УДК 004.8

ПРИМЕНЕНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ В ЗАДАЧЕ СЕКВЕНИРОВАНИЯ ГЕНОМА

Смагин Василий Денисович[1], Русакович Артем Николаевич[2]

[1]Ведущий инженер-программист;
ОАО ОКБ «Аэрокосмические системы»;
Россия, 141983, Московская область, г. Дубна, ул. Программистов, д. 4;
Старший преподаватель;
Государственный университет «Дубна»;
Россия, 141980, Московская обл., г. Дубна, ул. Университетская, д. 19;
e-mail: smagin.vasily@uni-dubna.ru.

[2]Младший научный сотрудник;
Группа молекулярной и радиационной генетики ЛЯП ОИЯИ;
Россия, 141983, Московская область, г. Дубна, ул. Жолио-Кюри, д. 6;
e-mail: arusakovich@jinr.ru.

Статья посвящена методам машинного обучения, применяемым для улучшения анализа результатов секвенирования. Предметом данного исследования является анализ существующих подходов к обработке молекулярно-биологических (МБ) данных, полученных с помощью различных методик секвенирования, с использованием машинного обучения (МО). Цель работы – обобщение актуальных способов получения «полезной» информации из «сырых» МБ данных. Секвенирование применяется для установления последовательности нуклеотидов в ДНК и является одной из наиболее важных процедур в рамках геномных исследований. В процессе проведения NGS (Next-Generation Sequencing) требуется обрабатывать колоссальные объемы данных, зачастую имеющих дефекты разного рода. В связи с существованием различных вариаций методов секвенирования и наличием в них трех и более этапов, спектр проблем решаемых с помощью МО в этой области также чрезвычайно широк. В работе представлен краткий обзор некоторых решений, основанных на машинном обучении, применяемых для повышения качества анализа и преобразования результатов отдельных этапов секвенирования. Описаны ключевые группы биоинформатических задач в рамках секвенирования, приведены примеры реализованных алгоритмов с использованием МО. Кроме того, разработаны различные подходы к решению одной и той же задачи, при этом имеющие свои преимущества и недостатки.

Ключевые слова: секвенирование, NGS, машинное обучения, сборка генома.

Для цитирования:
Смагин В. Д., Русакович А. Н. Применение методов машинного обучения в задаче секвенирования генома // Системный анализ в науке и образовании: сетевое научное издание. 2021. № 1. С. 23–32. URL : http://sanse.ru/download/423.

MACHINE LEARNING METHODS APPLICATIONS FOR GENOME SEQUENCING

Smagin Vasiliy[1], Rusakovich Artem[2]

[1]Senior software engineer;
JSC EDB «Aerospace systems»;
4 Programmistov Str., Dubna, Moscow region, 141983, Russia;
Senior teacher;
Dubna State University,
19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;
e-mail: smagin.vasily@uni-dubna.ru.

[2]Fellow scientist;
Joint institute for nuclear researches
Molecular and radiational genetics group;
6 Joliot-Curie Str., Dubna, Moscow region, 141980, Russia;
e-mail: arusakovich@jinr.ru.

The article is dedicated to the machine learning methods used to improve the analysis of sequencing results. The subject of this study is the analysis of the existing approaches to the processing of molecular-biological (MB) data obtained with the help of various sequencing techniques, using machine learning (ML). The purpose of the work is to generalize the current methods of obtaining "useful" information from " raw " MB data. Sequencing is used to establish the sequence of nucleotides in DNA and is one of the most important procedures within the framework of genomic research. In the process of conducting NGS (Next-Generation Sequencing), it is necessary to process huge amounts of data, often with various kinds of defects. Due to the existence of different variations of sequencing methods and the presence of three or more stages there, the range of the problems solved with the help of ML in this area is also extremely wide. The article provides a brief overview of some solutions based on machine learning and used to improve the quality of analysis and transform the results of individual stages of sequencing. The key groups of bioinformatics tasks in the framework of sequencing are described, and the examples of the implemented algorithms using ML are given. In addition, the different approaches to solving the same problem have been developed and at the same time they have their own advantages and disadvantages.

Keywords: sequencing, NGS, machine learning, genome assembly.

For citation:
Smagin V., Rusakovich A. Machine learning methods applications for genome sequencing. System Analysis in Science and Education, 2021;(1):23–32(In Russ). Available from: http://sanse.ru/download/423.

Введение

Развитие науки и информационных технологий привело к появлению новых междисциплинарных областей. Одной из них является биоинформатика, т.е. использование компьютерных, математических и статистических методов для решения различных биологических задач. На сегодняшний момент ученые-биологи имеют дело с огромными объемами данных, которые невозможно обработать вручную, в то время как число актуальных вопросов и целей растет вместе с увеличением известной информации. Достижения биоинформатики широко применяются в медицине и других биологических дисциплинах, особенно в геномике.

