ОЛИМПИАДА ШКОЛЬНИКОВ «ШАГ В БУДУЩЕЕ»
ПО ПРОФИЛЮ «ИНЖЕНЕРНОЕ ДЕЛО»

33520

регистрационный номер

Секция: Системы обработки информации (ИУ5)
название секции

Программа для мониторинга базы данных NCBI на предмет выявления новых штаммов вируса SARS-CoV-2

название работы

Автор: ***Плотников Константин Владимирович***
фамилия, имя, отчество
МБОУ СОШ №7 Г. РЕУТОВА, 10 «А»
наименование учебного заведения, класс

Научный руководитель: **Ланский Сергей Игоревич**
фамилия, имя, отчество
МБОУ СОШ №7 Г. РЕУТОВА
место работы
учитель информатики
звание, должность

_____
_____
подпись научного руководителя

Москва – 2024

Аннотация

Данный проект посвящен созданию программы на языке Python 3.10 для анализа базы данных NCBI Protein на предмет выявления новых штаммов коронавируса SARS-CoV-2 с сильно изменённым поверхностным S-белком (более 16 аминокислотных мутаций относительно ранее распространенных вариантов).

Данный вирус вызывает инфекционное заболевание, названное COVID-19, которое стало причиной пандемии в 2020 году. Этот вирус покрыт «шипиками», состоящими из S-белка, который отвечает за проникновение вируса в клетку и на который, в основном, и формируется иммунный ответ. Скачкообразный комплексный мутагенез S-белка уже имел место в 2022 году и привел к возникновению сильно заразного штамма Омикрон, на который не работал иммунитет ни от вакцины, ни от перенесенного ранее заболевания. На наше счастье, патогенез был не сильно смертельным в тот раз.

Цель создания программы - своевременное оповещение о возникновении штамма вируса с аномально измененным S-белком и как следствие с новыми иммунными и патогенными свойствами.

Предлагаемая программа регулярно (раз в неделю) производит поиск новых опубликованных в базе данных NCBI Protein последовательностей S-белка, удаляет низкокачественные последовательности, сравнивает с обновляемой библиотекой последовательностей и при нахождении кластера последовательностей с высоким количеством мутаций относительно предыдущей библиотеки выдает предупреждение со списком идентификаторов подозрительных последовательностей. Потенциально программа может быть настроена для более широкого перечня вирусов.

Содержание

Введение

Данный проект посвящен созданию программы на языке Python 3.10 для анализа базы данных NCBI Protein на предмет выявления новых штаммов коронавируса SARS-CoV-2 с сильно изменённым поверхностным S-белком. Идея создания такой программы у меня возникла после завершения прошлогоднего проекта по биоинформатики «Выявление наиболее частых мутаций поверхностного S-белка коронавируса SARS-CoV-2 человека, участвующих во взаимодействии с белком-рецептором АПФ-2, и предсказание животных-переносчиков для штамма Омикрон BA.1.», представленного на конференции (конкурс Вернадского, МГУ, Москва, 2022). Чтобы обосновать актуальность создания такой программы позвольте ввести вас в тематику поставленной задачи.

Обзор литературы

База данных NCBI имеет задачу хранить и облегчать доступ к различной биологической информации. В нашем случае нас интересуют белковые последовательности, хранящиеся в NCBI Proteins (GenPept). Белки — биологические молекулы, состоящие из различных аминокислот и выполняющие очень разные функции, включая ферментативную, рецепторную (лиганд-связывающую) и структурную. Информацию о первичной структуре (цепочке аминокислот) белка можно записать в виде последовательности аминокислотных остатков (основных 20 видов) в направлении от N- к С-концу. Каждая аминокислота имеет однобуквенное обозначение. Например: аланин -A, валин -V и т. д. [1]

Целевым объектом для нашей программы являются аминокислотные последовательности поверхностного белка коронавируса SARS-CoV-2, представленные в базе данных NCBI Proteins. Коронавирус SARS-CoV-2, вызывает инфекционное заболевание, которое было названо COVID-19 (coronavirus infectious disease 2019) [2]. В 2020 году это заболевание стало

причиной мировой эпидемии.

