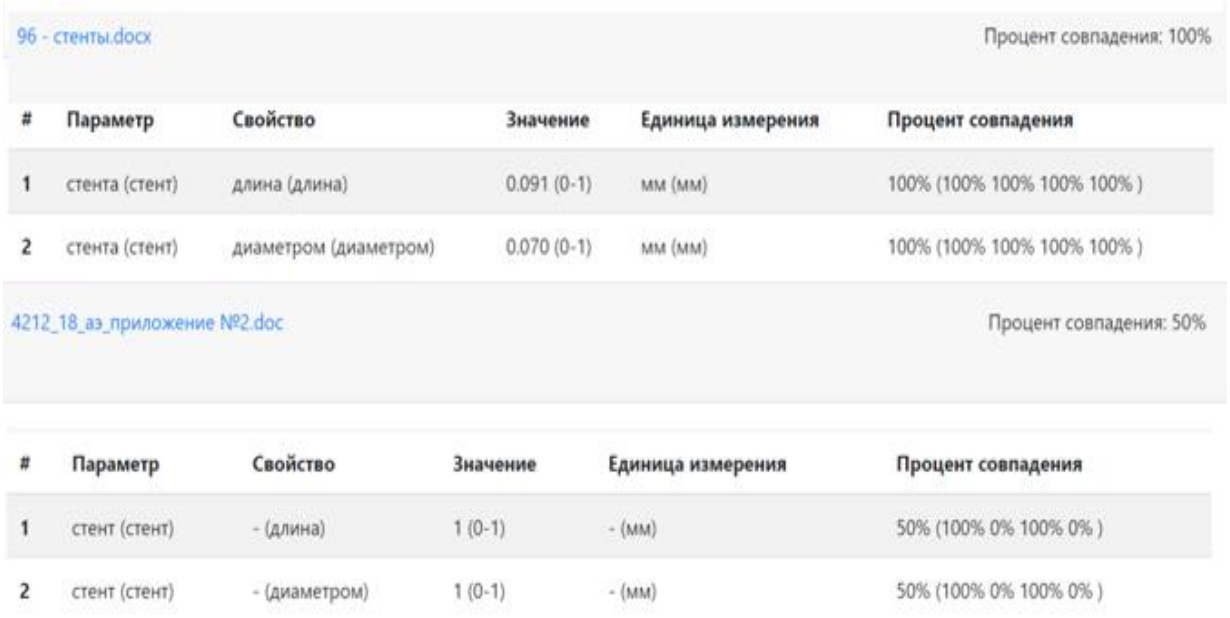



Рис.1 – Интерфейс главной страницы сервиса

Алгоритм основывается на стемминге текста, с целью получения более точных результатов поиска. Он выполняет поиск основы слова, учитывающий морфологию исходного слова. Выполняя морфологический разбор слова, находит общую для всех его грамматических форм основу, отсекая суффиксы и окончания. Пример результатов поиска по файлам представлен на рис.2. Общий алгоритм поиска в виде блок-схем – на рис. 3.



96 - стенты.docx — Процент совпадения: 100%

| # | Параметр | Свойство | Значение | Единица измерения | Процент совпадения |
|---|---|---|---|---|---|
| 1 | стента (стент) | длина (длина) | 0.091 (0-1) | мм (мм) | 100% (100% 100% 100% 100% ) |
| 2 | стента (стент) | диаметром (диаметром) | 0.070 (0-1) | мм (мм) | 100% (100% 100% 100% 100% ) |

4212_18_аэ_приложение №2.doc — Процент совпадения: 50%

| # | Параметр | Свойство | Значение | Единица измерения | Процент совпадения |
|---|---|---|---|---|---|
| 1 | стент (стент) | - (длина) | 1 (0-1) | - (мм) | 50% (100% 0% 100% 0% ) |
| 2 | стент (стент) | - (диаметром) | 1 (0-1) | - (мм) | 50% (100% 0% 100% 0% ) |

