УДК 004.91

В. В. Дягилев [1], **А. А. Цхай** [2], **С. В. Бутаков** [3]

[1] Алтайский государственный технический университет им. И. И. Ползунова
пр. Ленина, 46, Барнаул, 656038, Россия
E-mail: diagilev@gmail.com

[2] Алтайская академия экономики и права
пр. Комсомольский, 86, Барнаул, 656038, Россия
E-mail: taa1956@mail.ru

[3] Международная школа бизнеса Солбридж
151-13 Samsung 1-Dong, Dong-gu, Daejeon 300-814, Южная Корея
E-mail: butakov@solbridge.ac.kr

АРХИТЕКТУРА СЕРВИСА ОПРЕДЕЛЕНИЯ ПЛАГИАТА, ИСКЛЮЧАЮЩАЯ ВОЗМОЖНОСТЬ НАРУШЕНИЯ АВТОРСКИХ ПРАВ

Описывается архитектура сервиса определения плагиата, позволяющая защищать интеллектуальную собственность авторов документов. Предлагаемая архитектура разделяет обработку документов на две части, одна из которых использует вычислительные ресурсы локальной инфраструктуры, другая – вычислительные мощности поисковых машин Интернета.

Ключевые слова: определение плагиата, архитектура сервиса, охрана авторского права.

Введение

Стремительное развитие сети Интернет наряду с увеличивающейся компьютерной грамотностью, к сожалению, способствует проникновению плагиата в различные сферы человеческой деятельности: плагиат является острой проблемой в образовании, промышленности и научном сообществе. По данным Государственного университета – Высшая школа экономики в среднем около 50 % студентов российских вузов «скачивают» рефераты и курсовые работы из сети Интернет [1]. Имеются данные о том, что число студентов в американских средних школах, вовлеченных в различные виды плагиата, достигает 90 % [2]. Проблема плагиата существует и активно обсуждается в научном сообществе. Один из таких случаев публично рассматривался в обществе IEEE и привел к аннулированию ученой степени [1]. Проблема плагиата отмечена Росфиннадзором при рассмотрении целесообразности использования бюджетных средств, выделенных на НИОКР в 2009 г. [2]. Не менее актуально эта проблема стоит при рассмотрении конкурсных проектов в государственные инновационные программы и фонды, а также при регистрации заявок в Роспатенте.

Проблема плагиата многогранна. Сам плагиат может варьироваться от прямого копирования текстов до плагиата идей. Само понятие «сходства» может быть формализовано различными способами [3]. В данной статье внимание концентрируется на технической стороне построения систем поиска сходства в текстах. Рассматривается архитектура систем обнару-

[1] Kompas (2010) Saving Indonesia from Traps of Plagiarism. http://english.kompas.com/read/2010/04/28/02563687/Saving.Indonesia.from.Traps.of.Plagiarism

[2] Информационная справка о результатах проверки использования министерствами, ведомствами, внебюджетными фондами и подведомственными им организациями бюджетных средств, выделенных в 2009 г. на научно-исследовательские и опытно-конструкторские работы: http://www.rosfinnadzor.ru/page/index/1236/page/7550

жения плагиата (СОП). Предложена архитектура системы, позволяющая исключить возможность плагиата на этапе экспертизы представленных текстов.

Большинство коммерческих СОП не раскрывают деталей своей структуры и алгоритмов работы, чтобы снизить уровень уязвимости к различным способам их обмана (в теории информационной безопасности называемых атаками). Однако анализ декларируемых принципов работы СОП и детальное рассмотрение открытых СОП позволяют выявить типовую структуру для данных сервисов. Данная структура дает описание таких лидеров рынка СОП, как «Антиплагиат» (РФ), Turnitin, SafeAssign (США). Типовая структура СОП отображена на рис. 1.

Пользователь (студент или преподаватель) передает на проверку документ в СОП через информационную систему своего университета либо напрямую через web-интерфейс СОП. Затем содержимое документа преобразуется системой, с целью выделения «чистого» текста, т. е. избавления от форматирования документа присущего современным текстовым процессорам. На основе полученного текста строится запрос к базе данных документов СОП, результатом которого является набор документов, вероятных источников плагиата.