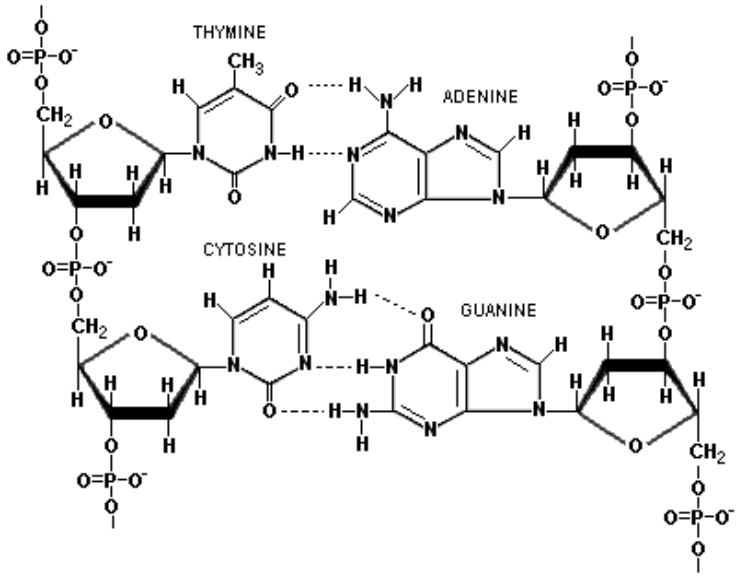

Рис. 1. Молекулярная структура ДНК

Дезоксирибонуклеиновая кислота (ДНК) является макромолекулой, хранящей и реализующей биологическую информацию. Эта информация хранится в виде последовательности нуклеотидов, всего существует четыре вида нуклеотидов.

Одной из важнейших задач биоинформатики является изучение структуры генома, решение которой невозможно без процедуры секвенирования. Секвенирование – определение нуклеотидной последовательности в белках и нуклеиновых кислотах. Существует множество применений для этих

данных: они полезны при работе с организмами, используемыми в научных исследованиях; для прогнозирования особенностей белков; для изучения генома человека, наследственных заболеваний и т.д. На сегодняшний день существует спектр методов секвенирования, каждый из которых имеет свои преимущества и недостатки перед другими и эффективность их применения зависит от решаемой задачи.

1. Методы и этапы секвенирования

Традиционным методом, который применяется на протяжении более 40 лет, является секвенирование ДНК по Сэнгеру. На текущий момент оно полностью автоматизировано и проводится на приборах, называемых секвенаторами. Результаты представляют собой спектрограмму и анализируются с помощью компьютера. Если лабораторные этапы выполнены качественно данные представляют собой последовательность разноцветных пиков, соответствующих четырем нуклеотидам.

Секвенаторы такого типа могут «прочитывать» за один раз последовательности длиной 500-1000 нуклеотидов. Данный метод предоставляет результаты высокой надежности, а его автоматизация значительно ускорила процесс секвенирования и позволила осуществить секвенирования целых геномов, включая геном человека (“*Human Genome Project*”). Однако использование данного метода не позволяет быстро обрабатывать длинные фрагменты ДНК. Отсюда следует задача большей автоматизации анализа, увеличения объема получаемой информации и снижения стоимости исследований.

Для достижения этих целей были разработаны методы секвенирование нового поколения (*next generation sequencing, NGS*). *NGS* – это группа методов, технология которых основана на массовом одновременном секвенировании множества фрагментов ДНК на базе заранее подготовленных коллекций коротких однонитевых фрагментов ДНК (т.н. библиотек). Общий подход в *NGS* заключается в том, чтобы параллельно и массово провести процедуру секвенирования отдельных небольших участков ДНК, после чего программно собрать эти небольшие участки в одну цепь. Каждая платформа имеет свои особенности и отличия. Две наиболее известные и востребованные на данный момент это *Illumina* и *Ion Torrent*. Саму процедуру *NGS* можно разделить на три крупных этапа. Общая цель каждого из этапов в основном одна и та же, независимо от платформы *NGS* [1].

Первый этап – это обработка «сырых» данных и их первичный анализ. На выходе получаются отдельные небольшие участки исследуемого фрагмента ДНК (риды). Для данного этапа выходными данными являются специализированные файлы (*FASTQ – Illumina, uBAM – Ion Torrent*).