Вирусный белок, который привлек наше внимание, имеет название S-белок (surface glycoprotein/поверхностный гликопротеин) или шипик (spike). Этот белок покрывает вирусную частицу, как следствие является основной целью иммунного ответа организма хозяина, и, что самое важное, взаимодействуя с белком рецептором АПФ-2 (ангиотензинпревращающий фермент 2) на поверхности клеток хозяина, запускает цикл событий, приводящих к проникновению вируса в клетку и его размножению. Специфичность S-белкового взаимодействия с белком рецептором АПФ-2 является одним из важных факторов, влияющих на видоспецифичность вируса [3]. Однако мутации, приводящие к заменам одних аминокислот на другие или их удалениям, или лишним вставкам, способны изменять характер взаимодействия между этими белками и привести к изменению видоспецифичности коронавируса. Уже не раз в прессе упоминается заражение животных человеческим коронавирусом [4]. Более того мутации могут приводить к тому, что сформированный ранее (в следствии вакцинации или перенесенного заболевания) иммунитет не сможет эффективно противостоять новым вариантам этого же вируса.

Мутагенез вируса происходит постоянно. Это воочию можно наблюдать, анализируя базы данных последовательностей РНК и белков вируса. Выбранная нами база данных NCBI постоянно обновляется, добавлением новых последовательностей, которые возникают в следствии секвинирования клинических изолятов, которые в свою очередь берут у больных пациентов во всем мире. Каждому изоляту присваиваться уникальный идентификатор, по которому можно получить исчерпывающую информацию о времени и месте забора образца. Совокупность изолятов со сходными последовательностями и общим происхождением образуют штамм вируса (то есть его разновидность). Например, широко нашумевшие в свое время штаммы Альфа, Бета и Дельта [5].

Актуальность

Последнее время волны заболеваний вирусом SARS-CoV-2 вызываются в основном подвариантами Омикрон штамма [6], который практически полностью вытеснил уже 2022 году другие штаммы, в следствии своей высокой заразности и способности обходить сформированный ранее (в следствии вакцинации или перенесенного заболевания) иммунитет [7]. На наше счастье, этот штамм, в отличии от более раннего штамма Дельта, «пощадил» нас, более легким течением заболевания и более низкой смертностью, отодвинув тем самым пандемию в ряду глобальных страхов человечества на несколько позиций вниз. «Удивительным» является то, что этот штамм содержит сразу 15 мутаций в рецептор связывающем домене S-белка, относительно исходного штамма из Китайского города Ухань. Сравнение от предыдущего фаворита - дельты от Уханя. Это очень существенное в плане генетики и очень принципиальное в плане последствий отличие нового штамма от предшественников. Такой комплексный мутагенез без промежуточных форм очень маловероятен. Самое простое и очевидное объяснение — это то, что мутагенез прошел либо в популяции животных переносчиков, а потом новый штамм вируса снова заразил человека, либо, что он был специально генетически изменен в биолаборатории и запущен в популяцию людей. В любом случае сам факт того, что вирус может изменяться не постепенно (эволюционировать), как это обычно происходит в следствии накопления мутаций в рамках одной популяции, а скачкообразно, сразу меняя много аминокислот в принципиальных для функционирования S-белка позициях (а следовательно, и для патогенеза всего вируса в целом) вызывает чувство настороженности по отношению к данному явлению. Особенно в рамках нынешней геополитической ситуации в мире. В связи с этим, считаю актуальным создание программы для мониторинга базы данных NCBI Protein на предмет выявления новых штаммов коронавируса SARS-CoV-2 с сильно изменённым поверхностным S-белком, относительно предшествующих вариантов.

Цель создания программы - своевременное оповещение о возникновении штамма вируса с аномально измененным S-белком и как следствие с новыми иммунными и патогенными свойствами.

Задачи работы

1. Сформулировать критерии для объединения последовательностей в группы внутри библиотеки и для отнесения к аномально измененным - для дальнейшего оповещения об угрозе.
2. Написать модуль загрузки последовательностей S-белков коронавируса SARS-CoV-2 из базы данных NCBI Proteins с функцией фильтрации от низкокачественных последовательностей.
3. Написать функцию сравнения последовательностей.
4. Создать модуль загрузки, обновления и сохранения локальных библиотек.
5. Настроить оповещения через Telegram.

Методы

Средство программной реализации — язык программирования Python 3.10 с использованием среды разработки Microsoft Visual Studio.

