

## Реализация локального антиплагиата с помощью объектного приложения языка Python

*Кизянов Антон Олегович*

*Приамурский государственный университет имени Шолом-Алейхема  
студент*

*Ленкин Алексей Викторович*

*Приамурский государственный университет имени Шолом-Алейхема  
студент*

*Лучанинов Дмитрий Васильевич*

*Приамурский государственный университет имени Шолом-Алейхема  
старший преподаватель кафедры информационных систем, математики и  
методик преподавания*

### **Аннотация**

В данной статье рассмотрена проблема присвоения текстовой информации, пути ее решения с помощью программных средств. Описана программная реализация локального антиплагиата с помощью объектной среды языка программирования Python.

**Ключевые слова:** Антиплагиат, язык программирования C++, информация, уникальность.

### **The implementation of local Anti-plagiarism using Python object language application**

*Kizyanov Anton Olegovich*

*Sholom-Aleichem Priamursky State University  
student*

*Lenkin Aleksei Viktorovich*

*Sholom-Aleichem Priamursky State University  
student*

*Luchaninov Dmitry Vasilyevich*

*Sholom-Aleichem Priamursky State University  
Senior lecturer of the Department of Information Systems, Mathematics and  
teaching methods*

### **Abstract**

In the article the problem of assigning textual information is considered, ways to solve the problem with the help of software is described. The implementation of

the local Anti-plagiarism software using object-oriented programming language Python is described.

**Keywords:** Anti-plagiarism, the programming language Python, information unique.

В современное время свободный доступ к сети Интернет есть почти у каждого. Развитие информационных технологий в совокупности с повсеместным доступом в сеть делает любую информацию доступной и открытой общественности. К сожалению, это также приносит и новые трудности. В частности, в последнее время появилась и стала актуальной проблема хищения информации, когда материалы, созданные одним человеком, используются без его согласия другими. В целях предотвращения данного процесса были созданы специальные программы для поиска плагиата информации, которые, используя собственную базу источников, проверяют, сколько в проверяемых материалах заимствованных данных и выводят результат, опираясь на который можно делать вывод об уникальности. В данной статье описано исследование процесса поиска плагиата и разработка собственной локальной версии проверки электронных документов.

На данный момент создано уже довольно много программ поиска плагиата в электронных документах, наиболее часто используемая программа принадлежит ЗАО «Анти-Плагиат». Данный продукт широко используется во всех образовательных организациях высшего образования страны для проверки работ студентов, а также научных работ. Алгоритм проверки плагиата хранится компанией в секрете, но общая методика проверки известна [1]:

1. Система собирает информацию из различных источников: загружает из сети Интернет и обрабатывает сайты, находящиеся в открытом доступе, базы научных статей и рефератов. Загруженные документы проходят процедуру фильтрации, в результате которой отбрасывается бесполезная (с точки зрения потенциального цитирования) информация.

2. На следующем этапе каждый из полученных таким образом текстов определённым образом форматируется и заносится в системную базу данных.

3. Все пользовательские документы, загружаемые для проверки, ставятся в очередь на обработку.

4. После проверки документа, пользователь получает доступ к отчёту, в котором представляются результаты.

Данная система во многом и хороша и используется уже достаточный промежуток времени, но она имеет некоторые недостатки [2]:

1. С каждым улучшением программы студенты придумывают новые способы обмана системы. К примеру, раньше было возможно заменить буквы кириллицы на аналогичные латиницы и программа считала такие слова разными.

2. Системе обязательно нужен выход в сеть Интернет для проверки

документа, чтобы сверить его с базой данных.

3. Даже если выход в сеть Интернет есть, проверка электронного документа может затянуться на долгое время, причем дело не в размере проверяемого файла, в обычной (бесплатной) версии на пользователя накладываются некоторые ограничения, а именно: 1 документ на 6 минут и низкий приоритет в очереди на проверку.

Частично решить данную проблему может программа для поиска плагиата в электронных документах, реализованная локально, то есть без подключения к сети Интернет. В ранних исследованиях была разработано программное обеспечение, реализующее данный процесс с помощью языка C++ [3]. В рамках написания данной статьи было разработано приложение, которое, в отличие от программы ЗАО «Анти-Плагиат», использует локальную (оффлайн) базу документов.

В основе данного приложения существует следующий алгоритм проверки документов:

1. Существует два вида файлов – оригинальный, который содержит в себе информацию, совпадение с которой можно оценивать как плагиат, и проверяемый, который вводится для проверки оригинальности.