В ходе второго этапа производится процесс объединения ридов. Сборки генома (*assembly*) в случае секвенирования *de novo*, или выравнивание (*alignment*) если имеется референсный геном. Задача сборки генома считается более сложной и включает в себя множество различных аспектов, таких как исправление ошибок секвенирования, заполнение пропусков в покрытии и разрешение геномных повторов. Для этого используют ассемблеры – программное обеспечение для сборки ридов в более длинные фрагменты (контиги). После этого формируются скаффолды – упорядоченные списки контигов. Чаще всего скаффолд представляется как нуклеотидная последовательность с неизвестными нуклеотидами в середине. Далее эти данные проходят процесс, называемый поиском вариантов (*variant calling*) – последовательно определяются конкретные нуклеотиды, на основании многократного прочтения одного и того же участка генома (локуса).

Третий этап служит для анализа полученной последовательности ДНК и извлечения полезной информации. Он состоит из этапа аннотации генома т.е. определения местоположения функционально значимых участков генома и его кодирующих областей. После этого следует их интерпретация – определение того, как и на что эти гены влияют или могут влиять.

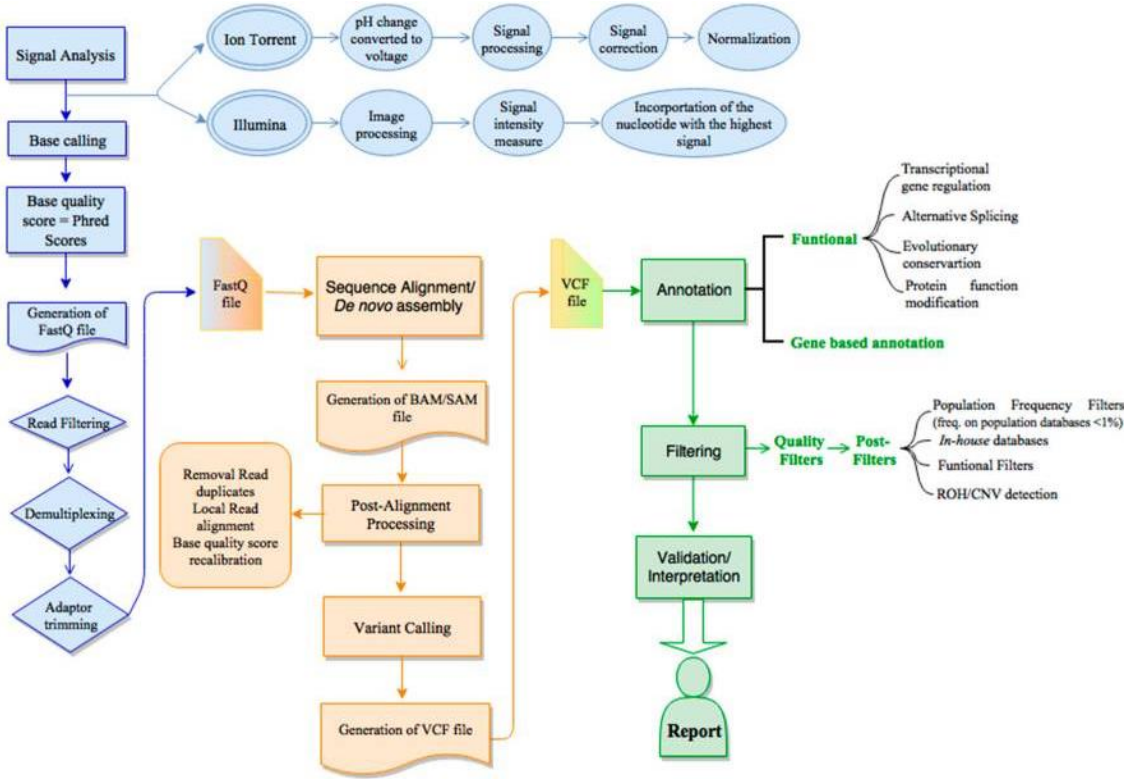

Рис. 2. Общая схема процесса NGS

2. Процедура сборки генома

Процедура сборки генома является сложной вычислительной задачей из-за ряда факторов: огромное количество прочтений, наличие ошибок в них и таких особенностей геномных последовательностей, как четырехбуквенный алфавит и наличие множества повторяющихся фрагментов в разных участках ДНК. Существует несколько классических подходов к решению задачи сборки генома. Все они используют теорию графов, но каждый из подходов определяет вершины и ребра по-разному. Последовательность генома закодирована в этих графах и соответствует некоторому обходу графа. Геномная сборка обычно представляет собой набор неветвящихся путей в графе, поскольку его однозначный и полный обход зачастую невозможен из-за сложной топологии, например наличия множества ветвей и циклов. Существуют три основных класса алгоритмов сборки генома [2]:

- Стратегия Перекрытие-Компоновка-Консенсус. Подход Перекрытие-Компоновка-Консенсус (англ. *Overlap-LayoutConsensus, OLC*) создает граф перекрытий, где вершины представляют собой прочтения и две вершины соединены, если соответствующие им прочтения перекрываются. Для построения такого графа требуется найти перекрытия между всеми парами прочтений, что является очень трудоемкой задачей даже при использовании дополнительных эвристик.