Для работы с последовательностями использовалась библиотека Bio и информационный ресурс по работе с ней [8].

В частности, использовались ее модули:

- модуль Align — для сравнения последовательностей;
- модуль Entrez — для взаимодействия с библиотекой NCBI;
- модуль SeqIO — для парсинга формата Fasta.

Для запуска программы с определенной периодичностью использовалась библиотека schedule, а для сопряжения с Телеграмм - библиотека telebot.

Основная часть работы

Критерии для объединения последовательностей в группы внутри библиотеки и для отнесения к аномально измененным

Для оптимизации программы было принято решение все последовательности разделить на группы. В качестве критерия использовалось расхождение относительно рута группы не менее 6 мутаций. Если меньше, то последовательность относим к группе, в противном случае создаём новую, где рутом будет являться этот сиквенс. Если группа отличается сразу на 18 и более мутаций от рута, то она считается аномальная группа, для которой необходимо дальнейшее оповещения об угрозе. Стоит заметить, что используется принцип приоритетности, то есть если последовательность, отличающаяся на 5, он будет добавлена в 0 группу, а не в 1. На основе последовательностей штаммов коронавируса по классификации в портале CoVariants.org мы составили таблицу схожести последовательностей S-белков этих штаммов (приложение рис. 1). Из рисунка видно, что подварианты Омикрон штамма коронавируса, который практически полностью вытеснил предшествующие штаммы, в следствии своей высокой заразности и способности обходить сформированный ранее иммунитет, содержали в S-белка более чем 27 дополнительных мутаций. Хотим с акцентировать внимание, что в данной работе под мутацией мы подразумеваем замену, делецию или вставку в аминокислотной последовательности белковой молекулы.

Загрузки последовательностей S-белков коронавируса SARS-CoV-2 из базы данных NCBI Proteins

На следующем этапе работы необходимо было написать модуль загрузки последовательностей S-белков коронавируса SARS-CoV-2 из базы данных NCBI Proteins с функцией фильтрации от низкокачественных последовательностей (приложение рис. 2).

В самом начале работы с базой данных NCBI нужно указать адрес электронной почты для обратной связи (приложение рис. 2 А). Это делается при

помощи модуля Entrez.email библиотеки Biopython.

Следующим шагом необходимо обозначить временной диапазон. При помощи библиотеки datetime задаём переменную **today** и **one_week_ago**. Обе переменные переводим в формат строки при помощи метода strftime.

Далее формируем поисковой запрос: название белка, название организма и диапазон дат публикации. Производим поиск ID белков по запросу. По найденным ID загружаем последовательности и очищаем их от низкокачественных образцов при помощи нашей функции фильтрации — delete_bad_sequences (приложение рис 2 Б).

Функции фильтрации на вход подаётся максимальное количество неизвестных аминокислот, минимальная длина и список последовательностей. Циклом for функция проверяет каждую последовательность и, если она удовлетворяет критериям, добавляет в новый список. В конце работы функции новый список возвращается.

Модуль сравнения последовательностей

Следующим модулем программы является функция сравнения последовательностей. Эта функция получает на вход одну последовательность из скаченного списка и одну из локальной библиотеки. При помощи модуля Align библиотеки Bio производится сравнение, но результат не подходящего формата (приложение рис. 3 А). Нам нужно получить количество несоответствий. Для этого удаляем все строки содержащие буквы, в оставшихся строках удаляем все цифры. Далее подсчитываться количество несовпадений «.» и отдельных делеций/инсерций «-» (приложение рис. 3 Б).

Модуль загрузки, обновления и сохранения локальных библиотек

Основным телом программы является модуль загрузки, обновления и сохранения локальных библиотек, которая состоит из функций main_alignment, scan_seqs и reflib_update.

Функция main_alignment в начале своей работы создает резервные копии библиотек ref_lib и ref_lib_bad, а сами данные библиотеки из файла переводятся

в формат многоуровневых списков (приложение рис. 4). Далее вызывается функция scan_seqs (приложение рис. 5). В самом конце работы этого модуля формируется отчёт о новых или быстрорастущих аномальных группах последовательностей (приложение рис. 4), который будет отправлен телеграмм ботом.