Детальное сравнение текстов документов позволяет определить подобные части и сформировать отчет о найденных совпадениях. В результате пользователь получает отчет о проведенной проверке с указанием частей текста и источников «заимствования», если таковые имелись. При этом база данных документов СОП может содержать как индексы открытых сегментов сети Интернет (как в случае с системой Turnitin), так и доступ к некоторым библиотекам с ограниченным доступом. Например, система «Антиплагиат» имеет доступ к базе данных диссертаций ВАК РФ.

Отметим два важных момента.

1. Содержимое проверяемого документа передается в СОП.

2. СОП по соглашению с пользователем сохраняет копию его документа в своей базе данных для использования в дальнейшем в качестве исходного материала для проверок.

Как поступить в случаях, когда необходимо осуществить проверку сведений, содержащихся в документе, на наличие плагиата и при этом соблюсти требования к конфиденциальности информации этих документов. Например, в случаях обработки заявок на изобретения,

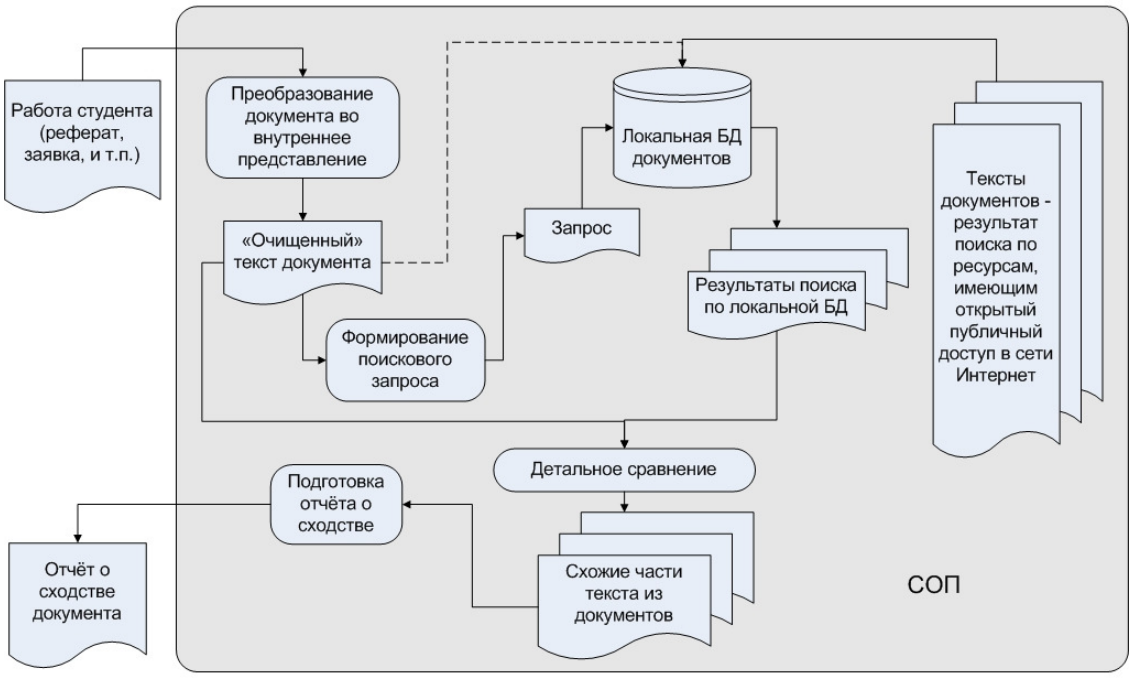

Рис. 1. Типовая архитектура СОП

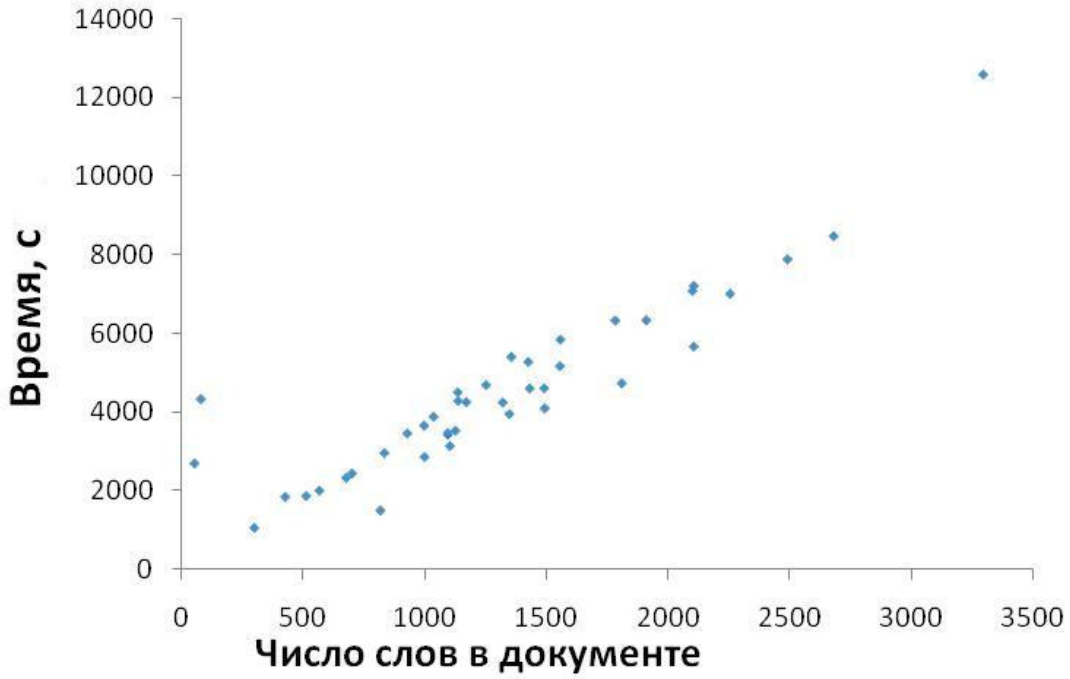

Рис. 2. Линейная зависимость между временем поиска и количеством слов в документе

полезные модели, промышленные образцы или при рассмотрении конкурсных проектов в государственные инновационные программы и фонды и т. п. Ограничение при помощи лицензирования не предупреждает техническую возможность неправомерного использования информации, передаваемой в СОП.

Один из возможных способов – это использование общедоступных поисковых машин Интернета, когда часть текста помещается в кавычки и осуществляется глобальный поиск схожих документов – производится попытка найти точные совпадения текста в документах, опубликованных в открытом доступе в сети Интернет. Подобная несложная техника может быть использована совместно с сервисом Google Alert [3].