Рис.2 – Пример отображения результатов поиска по двум файлам

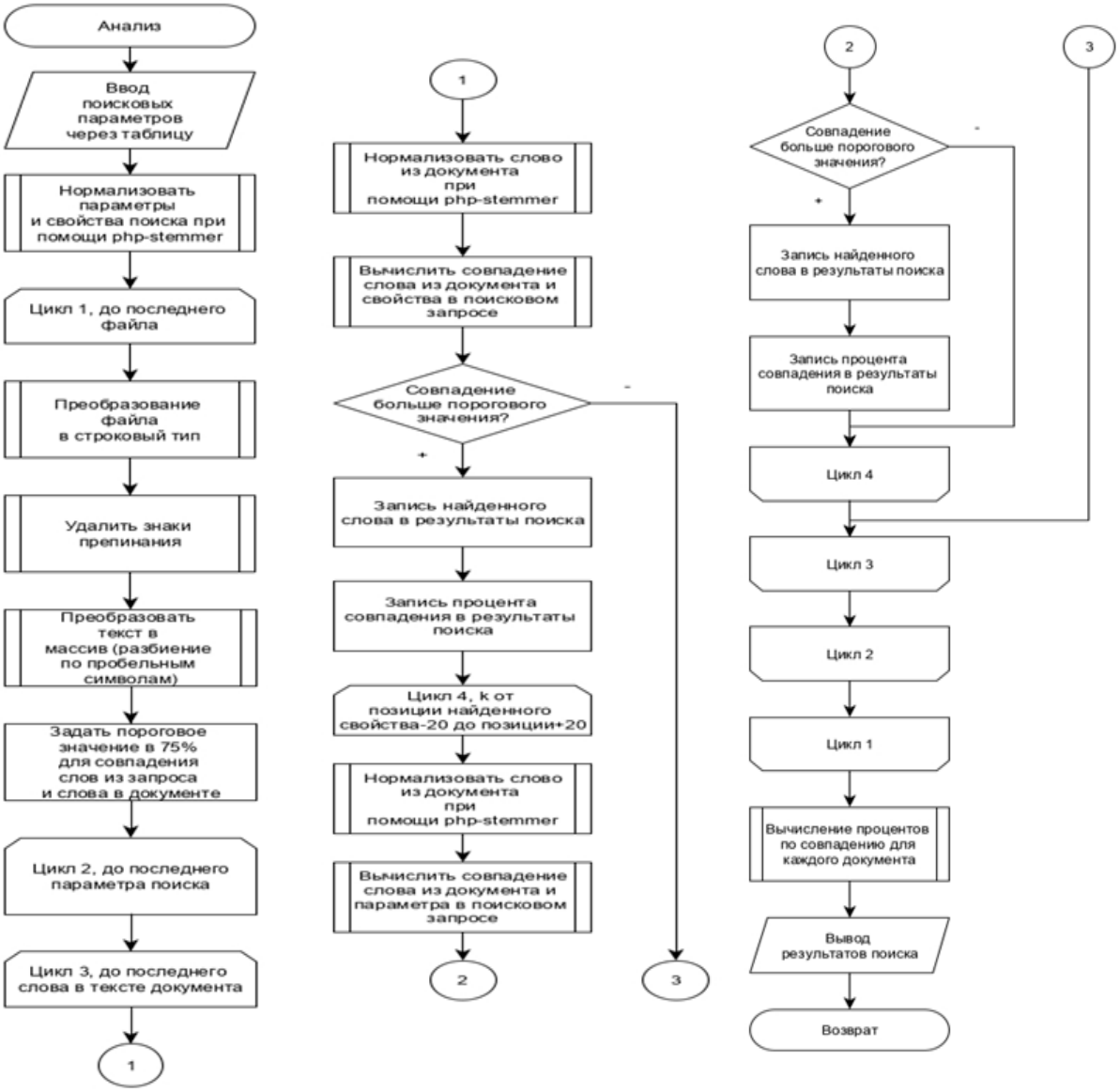



Рис.3. – Блок-схемы алгоритма анализа и поиска

В первой части блок-схемы алгоритма производится ввод и нормализация поисковых параметров из таблицы. Организовывается цикл по списку документов и каждый документ преобразовывается в строковый тип для удобной обработки в коде программы при помощи библиотеки-стеммера. Применяется символьная фильтрация, что означает удаление знаков препинания из текста и заменой их на пробельный символ [11]. Далее производится преобразование строки в массив, разделителем элементов служит пробельный символ.