Рассмотрим по подробнее функцию **scan_seqs** (приложение рис. 5). Эта программа последовательно направляет новые загруженные и отфильтрованные последовательности в функцию reflib_update. В конце своей работы выполняется перераспределение групп последовательностей на основе их актуальности и размеру каждой группы между библиотеками ref_lib и ref_lib_bad, а также сохранение этих обновленных библиотек.

Вызываемая функция reflib_update (приложение рис. 6) на входе должна получить две библиотеки и одну новую последовательность, которую ей подает функция scan_seqs. Эта последовательность сравнивается с каждым элементом библиотек ref_lib и ref_lib_bad с помощью ранее упомянутой функции сравнения до тех пор, пока не найдётся группа отличающееся менее чем на 6. Если минимальное расхождение меньше 6, то id новой последовательности добавляется к первой группе (всего в памяти группе содержится до 50 id последовательностей), с которой получился такой результат сравнения. В случае, если новая последовательность отличается от всех предшествовавших групп на 6 и более мутаций создаётся новая группа в библиотеке ref_lib, в которую добавляется сама последовательность.

Формирование актуальных локальных библиотек

После того, как мы смогли наладить работоспособность программы настало время для формирование актуальных библиотек последовательностей S-белка коронавируса. Для этого использовалась функция main_alignment с небольшими надстройками в виде порционной подачи последовательностей (от 2000 до 5000). Были проанализированы все последовательности за последние 3 года, начиная от уханьского штамма. Всего было обработано 2440081 высококачественных последовательностей, на основе которых и были

сформированы актуальные библиотеки ref_lib и ref_lib_bad с одинаковой архитектурой (приложение рис. 7). Первая содержит актуальные и часто встречающиеся группы, а вторая – редкие и устарелые. Группы могут перемещаться между библиотеками, за счет второй половины кода функции scan_seqs (приложение рис. 5). Распределение по библиотекам зависит от количества последовательностей в группе и последнего обновления группы. Такое разделение позволило ускорить работу программы.

Результаты работы программы

Важно отметить, что в результате формирования актуальных библиотек было выявлено 14 аномальных групп, среди которых стоит выделить три — группы 689, 847 и 1759, так как эти группы каждая включает более 1000 последовательностей, в то время как остальные группы имеют минимальный размер (приложение рис. 8). Группа 689 соответствует подварианту BA.1 штамма Омикрон, который вызвал первую волну заболеваемости Омикрон штаммов в конце 2021-начале 2022 годов. Группа 847, соответствует подварианту BA.2 штамма Омикрон, который вызвал следующую волну заболеваемости в начале-середине 2022 года. А группа 1759, которая соответствует подварианту BA.2.86 штамма Омикрон, известный также как штамм Пирола, является основным штаммом, вызвавшим последнюю волну заболеваемости начиная с ноября 2023.

Оповещения через Telegram

Для удобства удаленного пользования написанной программой было решено сопряжить ее с ботом Telegram (приложение рис. 9).

Для работы бота нужно задать уникальный токен. Далее, при помощи метода message_handler, программа может получить команду из чата, если была написана команда «start», то надо выполнить одноимённую функцию. Функция start отправляет оповещение о начале работы и при помощи модуля schedule вызывает основную программу main.

Для отправки оповещения пришлось немного переделать основное тело программы, в ней добавили глобальную переменную report, чтобы не

передавать ни ID сообщения, ни сам репорт. Сразу после выполнения функции main_alignment запускается проверка: если репорт пуст, то ничего отправлять не надо, в противном случае – отправить репорт.

Выводы

1. Нами создана и запущена в работу программа своевременного оповещения о возникновении штамма вируса с аномально измененным S-белком.
2. Результаты анализа нашей программой базы данных сиквенсов прошлых лет, продемонстрировали, что она способна выявлять и оповещать об опасных штаммах вируса SARS-CoV-2, которые уже вызвали новые волны эпидемии.