Исследования показали, что такая техника поиска, с использованием публичных поисковых машин по критерию точного совпадения фраз, может быть очень эффективной [4], но такой поиск очень медленный, так как осуществляется вручную. При этом остается открытым вопрос: как определить ключевые фразы в документе, по которым вести поиск.

Для сформулированной цели возможно использование открытого программного обеспечения, например, системы Crot [5]. Данная система имеет типовую архитектуру СОП за исключением того, что локальная база данных документов состоит только из внутренних ресурсов организации пользователя, а нахождение документов – потенциальных кандидатов источников плагиата – выполняется путем отправки запросов к поисковой машине Интернета.

Система Crot выполняет исчерпывающий поиск, посылая запросы, сформированные «плавающим окном». Алгоритм «плавающего окна» выполняет прямой перебор фраз. Например, для Шекспировской фразы «to be, or not to be: that is the question» при длине окна $X = 4$ алгоритм сформирует 7 следующих запросов: «to be or not», «be or not to», «or not to be», «not to be that», «to be that is », «be that is the», «that is the question». Авторы системы Crot указывают, что если значительная часть текста документа была присвоена из какого-либо

[3] Carter M. (2008). How to Use Google Alerts to Detect Plagiarism. http://www.suite101.com/content/how-to-use-google-alerts-for-web-writers-a86525

источника в Интернете, то нет необходимости посылать все возможные запросы, а достаточно только 10 %, чтобы определить местонахождение этого источника [5]. Однако из-за большого количества запросов поиск «плавающим окном» значительно замедляет весь процесс обнаружения плагиата.