- Стратегия графа де Брюйна. Этот метод сначала разрезает прочтения на более короткие *k*-меры (строки длины *k*), а затем использует все полученные *k*-меры для построения графа де Брюйна (англ. *de Bruijn graph, DBG*). Вершины *DBG* являются *k*-мерами, и две вершины связаны, если они являются префиксом и суффиксом *k* + 1-мера, присутствующего во входных прочтениях. Граф также может быть упрощен в так называемый сжатый *DBG* путем замены неветвящихся путей на отдельные ребра.

- Стратегия строкового графа. Концепция является улучшением *OLC* подхода. Граф создается путем конструирования графа перекрытий и дальнейшего объединения перекрывающихся прочтений и удаления транзитивных ребер и прочтений, которые содержатся в некотором другом прочтении по сравнению с *DBG*, строковый граф намного сложнее построить, но он обычно лучше разрешает неоднозначности, вызванные короткими повторениями в геноме (длиннее, чем *k*, но короче длины прочтения) и более компактен для хранения.

Отдельный класс задач сборки генома возникает при метагеномном секвенировании. Оно предполагает изучение генетического материала, извлеченного непосредственно из образцов окружающей

среды, то есть смеси различных бактерий с различными уровнями представленности. При этом покрытие прочтениями различных геномов в образце является крайне неоднородным, а высококонсервативные геномные области часто присутствуют у разных видов. Также, родственные штаммы (штамм-группа организмов одного вида вирусов, бактерий, других микроорганизмов имеющая идентичную нуклеотидную последовательность) с очень похожими геномами часто также имеют и значительные различия в геномной последовательности. Все это усложняет процесс секвенирования и приводит к специфическим особенностям процедуры сборки метагеномных данных.

3. ML в задачах сборки генома

Существуют подходы, решающие некоторые проблемы сборки генома с помощью машинного обучения. Машинным обучением (*Machine Learning, ML*) называют раздел прикладной математики, в который входят статистические, оптимизационные методы и методы искусственного интеллекта, и исследующий способы построения алгоритмов обучения на основе эмпирических данных. Однако, его использование на любом этапе анализа на данный момент подразумевает тщательный контроль, так как необходимо учитывать большое количество различных аспектов. Поэтому для каждого конкретного случая выбираются наборы данных, их специфика и обучение модели, пригодное для этих данных [3].

Важно отметить, что сборка генома – это комплексный процесс, который состоит групп процедур, которые можно разделить на три этапа:

- предварительная сборка – производится сразу после получения ридов и направлена на их обработку. Во время процедуры выполняется обнаружение ошибок считывания и их исправление. Также может производиться процедура кластеризации ридов в «корзины» (*bins*) – *binning*;
- непосредственная сборка – восстановление последовательности ДНК из ридов;
- валидация сборки – цель этого процесса, состоит в том, чтобы проверить согласованность геномных последовательностей, полученных в результате сборки, выявить – и в некоторых случаях исправить – ошибки в последовательностях.