Установлено пороговое значение для совпадения слов равное 75%. Это было выяснено эмпирическим путём в ходе подбора на входных тестовых данных. Старт вложенного цикла по поисковым параметрам и в нём же, вложенный цикл по словам в выбранном преобразованном документе. В результате получается 2 уровня вложенности циклов. В цикле нормализуется слово из документа, вычисляется его совпадение с поисковым словом и если совпадение больше порогового значения, то производится запись искомого слова в результаты поиска и процент совпадения. Иначе, цикл переходит к следующему слову в документе.

Если совпадение найдено, то производится поиск следующего слова в окрестности от данного в диапазоне от -20 до +20 от текущей позиции. Вычисление совпадения производится аналогичным образом.

Если таблица заполнена результатами поиска, то она выводится в виде HTML-таблицы и с взвешенными процентами совпадения для каждого документа, что позволяет выявить релевантный документ в соответствии с поисковым запросом пользователя [12]. Взвешенный итоговый процент по каждому документу рассчитывается по следующему алгоритму.

Вычисляется среднее арифметическое из всех процентов. Если есть пустые результаты поиска для одного из параметров или же поисковых параметров 2 или меньше, то производится расчет по формуле, так как влияние меньшего количества найденных параметров слишком мало, поэтому нужно производить коррекцию формулы.

$$averagePercent = \frac{sumPercent}{searchParameters - 1} \qquad (1)$$

Иначе, расчет производится в другом виде:

$$averagePercent = \frac{sumPercent}{searchParameters} \qquad (2)$$

## Заключение

В результате проектирования и разработки был реализован веб-сервис для поиска файлов по заданным ключевым словам. Сервис размещен в сети Интернет на сервере, им может воспользоваться любой желающий. Тестирование функциональности и качества поиска было проведено на примере задачи поиска тендеров, подходящих под заданные критерии медицинской компании. С тендерной площадки была скачана конкурсная документация по десяти лотам, которая была загружена на сервис и далее был осуществлен поиск тендеров, посвященным поставкам стентов с заданными параметрами. В результате были выделены тендеры наиболее интересных компаний, было сэкономлено время сотрудников по ручному анализу конкурсной документации. Получен положительный отзыв на разработанную систему. В дальнейшем для добавления возможности анализировать графическую информацию в сервис планируется встроить обученную глубокую сверточную нейросеть [13-15].

## Литература

1. Hobson L., Cole H., Hannes H. Natural Language Processing in Action - Munning, 2019. - 544p.

2. Строцев В.А.. Информативность частотных характеристик N-грамм текстовых фрагментов // Инженерный вестник Дона, 2013, №1. URL: ivdon.ru/ru/magazine/archive/n1y2013/1492.

3. Белоногов Г.Г., Кузнецов Б.А. Языковые средства автоматизированных информационных систем. – М.: Наука, 1983. – 288 с.

4. Oliveira R.A., Junior C. M. Experimental Analysis of Stemming on Jurisprudential Documents Retrieval. Information 2018, 9, 28.

5. Иванова Д.Н., Яровая Л.Е.Модели анализа словообразования в современном английском языке // Инженерный вестник Дона, 2020, №8. URL: ivdon.ru/ru/magazine/archive/n8y2020/6584.

6. Nuriev S.I., Gazizova A.I., Minyazev R.Sh. Searching inside binary and text files Материалы конференций ГНИИ «НАЦРАЗВИТИЕ». 2019. С. 271-274.

7. Минязев Р.Ш., Дыганов С.А., Гумеров И.Р., Перухин М.Ю. Разработка сервиса для идентификации полей сканированного документа с использованием библиотеки машинного распознавания tesseract-ocr // Вестник технологического университета, 21, 9, 132-135, 2018.

8. Minyazev R. Sh., Rumyantsev A. A., Dyganov S. A. and Baev A. A. X-Ray Image Analysis for the Neural Network-Based Detection of Pathology // Bulletin of the russian academy of sciences: physics Vol. 82, № 12 2018 pp. 1685–1688.