Список использованных источников

1. Грин Н., Стаут У., Тейлор Д., «Биология в 3-х томах.» том1, 1993, Москва «Мир»
2. Википедия. Ковид-19. 2021 г. https://ru.wikipedia.org/wiki/COVID-19
3. Damas J., Hughes G.M., Keough K.C., et. al. (2020) “Broad host range of SARS-CoV-2 predicted by comparative and structural analysis of ACE2 in vertebrates.” Proc Natl Acad Sci U S A 117(36):22311-22322. doi: 10.1073/pnas.2010146117.
4. BBC News/Русская Служба. “Норок в Дании истребили из-за Covid-19. Но будущее индустрии меха и до пандемии было под вопросом”. 24 ноября 2020 года. https://www.bbc.com/russian/features-55061282.
5. Cocherie T, Zafilaza K, Leducq V, Marot S, Calvez V, Marcelin AG, Todesco E (2022) Epidemiology and Characteristics of SARS-CoV-2 Variants of Concern: The Impacts of the Spike Mutations. Microorganisms. 22;11(1):30. doi: 10.3390/microorganisms11010030.
6. Sabbatucci M, Vitiello A, Clemente S, Zovi A, Boccellino M, Ferrara F, Cimmino C, Langella R, Ponzo A, Stefanelli P, Rezza G (2023) Omicron variant evolution on vaccines and monoclonal antibodies. Inflammopharmacology. 31(4):1779-1788. doi: 10.1007/s10787-023-01253-6.
7. Wang SC, Rai CI, Chen YC (2023) Challenges and Recent Advancements in COVID-19 Vaccines. Microorganisms. 11(3):787. doi: 10.3390/microorganisms11030787.
8. Biopython Tutorial and CookbookJeff Chang, Brad Chapman, Iddo Friedberg, Thomas Hamelryck, Michiel de Hoon, Peter Cock, Tiago Antao, Eric Talevich, Bartek WilczyńskiLast Update – January 10, 2024 (Biopython 1.83) https://biopython.org/DIST/docs/tutorial/Tutorial.html

Приложение

Рис.1. Сравнение S-белков у разных штаммов SARS-CoV-2 из базы Covariants: зеленым цветом обозначено расхождение менее 6 мутаций, красным — более 17 мутаций, а белым промежуточное показание по количеству мутаций. Видно, что штаммы Омикрон сильно отличаются от более ранних штаммов.

```python
def download(day_delta, X_max, Len):

    print("Загрузка последовательностей")
    Entrez.email = "vap.kosta@gmail.com"

    # Временной диапазон.
    today = datetime.date.today()
    one_week_ago = (today - datetime.timedelta(days=day_delta)).strftime("%Y/%m/%d")
    today = today.strftime("%Y/%m/%d")
    # поиск последовательностей.
    search_term = f"surface glycoprotein AND Severe acute respiratory syndrome coronavirus 2[Orgn] AND {one_week_ago}:{today}[Date - Publication]"
    search_result = Entrez.read(Entrez.esearch(db="protein", term=search_term, retmax=10000000))
    protein_ids = search_result["IdList"]
    # загруска последовательностей
    fetch_handle = Entrez.efetch(db="protein", id=protein_ids, rettype="gb", retmode="text")
    sequences = delete_bad_sequences(X_max, Len, list(SeqIO.parse(fetch_handle, "genbank")))
    fetch_handle.close()

    print(f"Скачано {len(sequences)} последовательностей SPIKE белков за последнюю неделю.")
    return sequences
```

Рис 2 А. Модуль загрузки последовательностей.

```python
def delete_bad_sequences(X_max, Len, sequences):
    newlist = []
    for i in sequences:
        if len(i) >= Len and (i.seq).count("X") <= X_max:
            newlist.append(i)
    return newlist
```

Рис 2 Б. Функция фильтрации для модуля загрузки.

А

```
target          120 NNATNVVIKVCEFQFCNDPFLGVYYHKNNKSWMESEFRVYSSANNCTFEYVSQPFLMDLE
                120 ||||||||||||||||||||||.|-|.||||||||||.||||||||||||||||||||||
query           117 NNATNVVIKVCEFQFCNDPFLDV-YQKNNKSWMESELRVYSSANNCTFEYVSQPFLMDLE
```

