и выгодным решением, позволяющим планировать, анализировать и контролировать деятельность всех производственных процессов.

Библиографический список

1. Пятковский, О.И. Практикум по дисциплине «Проектирование информационных систем» (в двух частях): учебное пособие / О.И. Пятковский, М.В. Гунер; Алт. гос. техн. ун-т им. И. И. Ползунова. – Барнаул: кафедра ИСЭ, АлтГТУ, 2010.
2. Смирнова Г.Н., Сорокин А.А., Тельнов Ю.Ф. Проектирование экономических информационных систем. - М.: Финансы и статистика, 2001.

УДК 004

Поиск совпадений значений полей в нескольких файлах Excel на основе массива данных NumPy

К.В. Рыбников, О.Н. Половикова
АлтГУ, г. Барнаул

Электронные таблицы повсеместно встречаются в самых разных отраслях экономики. Их применение в сферах бухгалтерии, научно-исследовательской деятельности, автоматизации расчетов и других значительно упрощает работу с данными и позволяет производить вычисления разной степени сложности с высокой эффективностью, при этом уменьшая вероятность ошибок расчетов.

В процессе работы с табличными процессорами часто возникает задача произвести сравнение файлов таблиц со схожей структурой и выявить то, какие имеются изменения между ними. Как правило, это достигается путем нахождения совпадений значений определенного поля таблицы одного файла в другом поле таблицы иного файла. Данная задача усложняется еще и тем, что на практике различные операции над таблицами производятся на большом количестве данных, а это, в свою очередь, требует наличие высокой вычислительной мощности аппаратного обеспечения пользователя и в большинстве случаев приводит к увеличению времени поиска требуемых значений.

Реализация поиска средствами самого табличного процессора через его графический интерфейс занимает определенное время и в случаях, когда необходимо производить многократный поиск значений нескольких полей таблицы, данный подход становится особенно неэффективным и монотонным, при этом повышая риск возникновения

ошибок различного рода. Также стоит добавить, что этот подход нельзя автоматизировать и впоследствии интегрировать в различные автономные системы.

В данном исследовании рассматривается проблема нахождения отличий в данных электронных таблиц схожей структуры через поиск совпадений значений их соответствующих полей.

Главной целью работы является разработка специального программного обеспечения, которое будет производить считывание данных из файлов Excel, предоставлять простой и удобный графический интерфейс для нахождения изменений в файлах и предоставлять отчет о найденных различиях.

В качестве инструмента реализации данной системы был выбран язык программирования Python, поскольку он обладает большой гибкостью и значительными функциональными возможностями. Однако стандартные средства языка Python не подходят для загрузки информации электронных таблиц подобного рода, так как встроенные двухмерные массивы разработаны для общих случаев и не дают оптимальных результатов при работе с большими данными с точки зрения производительности и удобства использования. В связи с этим было решено хранить таблицы в памяти программы в специальных массивах данных библиотеки **NumPy** [1].

Предполагается, что входные данные таблицы имеют поля с уникальными значениями, которые можно использовать для адресации записей. Для тестирования были сгенерированы две таблицы с большим количеством данных, представляющие записи заказов клиентов какого-либо магазина. Фрагмент первой таблицы изображен на рисунке 1.

	A	B	C	D
1	ID клиента	ФИО	Номер продукта	Почтовый индекс
2	1	Щекочихин Касьян Самуилович	62	630176
3	2	Ширяев Даниил Ульянович	3	664038
4	3	Кирьянов Осип Кондратиевич	90	685475
5	4	Котяш Ефим Игнатиевич	65	697235
6	5	Мармазов Михей Ульянович	89	619052
7	6	Тихомиров Еремей Елизарович	36	684383
8	7	Ячменцев Андриян Афанасиевич	47	665042
9	8	Антимонов Эрнст Игнатиевич	77	633888
10	9	Слобожанин Гаврила Богданович	87	626589
11	10	Янкилович Юрий Наумович	61	656067

Рисунок 1 – фрагмент данных первой тестовой таблицы

Вторая таблица по большей части является модифицированной копией первой, где присутствуют изменённые значения некоторых ячеек и добавленные новые записи, а также имеются полностью новые

поля. При этом важно отметить, что первое поле уникальных значений «ID клиента» остается неизменным.

На вход в программу подаются названия файлов обеих таблиц, и далее необходимо задать список индексов полей, которые присутствуют в обеих таблицах и имеют одинаковую последовательность. Беря это во внимание, происходит загрузка указанных полей в массивы данных [2]. Далее, к первой таблице добавляется вспомогательное поле со значением “OLD”, а ко второй добавляется поле со значением “NEW” – это специальные метки, которые в дальнейшем позволят отличить старые значения от новых.

Для того, чтобы найти отличия между двумя таблицами, производится объединение получившихся массивов в один вдоль вертикальной оси. Затем, производя операцию удаления повторяющихся элементов по всем полям, кроме вспомогательного, выявляется множество записей, которые были либо добавлены в новой таблице (и имеют метку “NEW”), либо были изменены (присутствует запись как со старым, так и новым значениями). Применяя аналогичные методы удаления повторяющихся элементов, массив данных разделяется на группу только добавленных записей и на группу только измененных. Наконец, программа сохраняет результаты обеих групп в соответствующие файлы Excel. Пример найденных изменений в записях приведен на рисунке 2.

	A	B	C
1	ID клиента	ФИО	Номер продукта
2	9866.0	Шеин Михей Игнатиевич --> Иванов Михей Игнатиевич	41.0
3	9913.0	Ширяев Степан Капитонович	72.0 --> 70.0
4	9960.0	Антимонов Иван Измаилович	77.0 --> 78.0

Рисунок 2 – результат поиска изменений в записях таблиц

На данном этапе разработанная система может обнаруживать измененные ячейки указанных полей, добавленные записи. Планируется расширить функциональные возможности программы (например, внедрить нахождение удаленных записей), а также сделать графический интерфейс пользователя для более тонкой настройки параметров поиска.

Библиографический список

1. Jake VanderPlas. Python Data Science Handbook: Tools and Techniques for Developers Paperback. – O'Reilly Media, 2016. – 300 с.

2. Robert Johansson. Numerical Python: A Practical Techniques Approach for Industry. – Apress, 2015. – 506 с.