9. Porter M.F. An algorithm for suffix stripping, Program, 14(3), 1980, pp. 130−137.

10. Стеммер Портера для русского языка. DateViews 24.06.2020.URL: github.com/NeonXP/Stemmer/.

11. Лойко В.И. Структуры и алгоритмы обработки данных. Учебное пособие для вузов.– Краснодар: КубГАУ. 2004. - 261 с.

12. Линник, Ю.В. Метод наименьших квадратов и основы математико-статистической теории обработки наблюдений. – М.: Физматгиз, 1958. – 336 с.

13. Minyazev R. Sh., Rumyantsev A. A., Baev A. A. and Baeva T. D. Using a Neural Network to Separate Lungs in X-ray Images // Bulletin of the Russian Academy of Sciences: Physics, 2019, Vol. 83, № 12, pp. 1494–1497.

14. Vankov Y., Rumyantsev A., Ziganshin S., Politova T., Minyazev R.and Zagretdinov A. Assessment of the Condition of Pipelines Using Convolutional Neural Networks // 2020 Energies, 13 (3), art. № 618.

15. Гибадуллин Р.Ф., Лекомцев Д.В., Перухин М.Ю. Анализ параметров промышленных сетей с применением нейросетевой обработки // Искусственный интеллект и принятие решений. 2020. № 1. С. 80-87.

## References

1. Hobson L., Cole H., Hannes H. Natural Language Processing in Action - Munning, 2019. - 544p.

2. Strocev V.A. Inzhenernyj vestnik Dona, 2013, №1. URL: ivdon.ru/ru/magazine/archive/n1y2013/1492.

3. Belonogov, G.G., Kuzneczov B.A. Yazy`kovy`e sredstva avtomatizirovanny`x informacionny`x system [Language tools of automated information systems]. M.: Nauka, 1983. 288 p.

4. Oliveira R.A., Junior C. M. Experimental Analysis of Stemming on Jurisprudential Documents Retrieval. Information 2018, 9, 28.

5. Ivanova D.N., Yarovaya L.E. Inzhenernyj vestnik Dona, 2020, №8. URL: ivdon.ru/ru/magazine/archive/n8y2020/6584.

6. Nuriev S.I., Gazizova A.I., Minyazev R.Sh. Materialy` konferencij GNII «NACzRAZVITIE». 2019. pp. 271-274.

7. Minyazev R.Sh., Dy`ganov S.A., Gumerov I.R., Peruxin M.Yu. Vestnik texnologicheskogo universiteta, 21, 9, 132-135, 2018.

8. Minyazev R. Sh., Rumyantsev A. A., Dyganov S. A. and Baev A. A. Bulletin of the russian academy of sciences: physics Vol. 82, № 12 2018 pp. 1685–1688.

9. Porter M.F. An algorithm for suffix stripping, Program, 14(3), 1980, pp. 130−137.

10. Stemmer Portera dlya russkogo yazy`ka [Porter's Stemmer for Russian language]. DateViews 24.06.2020. URL: github.com/NeonXP/Stemmer/.

11. Lojko V.I. Struktury` i algoritmy` obrabotki danny`x. Uchebnoe posobie dlya vuzov [Data processing structures and algorithms. Textbook for universities]. Krasnodar: KubGAU. 2004. 261 p.

12. Linnik Yu.V. Metod naimen`shix kvadratov i osnovy` matematiko-statisticheskoj teorii obrabotki nablyudenij [The method of least squares and the

foundations of the mathematical and statistical theory of observation processing]. M.: Fizmatgiz, 1958. 336 p.

13. Minyazev R. Sh., Rumyantsev A. A., Baev A. A. and Baeva T. D. Bulletin of the Russian Academy of Sciences: Physics, 2019, Vol. 83, No. 12, pp. 1494–1497.

14. Vankov Y., Rumyantsev A., Ziganshin S., Politova T., Minyazev R.and Zagretdinov A. 2020 Energies, 13 (3), art. № 618.

15. Gibadullin R.F., Lekomcev D.V., Peruxin M.Yu. Iskusstvenny`j intellekt i prinyatie reshenij. 2020. № 1. pp. 80-87.