Б

```python
def alignment_results_short(target, query):
    aligner = Align.PairwiseAligner(scoring='blastp')
    aligner.mode ='global'
    alignments = aligner.align(target,query)
    align_str = str(alignments[0])
    misses = align_str.count('.')
    collapsed_str = ''
    align_split = align_str.split('\n')
    for i in range(len(align_split)//4):
        collapsed_str += align_split[4*i+1]
    collapsed_non_num = collapsed_str.replace(' ', '')
    for i in range(10):
        collapsed_non_num = collapsed_non_num.replace(str(i), '')
    collapsed_gaps = collapsed_non_num.replace('-----', '-')
    collapsed_gaps = collapsed_gaps.replace('----', '-')
    collapsed_gaps = collapsed_gaps.replace('---', '-')
    collapsed_gaps = collapsed_gaps.replace('--', '-')
    contiguous_gaps = collapsed_gaps.count('-')
    results = misses + contiguous_gaps
    return results
```

Подсчёт несовпадений

Удаление букв

Удаление цифр

Подсчёт делеций

Рис.3 Функция сравнения последовательностей. А. Данные до форматирования. Б. Код функции для форматирования.

```python
def main_alignment(sequences):
    global report
    report = ""
    new_name = copy_reflib()
    copy_reflib_bad()
    ref_lib = reflib_to_list()
    ref_lib_bad = reflib_bad_to_list()
    x = len(ref_lib) + len(ref_lib_bad) - 1
    scan_seqs(ref_lib, ref_lib_bad, sequences)
    warning = []
    warn_new = []
    for i in range(len(ref_lib)):
        if (int(ref_lib[i][0]) > x) and (ref_lib[i][6] >= 16):
            warn_new.append('Group #' + str(ref_lib[i][0]) + ' with ' +str(ref_lib[i][6]) + ' mutations from ' + ref_lib[i][5])
    warn_grow = []
    for i in range(len(ref_lib)):
        if ref_lib[i][6] >= 16:
            if ref_lib[i][2]-ref_lib[i][3] >= 3:
                warn_grow.append('Group #' + str(ref_lib[i][0]) + ' with ' +str(ref_lib[i][6]) + ' mutations from ' + ref_lib[i][5]
    if len(warn_new)+len(warn_grow) > 0:
        warns = open(f'new_warnings_{new_name}', 'w', encoding='utf-8')
        if print_reports: print(f'Warning list was generated. See file new_warnings_{new_name}')
        if warn_new != []:
            warns.write('New strange variants:\n')
            for i in range(len(warn_new)):
                warns.write(warn_new[i]+'\n')
        if warn_grow != []:
            warns.write('Growing strange variants:\n')
            for i in range(len(warn_grow)):
                warns.write(warn_grow[i]+'\n')
        warns.close()
        warns = open(f'new_warnings_{new_name}', 'r', encoding='utf-8')
        #print(warns.read())

        report = str(warns.read())
        warns.close()
```

Рис. 4. Модуль загрузки, обновления и сохранения локальных библиотек. (main_alignment).

```python
def scan_seqs(ref_lib, ref_lib_bad, seq_file):
    nume = 0
    for record in seq_file:
        reflib_update(ref_lib, ref_lib_bad, record)
        nume += 1
        print("*" + str(nume))
    if print_reports: print(f'{datetime.datetime.today().strftime("%y%m%d_%H%M%S")}: analysis completed')
    ref_lib_raw = open('reflib_latest.txt', 'w', encoding= 'utf-8')
    ref_lib_bad_raw = open('reflib_bad_latest.txt', 'w', encoding= 'utf-8')
    for_write = []
    for_write_bad = []
    for i in range(len(ref_lib)):
        members = ' '.join(ref_lib[i][7])
        lineZ = ['>Group'+str(ref_lib[i][0]), ref_lib[i][1], str(ref_lib[i][2]), str(ref_lib[i][3]), str(ref_lib[
        #print(lineZ)
        if (int(lineZ[2]) < 10) and ((int(lineZ[4])) >= 3): for_write_bad.append(' '.join(lineZ))
        elif (int(lineZ[2]) < 32) and ((int(lineZ[4])) >= 4): for_write_bad.append(' '.join(lineZ))
        elif (int(lineZ[2]) < 100) and ((int(lineZ[4])) >= 5): for_write_bad.append(' '.join(lineZ))
        elif (int(lineZ[2]) < 315) and ((int(lineZ[4])) >= 6): for_write_bad.append(' '.join(lineZ))
        elif (int(lineZ[2]) < 1000) and ((int(lineZ[4])) >= 7): for_write_bad.append(' '.join(lineZ))
        elif (int(lineZ[2]) < 3150) and ((int(lineZ[4])) >= 8): for_write_bad.append(' '.join(lineZ))
        elif (int(lineZ[2]) < 10000) and ((int(lineZ[4])) >= 9): for_write_bad.append(' '.join(lineZ))
        elif (int(lineZ[2]) < 31500) and ((int(lineZ[4])) >= 10): for_write_bad.append(' '.join(lineZ))
        elif (int(lineZ[2]) < 100000) and ((int(lineZ[4])) >= 11): for_write_bad.append(' '.join(lineZ))
        else: for_write.append(' '.join(lineZ))
    for i in range(len(ref_lib_bad)):
        members = ' '.join(ref_lib_bad[i][7])
        lineZ = [str('>Group'+ str(ref_lib_bad[i][0])), ref_lib_bad[i][1], str(ref_lib_bad[i][2]), str(ref_lib_bad
        #print(lineZ)
        if (int(lineZ[2]) > 25) and ((int(lineZ[4])) <= 5):
            for_write.append(' '.join(lineZ))
        else: for_write_bad.append(' '.join(lineZ))
    ref_lib_raw.write('\n'.join(for_write))
    ref_lib_raw.close()
    ref_lib_bad_raw.write('\n'.join(for_write_bad))
    ref_lib_bad_raw.close()
    #if print_reports: print(iter)
    if print_reports: print(f'{datetime.datetime.today().strftime("%y%m%d_%H%M%S")}: reflib was updated based on r
```