Результаты проведенного эксперимента показали линейную зависимость времени поиска от количества слов в документе. Эксперимент был выполнен с 60 документами объемом 350–3 500 слов, проводился на выделенном сервере с 100 Mbs интернет-каналом. Как показано на рис. 2, время поиска составляло около пяти минут на каждую 1 000 слов документа.

Предлагаемая архитектура сервиса определения плагиата

На рис. 3 отображена основная концепция предлагаемой архитектуры СОП. Сам сервис разделен на внутреннюю (клиентскую) часть, работающую на инфраструктуре пользователя, и на внешнюю (серверную) часть, работающую на инфраструктуре сторонней организации. Внутренняя часть выполняет функции сервера для обращений со стороны пользователей и одновременно с этим является клиентом, выполняющим запросы к внешней части сервиса.

Предполагается, что разделенная структура сервиса будет выполнять работы в следующем порядке.

1. Клиентская часть системы, получив документ, переданный пользователем для проверки, преобразует его в «чистый» текст.
2. Клиентская часть создает специальный запрос путем случайного выбора определенного количества запросов, сформированных методом «плавающего» окна.
3. Клиентская часть отправляет специальный запрос в серверную часть СОП.
4. Серверная часть, получив специальный запрос, формирует к поисковой машине Интернета запросы на нахождение документов, опубликованных в открытом доступе в сети Интернет.

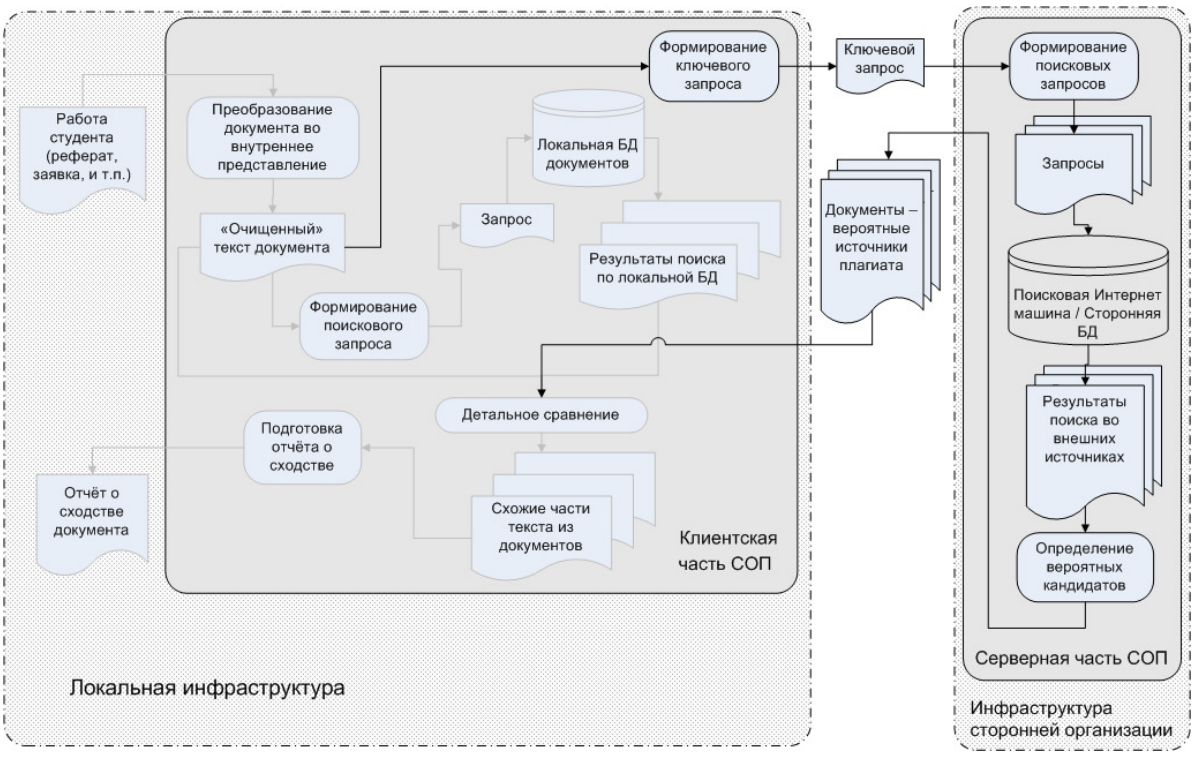

Рис. 3. Схема предлагаемой архитектуры СОП

5. Серверная часть получает множество ссылок на документы как результат выполнения запросов и выбирает те ссылки, которые встречаются чаще всего.