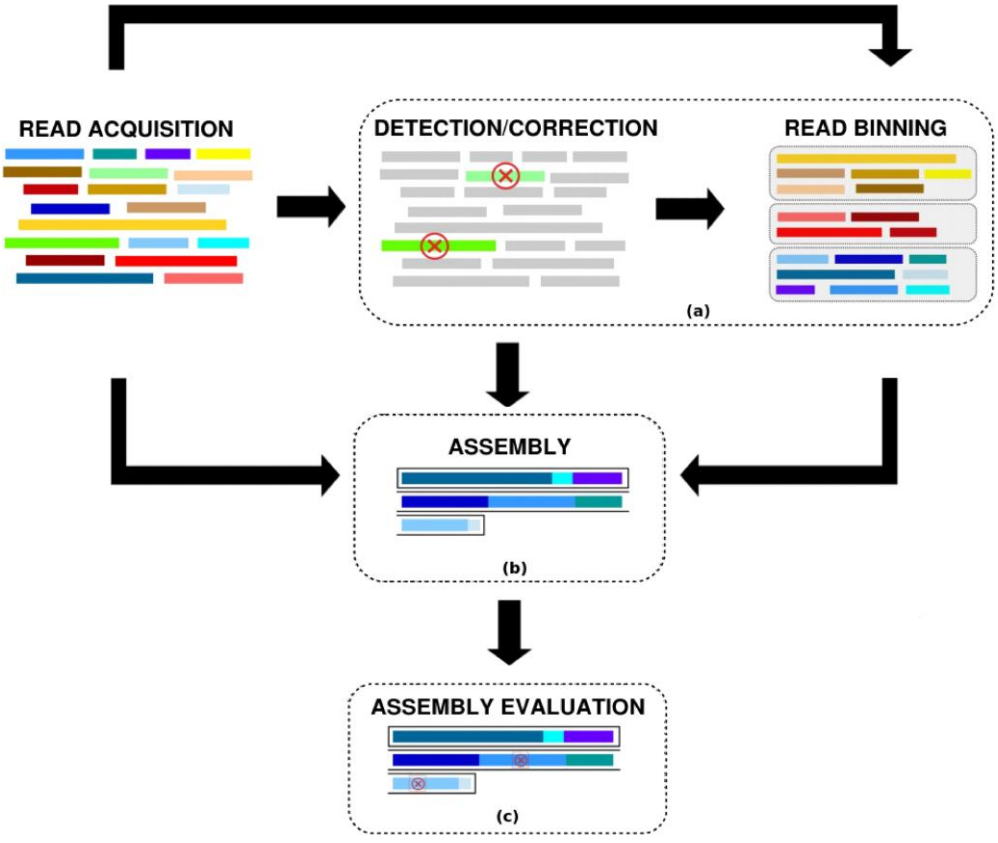

Рис. 3. Этапы сборки генома

Подходы, применяющие *ML* для сборки генома, также можно разделить на три вида: для пресборки (*preassembly*), для постсборки (*postassembly*) и для автосборки (*autoassembly*), в зависимости от того, какую часть процесса они совершенствуют. Термин «автосборка» подразумевает, что ассемблер самостоятельно и недетерминированно определяет действия для сборки ридов исходя из знаний, получаемых в процессе обучения. Функции для обучения модели выбираются в соответствии с целью, и элементы, используемые в подходах пресборки, могут не совпадать с теми, которые используются для автосборки.

Ангелери и др. описали использование машинного обучения для группировки ридов для предварительной сборки: рекуррентная нейронная сеть (*RNN*) моделировала последовательность оснований в заданном фрагменте, затем оценивались реальные и моделируемые наборы ридов. Для каждого рида строилась *RNN*, способная моделировать соответствующую последовательность и обучалась до тех пор, пока процент попаданий не превысит установленный порог. В итоге, все оставшиеся риды группировались вместе с тем, в которой *RNN* показывала минимальный коэффициент неправильной классификации. Результатом этого метода стало улучшение результатов сборки с точки зрения покрытия по сравнению с другими методами кластеризации. Сборка имела меньшее количество ошибок и высокие показатели для решения задач о подобии для выборок с низким охватом. Тем не менее, итоговые показатели сравнения были менее 70%, как с учетом улучшенных результатов, так и без них. Однако, изначальные риды подбирались аналитически, поэтому их было возможно повторно использовать для других наборов данных.

Константинеску и др. предложили построить (*RNN*) основанную на сортировке случайно отобранных ридов по жадной стратегии и стратегии графа де Брюйна. С помощью жадного алгоритма время сборки уменьшилось в 10 раз без потери её качества и увеличилось покрытие в 5 раз, если для обучения использовался кластер ридов. Стратегия графа де Брюйна использовалась для группировки в неконтролируемом машинном обучении [3].

Машинное обучение было использовано Крачуновым и для обнаружения ошибок и геномных вариаций, содержащихся в данных секвенирования. Поиск ошибок основывался на анализе базовой частоты и предположении, что редкие основания непосредственно связанны с ошибками или вариациями. Изначально выбираются области-кандидаты, затем в них с помощью машинного обучения проводился поиск ошибок и вариаций. Все риды группируются по сходству, затем в группах вычисляется частота для каждого нуклеотида с учетом частоты каждого основания среди анализируемых ридов. Если частота ридов была ниже заданного порога, то они заносились в число кандидатов, которые позже анализировались и классифицировались с помощью *ML*. Такой подход позволил анализировать метагеномные данные с точностью выше 99%, поэтому применение *ML*, особенно нейронных сетей, обладает большим потенциалом для обнаружения ошибок и вариаций. Минусом является то, что ошибки генерируются на основе корректных данных, что ограничивает переиспользование для других наборов данных [4].

Чой и др. был описан подход к валидации геномных сборок с использованием *ML*. Цель подхода – создание интеллектуальной компьютерной системы распознавания. Поиск ошибок сборки осуществлялся разделением контигов на блоки по 500 пар оснований, далее следовал их анализ и при необходимости коррекция специалистами. Однако, этот подход не был достаточно эффективен, так как средние показатели точности не превышали 60%. Это приводило к большому объему ложноположительных результатов, и примерно 40% процентов лишних блоков передавались на рассмотрение эксперту. Подход также нельзя использовать для нескольких образцов, так как для обучения необходим законченный образец генома и метрика идентификации ошибок, извлеченные из каждого блока [5].