Обновление reflib

Критерии переноса группы из библиотеки reflib в reflib_bad

Возврат из reflib_bad в reflib

Сохранение библиотек

Рис.5 Модуль загрузки, обновления и сохранения локальных библиотек. (scan_seqs)


```
def reflib_update(ref_lib, ref_lib_bad, seqrecord):
    latest_unique_number = len(ref_lib)+len(ref_lib_bad)-1
    nearest = ['none', 1000]
    if print_curren_seq_ID: print(str(datetime.datetime.today().strftime("%y%m%d_%H%M%S")) + ': id ' + str(seqrecord.id))
    for i in range(len(ref_lib)):
        align_diff = alignment_results_short(seqrecord.seq, ref_lib[i][1])
        if align_diff < criteria[0]:
            ref_lib[i][2] += 1
            ref_lib[i][4] = 0
            if len(ref_lib[i][7]) < 51:
                ref_lib[i][7].append(str(seqrecord.id))
            break
        if nearest[1] > align_diff:
            nearest[0] = ref_lib[i][0]
            nearest[1] = align_diff
        if i == len(ref_lib)-1:
            for j in range(len(ref_lib_bad)):
                align_diff = alignment_results_short(seqrecord.seq, ref_lib_bad[j][1])
                if align_diff < criteria[0]:
                    ref_lib_bad[j][2] += 1
                    ref_lib_bad[j][4] = 0
                    if len(ref_lib_bad[j][7]) < 51:
                        ref_lib_bad[j][7].append(str(seqrecord.id))
                    break
                if nearest[1] > align_diff:
                    nearest[0] = ref_lib_bad[j][0]
                    nearest[1] = align_diff
                if j == len(ref_lib_bad)-1:
                    new_ref = []
                    new_ref.append(str(latest_unique_number))
                    new_ref.append(str(seqrecord.seq))
                    new_ref.append(1)
                    new_ref.append(1)
                    new_ref.append(0)
                    new_ref.append('Group' + str(nearest[0]))
                    new_ref.append(nearest[1])
                    new_ref.append([str(seqrecord.id)])
                    ref_lib.append(new_ref)
```

Рис. 6. Модуль загрузки, обновления и сохранения локальных библиотек. (reflib_update).

Структура библиотеки reflib и reflib_bad:

```
[[group_no, sequence, memb_now, memb_earlier, idle_runs, root, diff_from_root, [memb1_id, ...,
member50max_id],[...],...]
[
[>Group0,  MFVF…KLHYT, 180681, 180643,  0,  Unknown,  0,  [YP_009724390.1, … , QKE43667.1]],
[>Group44, MFVF…KLHYT, 143742, 143737,  0,  Group0,   9,  [QQH18545.1, … , QQX36209.1]]
]
```

Рис. 7. Структура библиотек reflib и reflib_bad.