6. Серверная часть загружает документы, расположенные по выбранным ссылкам, и отправляет эти документы в клиентскую часть.

7. Клиентская часть производит детальное сравнение исходного текста и полученных документов.

8. Клиентская часть определяет схожие части текста и направляет отчет о сходстве исходного документа пользователю.

Большая часть работы сервиса определения плагиата происходит на оборудовании организации пользователя. Сторонняя же организация выполняет только глобальный поиск и предварительное (черновое) сравнение возможных источников плагиата из сети Интернет.

Для составления специального запроса в клиентской части СОП можно использовать алгоритм «плавающего окна», реализованный в системе Crot [5]. Данный алгоритм осуществляет полный перебор всех фраз длины *X*, доступных в проверяемом документе. С учетом того, что большинство поисковых машин допускает в обрабатываемых запросах не более 10 слов, то можно ограничить $X <= 10$.

Очевидно, что при длине текста *Y* слов, общее количество запросов $N = |Q|$, где $Q = \{q_1, q_2, ..., q_n\}$ – массив запросов, определяющийся формулой $N = Y - X + 1$. С учетом того, что *Y* существенно больше *X*, можно утверждать, что подобный алгоритм сформирует число запросов близкое к числу слов в документе.

При выполнении большого числа запросов к поисковой машине необходимо рассмотреть возможность появления следующих затруднений:

- увеличения времени поиска и повышенные требования к интернет-каналу в случае распараллеливания выполнения данных запросов;
- возможность восстановления документа из фраз, переданных поисковой машине.

Исследования показывают, что порядок передачи поисковых фраз *Q* не влияет на результаты поиска [5]. Иными словами, если фразы массива *Q* будут перемешаны в случайном порядке, то результат поиска совпадет с результатом, полученным при последовательной передаче. Случайное перемешивание не решает проблемы возможного восстановления документа – случайно перемешанная мозаика может быть легко восстановлена простым перебором, так как в полном массиве соседние элементы q_i и q_{i+1} содержат $X - 1$ совпадающих слов, что делает восстановление тривиальной задачей сбора мозаики фраз по пересечениям из $X - 1$ слов.

Чтобы определить местонахождение источника «заимствования», нет необходимости посылать все возможные запросы *Q*. Достаточно использовать только небольшой процент случайно выбранных элементов данного массива [5]. Насколько возможно использование этих свойств для ограничения передачи текста сторонней организации?

Пусть $Q_1 \subseteq Q$ – массив случайно выбранных элементов из полного массива запросов $Q : |Q_1| < |Q|$. Общее число слов в запросах, передаваемых в поисковую машину, будет равно $Y_s = |Q_1| \cdot X$. Таким образом, если $|Q_1|$ удовлетворяет неравенству $Y_s < Y$, где *Y* – общее число слов в документе, то можно гарантировать, что полное восстановление исходного текста на стороне поисковой машины из запросов, переданных ей, становится невозможным.

Таким образом, если $|Q_1| < \frac{Y}{X}$, то исходный документ гарантированно невосстановим. Это ограничение рекомендуется для определения доли запросов при формировании специального запроса, направляемого в серверную часть СОП.

Очевидно, что приведенная архитектура увеличивает требования к мощности вычислительных ресурсов, используемых на стороне клиентской части СОП. Данные увеличения касаются как вычислительной мощности, поскольку требуются вычислительные затраты для детального сравнения документов, так и дискового пространства для хранения данных документов.

В части дискового пространства, в случае использования алгоритмов хеширования, схожих с алгоритмом Винновинг [6], для хранения хешей требуется хранить около 5 % хешей, в расчете от числа символов в документе. При использовании 128- или 256-битных хешей и однобайтной кодировки текста можно говорить о том, что объем хешей будет примерно равен объему чистого текста в документе. Данное увеличение дискового пространства не представляется сколько-нибудь значимым с учетом постоянно снижающейся стоимости дисковой памяти.