Бодили и др. предложили подход под названием «Скафолдер скафолдов» (*ScaffoldScaffolder*), использующий машинное обучение для повышения качества процедуры скафолдинга генома диплоидных гетерозигот. В диплоидных гетерозиготных организмах имеются пары хромосом, называемых гомологичными, молекулы ДНК имеют схожие структуры, но обычно содержат небольшие вариации, которые традиционные ассемблеры не могут обработать и создают сиквенсы для каждой пары гомологичных хромосом. Автор подхода использовал машинное обучение с учителем для идентификации контигов из гомологичных гетерозиготных хромосом, названных «гомолотигами» (*homolotigs*). Алгоритм выделял структуры, называемые пузырями, в которых вероятно могли содержаться гомолотиги. Из них выделялись признаки, на основе которых затем производилось обучение двоичного классификатора, который позже предсказывал другие пары контигов. Скафолдер скафолдов может широко

применяться так как на вход необходимы только контиги и риды, но ограничен тем, что требует сиквенсы без ошибочных данных [3].

Куринг и др. представили подход на основе ML под названием *SuRankCo* (акроним *Supervised Ranking of Contigs*), который направлен не на улучшение качества контигов при сборке, а на улучшение результатов объединения контигов путем их ранжирования, что позволяет надежнее проанализировать геномную информацию. Каждый контиг сначала таксономически классифицировался и высчитывался процент идентичности с наиболее вероятным таксоном. Далее рассчитывалась корреляция между процентами и соответствующими оценками *SuRankCo*, учитывая коэффициент корреляции Пирсона и Спирмена 0,77 и 0,72. Для выравнивания контигов методу требуются сиквенсы родственных организмов, что может помешать использовать модель повторно в других образцах, особенно для анализа метагеномных данных [6].

Методы машинного обучения возможно использовать не только во время процесса сборки, но и в качестве стратегии или части стратегии сборки, т.е. для автосборки. Например, Палмер предложил использовать ML для нахождения перекрытий в ридах до создания контигов во время сборки. Выровненные пары ридов оценивались по наличию соответствий и отделялись от тех, в которых соответствия нет, что обычно бывает в парах ридов из сегментов с геномными повторами. Был создан ассемблер на основе адаптированного *OLC*, названный *Minimus*, который оценивает особенности выравнивания ридов и находит перекрытия в них. Точность результатов этого подхода достигает 97,82%, в то время как без использования *ML* – 74,9%. Также, были получены контиги вдвое больше средней длинны, которые не увеличивали количество ошибок в сборке генома, что дало возможность применять этот метод для улучшения качества сборки, в том числе в подходах, основанных на графах де Брюйна [7].

Чжу и др. представили ассемблер *PERGA* на основе обучения с учителем, который использует жадную стратегию и *SVM* (*support vector machine*) модель для сборки контигов и скаффолдов. Методом опорных векторов прогнозируется необходимость расширения контигов, если ветви образовались из повторов и ошибок секвенирования, и указывается наиболее вероятные пути продолжения. В прогнозах показатель успеха достигает 99,7%, что позволяет рассматривать более длинные контиги и скаффолды, которые при этом проверены лучше, чем в других ассемблерах.

Ванг и др. предложили ассемблер Ксандер для специфического протеинкодирования генов, который сочетает в себе графы де Брюйна и скрытую марковскую модель (СММ). По сравнению с другими ассемблерами этот подход позволяет получить больше контигов с более широким охватом генов до 94,6%. Помимо этого, Ксандер сокращает пространство поиска и позволяет найти редко встречающиеся гены в метагеномных образцах без дополнительных операций. Это делает ассемблер эффективным инструментом для анализа микробиологического состава и разнообразия в метагеномных образцах. При наличии материалов для обучения СММ Ксандер может использоваться при любом наборе данных [3].

Жи и др. разработали всеобъемлющую процедуру сборки *MetaSort* с использованием машинного обучения и существующих ассемблеров. Процедура состоит из двух разных, но сообщающихся сборок мета-*O* и мета-*S*, которые собираются в одну окончательную сборку. Мета-*O* является традиционной сборкой, в ходе которой анализируется полный метагеномный образец. При использовании частей образцов, называемых мини-метагеномами, запускается мета-*S*, которая получает результаты, обычно недоступным при анализе полного образца. *MetaSort* использует метод опорных векторов для обогащения набора контигов каждого генома, найденного мета-*S*, с помощью прогнозирования участков сборки мета-*O*, которые могут принадлежать соответствующему геному. Этот метод показал высокую точность и чувствительность, что дает возможность собирать высококачественные геномы и восстанавливать геномы организмов с низким содержанием в метагеномных образцах, а также находить геномные вариации, которые подавляются в некоторых сборщиках. Ограничения метода заключаются в том, что требуются мини-метагеномы для получения сборок, однако ограничение можно обойти использованием родственных геномов как мини-метагеномов [8].