2. В программу вводятся оригинальный и проверяемый файлы, для проверки оригинальности последовательно проверяются три слова из каждого файла. Если тройки совпадают, все слова из данной тройки считаются плагиатом.

3. В результате работы программы, процент плагиата вычисляется по следующей формуле («количество повторившихся слов»\*100 % / «общее число слов в документе»).

Данное приложение разработано с помощью среды программирования IDLE (Python GUI). Оно имеет минималистичный дизайн, лишенный ненужных опций и предназначено для проверки документов формата docx. Интерфейс программы очень прост: 2 кнопки запроса документа, 2 поля вывода локального адреса документа, 1 кнопка запроса результата. После добавления необходимых файлов на проверку, они соответственно отобразятся в соответствующих полях вывода.

Опишем процесс работы с программой. Для запуска нужно на ярлыке программы (см. рис. 1).



Рисунок 1. Ярлык для запуска программы

Для добавления проверяемых файлов нужно нажать пункт меню «Выбрать файл для проверки» (см. рис. 2). Для проверки файлов нужно нажать кнопку выбора оригинального файла (выбрать файл) и кнопку выбора проверяемого файла.



Рисунок 2. Главная форма программы

После этого необходимо:

1. Убедиться, что все добавленные файлы отобразились в боковых полях.

2. Нажать кнопку «Получить результат».

В результате работы программы снизу отобразится процент плагиата (см. рис. 3).

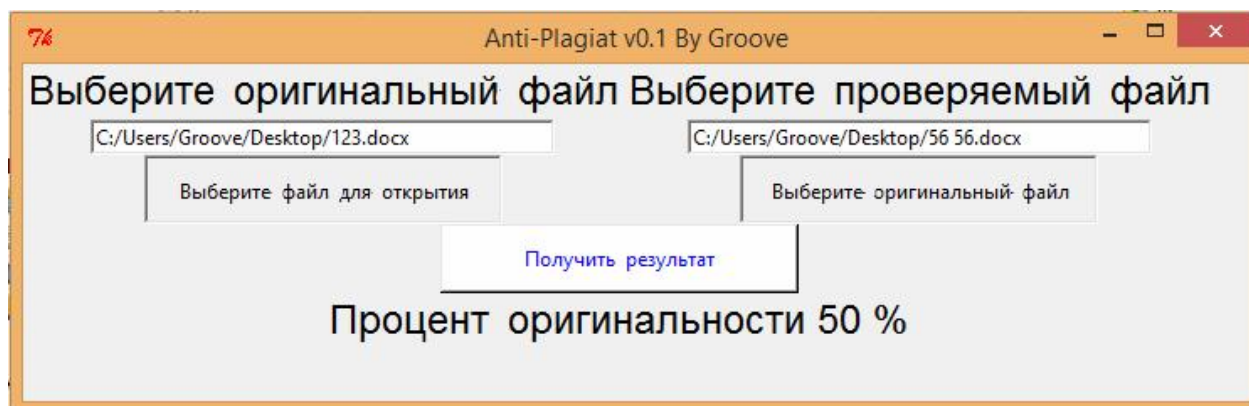


Рисунок 3. Результат работы программы

При этом, для громоздких текстов, скорость выполнения программы не как не подействует вследствие особенностей алгоритма.

Программа использует простой алгоритм проверки и состоит из 2 модулей:

1. Docx2txt – производит адаптацию входного файла, забирая из .docx документа только текст.

2. Tkinter – стандартная библиотека Python для создания разнообразных интерфейсов для программ.

В результате исследования можно описать преимущества и недостатки разработанного программного продукта.

Преимущества программы:

1. Не требуется подключение к сети Интернет для работоспособности.
2. Быстрота проверки текста.
3. Минималистичный дизайн.

#### 4. Кроссплатформенность.

Недостатки программы:

1. Нет проверки изображений и формул, они попросту игнорируются при проверке.

### **Библиографический список**

1. Против рейдерства в науке [Электронный ресурс]. URL: [http://www.chaskor.ru/article/protiv\\_rejderstva\\_v\\_nauke\\_32043](http://www.chaskor.ru/article/protiv_rejderstva_v_nauke_32043) (дата обращения 25.01.2016).
2. Онлайн сервис повышения индивидуальности [Электронный ресурс]. URL: <http://anexr.ru/> (дата обращения 25.01.2016).
3. Ленкин А.В. Колесников А.А. Лучанинов Д.В. Реализация локального антиплагиата с помощью объектного приложения языка C++ // Постулат. 2015. №2. [Электронный ресурс]. URL: <http://e-postulat.ru/index.php/Postulat/article/view/21> (дата обращения 25.01.16)