В части вычислительных ресурсов хеширование одного документа не требует существенных вычислительных затрат при использовании локальных алгоритмов, подобных Винновинг [6], так как затраты линейны по отношению к длине текста. Повышенные требования могут предъявляться к СУБД на клиентской части СОП. Фактически именно затраты на приобретение лицензии и обслуживание СУБД будут определять увеличение стоимости клиентской части СОП в предложенной архитектуре. Данное увеличение должно компенсироваться снижением стоимости работы серверной части СОП, так как пользовательские документы не хранятся на клиентской части СОП.

Заключение

В статье рассмотрена новая архитектура СОП, позволяющая определять в проверяемом документе наличие присвоенного материала из текстов, опубликованных в открытом доступе в сети Интернет. При этом сторонняя организация, осуществляющая поиск плагиата, не сможет получить «читабельное» содержимое исходного документа из передаваемой ей информации. Качество поиска документов в сети Интернет при этом не ухудшается. Предлагаемая архитектура увеличит производительность СОП и уменьшит нагрузку на локальную информационную инфраструктуру пользователя.

В дальнейшем планируется программная реализация предложенной архитектуры СОП и совершенствование ее компонент. Одно из возможных направлений исследований – это включение стилеметрии [7; 8] (определения стиля текста) во внешнюю часть СОП, что позволило бы фильтровать результаты поиска на ранних стадиях, до их загрузки во внутреннюю часть СОП.

Список литературы

1. *Ивойлова И.* Украденные мысли: половина студенческих рефератов и курсовых скачивается из Интернета // Российская газета. 2009. № 4830. 20 янв. URL: http://www.rg.ru/2009/01/20/referaty.html

2. *Jensen L. A., Arnett J. J., Feldman S. S., Cauffman E.* It's Wrong, but Everybody Does It: Academic Dishonesty among High School Students // Contemporary Educational Psychology. 2002. Vol. 27 (2). P. 209–228.

3. *Федотов А. М., Барахнин В. Б.* К вопросу о поиске документов «по аналогии» // Вестн. Новосиб. гос. ун-та. Серия: Информационные технологии. 2009. Т. 7, вып. 4. С. 5–7.

4. *Culwin F., Child M.* Optimizing and Automating the Choice of Search Strings when Investigating Possible Plagiarism. In Proceedings of 4th International Plagiarism Conference. Newcastle, June 2010. URL: http://www.plagiarismadvice.org/documents/conference2010/abstracts/4IPC_0014.pdf

5. *Butakov S., Shcherbinin V.* On the Number of Search Queries Required for Internet Plagiarism Detection // Proc. of the 9th IEEE International Conference on Advanced Learning Technologies (July 15–17, 2009). Vol. 00. Washington, DC: ICALT. IEEE Computer Society, 2009. P. 482–483.

6. *Schleimer S., Wilkerson D., Aiken A.* Winnowing: Local Algorithms for Document Fingerprinting // Proc. of the ACM SIGMOD International Conference on Management of Data. 2003. P. 76–85.

7. *Романов А. С.* Структура программного комплекса для исследования подходов к идентификации авторства текстов // Докл. Том. гос. ун-та систем управления и радиоэлектроники. 2008. № 2 (18), ч. 1. С. 106–109.

8. *Яцко В. А., Стариков М. С., Бутаков А. В.* Автоматическое распознавание жанра и адаптивное реферирование текста // Научно-техническая информация. Сер. 2: Информационные процессы и системы. 2010. № 5. С. 9–18.

Материал поступил в редколлегию 04.03.2011

V. V. Dyagilev, A. A. Tskhay, S. V. Butakov

ARCHITECTURE OF PLAGIARISM DETECTION SERVICE THAT DOES NOT VIOLATE INTELLECTUAL PROPERTY RIGHTS

The paper describes architecture for plagiarism detection service. The proposed novel approach fulfills the copyrights protection requirements for the documents submitted for the check up. The proposed architecture divides the checkup process into two parts. First part utilizes local resources within the organization and another one uses external engine to search documents on the Internet. The executed division maintains the search quality and assures that copyrighted documents cannot be restored from the search requests.

Keywords: Plagiarism Detection, Service Architectures, Copyright Protection.