После сборки при секвенировании производится процедура поиска вариаций, которая также может быть модернизирована, с помощью машинного обучения. Например, проект *DeepVariant* от *Google* – это технология глубокого обучения для восстановления истинной последовательности генома из данных секвенатора с точностью большей, чем у значительной части классических методов. Интересной особенностью данной технологии является то, что она превращает задачу поиска

вариаций, в задачу классификации изображений, что позволяет использовать существующие технологий *Google* [9]. Несмотря на находчивость такого подхода, данное решение не всегда корректно работает, особенно в случае инделов. Индел (*INDEL, Insertion/Deletion*) – тип генетических вариаций, при котором происходит удаление или вставка одного, или нескольких нуклеотидов. Поэтому производятся попытки разработки нейронных сетей меньшего размера, создаваемых непосредственно для проведения процедуры выбора вариаций и повышения его качества [10].

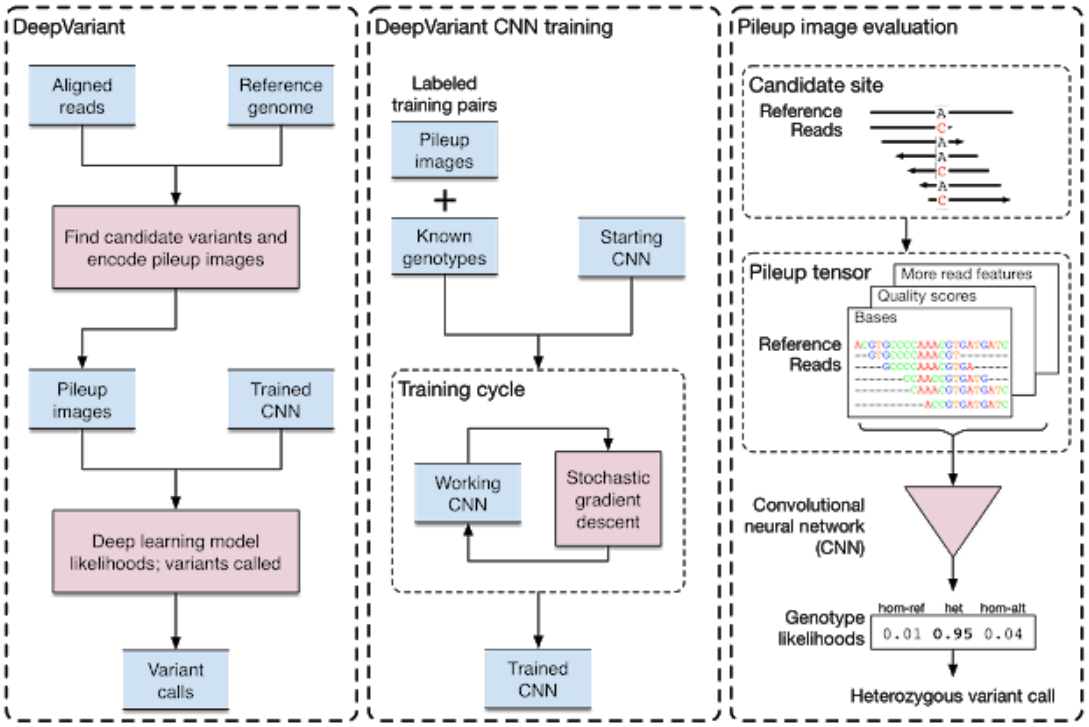

Рис. 4. Общая схема работы DeepVariant

4. ML в задачах аннотирования генома и классификации вариаций

Аннотирование генома и классификация вариаций стандартные процедуры при проведении генетического исследования. Для классификации вариаций разработано достаточно много методов [3]. Часть из этих методов была развита с помощью технологий машинного обучения. К примеру, *Combined Annotation Dependent Depletion* (*CADD*) – это инструмент, который оценивает вредоносность (патогенность) отдельных вариаций различного вида в геноме человека. Расширение *CADD*, основанное на глубоком обучении, названное *DANN*, продемонстрировало большую эффективность с использованием того же набора признаков [11]. Это позволяет говорить, что подход, предполагающий расширение уже существующих инструментов с помощью методов глубокого обучения, может быть крайне действенным. К сожалению, точность классификации данных инструментов зачастую недостаточна для их применения самих по себе, особенно для клинических исследований. Однако они могут служить для выбора вариаций, которые будут приоритетными для дальнейшего исследования стандартными средствами.