Имя Группы	Кол-во	Родитель	Отличие от род.	ID первого члена группы	Синоним в литературе
Group27	1	Group0	41	QOC89639.1	н.к.
Group12	1	Group0	35	QOD59283.1	н.к.
Group689	266682	Group103	31	UFO69279.1	BA.1
Group1759	18425	Group1249	29	WLW39834.1	BA.2.86
Group1782	1	Group818	28	WMQ65998.1	н.к.
Group11	1	Group0	22	QOD59282.1	н.к.
Group847	637769	Group689	21	UHU97100.1	BA.2.9
Group36	1	Group0	21	QOJ75924.1	н.к.
Group6	1	Group0	20	QOE84223.1	н.к.
Group32	1	Group0	20	QOJ86685.1	н.к.
Group64	1	Group29	19	QRA18925.1	н.к.
Group1596	1	Group798	19	WDY97105.1	BA.1.1
Group42	1	Group0	18	QOQ53339.1	н.к.
Group1523	1	Group847	18	WCI23486.1	BA.2.12.1

Рис. 8. Группы с аномально большим отличием от родительской группы, выявленные при формировании библиотек

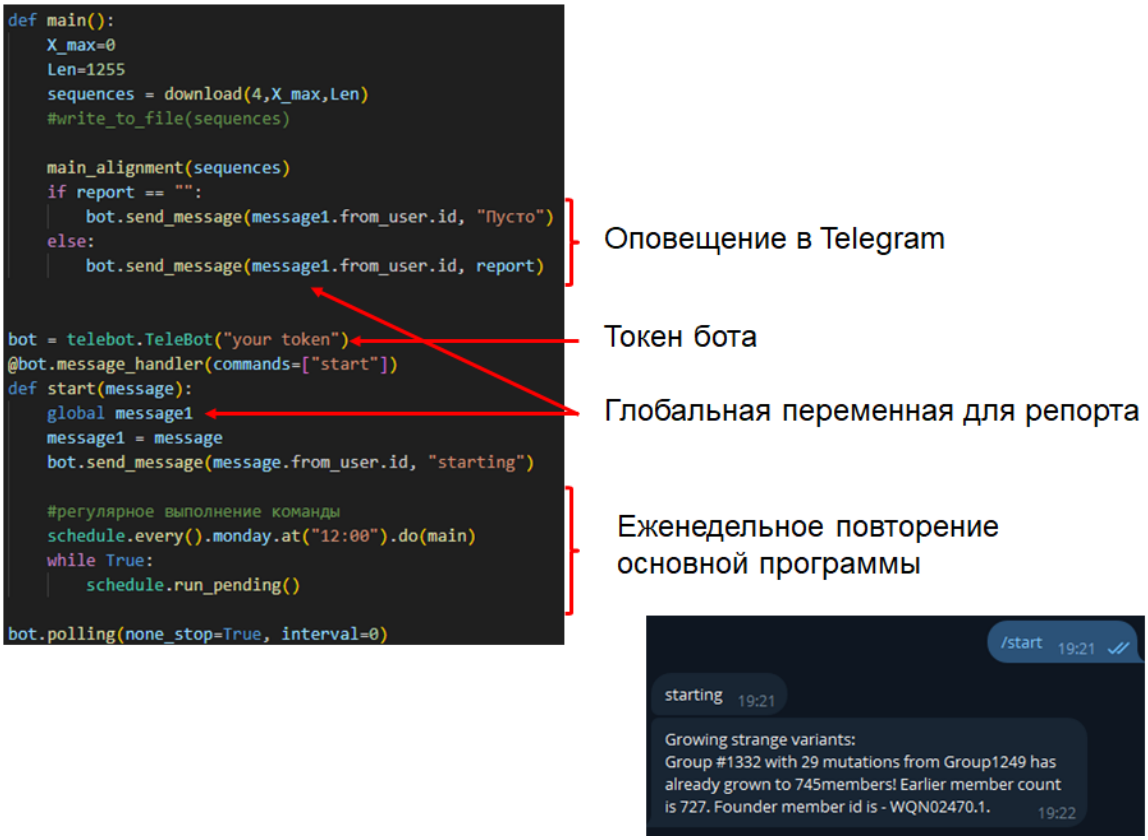

Рис. 9. Оповещение через Telegram.