Другая проблема, требующая решения – это обнаружение и прогнозирование патогенных вариаций, часть из которых происходит из-за неправильного процесса сплайсинга (около 10%). Сплайсинг – процесс вырезания определённых нуклеотидных последовательностей из молекул РНК и соединения последовательностей, сохраняющихся в молекуле, в ходе формирования «зрелой» РНК. *SpliceAI*, 32-х слойная глубокая нейронная сеть, способна предсказывать различные виды сплайсинга непосредственно на основе данных последовательности нуклеотидов [12]. Примечательно, что при стандартном размере окна в 80 нуклеотидов точность прогнозирования *SpliceAI* составляла 57%, что типично для многих существующих инструментов. Однако при увеличении окна до размера в 10 КБ, точность была увеличена до 95%. Так алгоритм смог обнаружить вариации скрытого сплайсинга, которые могли вызывать нарушения нервного развития.

Заключение

Несмотря на все существующие на настоящий момент достижения, предстоит еще долгий путь в развитии методов и технологий секвенирования. Необходимы дальнейшие улучшения платформ и стратегий обработки данных, для снижения частоты появления ошибок и повышения качество обнаружения и классификации вариаций.

Современные вычислительные методы могут быть не в состоянии обрабатывать и извлекать весь потенциал из столь больших геномных данных. Однако использование машинного обучения при наличии качественной модели и данных позволяет достичь лучших результатов в разнообразных задачах, например сборке генома, по сравнению со стандартными подходами, особенно в приложении к метагеномным исследованиям.

Список литературы

1. Pereira R., Oliveira J., Sousa M. Bioinformatics and Computational Tools for Next-Generation Sequencing Analysis in Clinical Genetics // Journal of Clinical Medicine. 2020. Vol. 9. N. 1. Article number: 132. URL : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7019349/.
2. Гуревич А. А. Вычислительные методы для анализа подверженных ошибкам метабологеномных данных : дис. канд. физ.-мат. наук : 03.01.09. Санкт-Петербург, 2018. 224 с. URL : https://disser.spbu.ru/files/2018/disser_gyrevich_aa.pdf.
3. Padovani de Souza K., Setubal J. C., Ponce de Leon F de Carvalho A. C. , Oliveira G., Chateau A., Alves R.Machine learning meets genome assembly // Briefings in Bioinformatics. 2019. Vol. 20. N. 6. P. 2116–2129. UR L: https://pubmed.ncbi.nlm.nih.gov/30137230/.
4. Krachunov M., Nisheva M., Vassilev D. Machine learning models in error and variant detection in high-variation high-throughput sequencing datasets. // Procedia Computer Science. 2017. Vol. 108. P. 1145–1154.
1. Choi J.-H., Kim S., Tang H., et al. A machine-learning approach to combined evidence validation of genome assemblies. // Bioinformatics. 2008. Vol. 4. N. 6. P. 744–750.
2. Kuhring M., Dabrowski P. W., Piro V. C., et al. SuRankCo: supervised ranking of contigs in de novo assemblies. // BMC Bioinformatics. 2015. Vol. 16. N. 1. P. 240. URL : https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-015-0644-7.
3. Palmer L. E., Dejori M., Bolanos R., et al. Improving de novo sequence assembly using machine learning and comparative genomics for overlap correction. // BMC Bioinformatics. 2010. Vol. 11. N. 1. P. 33. URL : https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-11-33.
4. Ji P., Zhang Y., Wang J., et al. MetaSort untangles metagenome assembly by reducing microbial community complexity. // Nature Communications. 2017. Vol. 8. P. 14306. URL : https://www.nature.com/articles/ncomms14306.
5. DePristo M. Poplin R.; Google Brain Team. DeepVariant : Highly Accurate Genomes With Deep Neural Networks // Google AI Blog. 2017 December 4. URL : https://ai.googleblog.com/2017/12/deepvariant-highly-accurate-genomes.html.
6. Chin J. Simple Convolutional Neural Network for Genomic Variant Calling with TensorFlow // Medium: an online publishing platform / Medium Corporation. 2017 Jul 17. URL : https://towardsdatascience.com/simple-convolution-neural-network-for-genomic-variant-calling-with-tensorflow-c085dbc2026f.
7. Dias R., Torkamani A. Artificial intelligence in clinical and genomic diagnostics // Genome Medicine. 2019. Vol. 11. |Article number: 70. 12 p. URL : https://genomemedicine.biomedcentral.com/articles/10.1186/s13073-019-0689-8.
8. Jaganathan K., Kyriazopoulou Panagiotopoulou S., McRae J. F., et al. Predicting Splicing from Primary Sequence with Deep Learning // Cell. 2019. Vol. 176. N. 3. P. 535–548. URL : https://pubmed.ncbi.nlm.nih.gov/30661751/.

9. Quang D., Chen Y., Xie X. DANN : a deep learning approach for annotating the pathogenicity of genetic variants // Bioinformatics. 2015. Vol. 31. N. 5. P. 761–763. URL : https://pubmed.ncbi.nlm.nih.gov/25338716/.