1. **Обзор аналогов и проблемной области**
   1. **Аналитический обзор литературы**

В качестве литературных источников при ознакомлении с предметной областью можно использовать следующие материалы:

1. **Stuart Russel, Peter Norvig. Artifical Intelligence. A Modern Approach. 2010 – 1154c.**

Книга известных исследователей в области искусственного интеллекта Стюарта Рассела и Питера Норвига раскрывает последние наиболее важные применения искусственного интеллекта: распознавание речи, машинный перевод, роботизированные транспортные средства и домашних роботов. Также рассмотрено применение систем искусственного интеллекта в области поиска и обработки информации, техники использования больших объемов данных.

Что характерно, Питер Норвиг является директором по исследованиям в корпорации Google. Так что можно надеяться на описание наиболее интригующих исследований непосредственно от активного участника событий. Также он вместе со своим харизматичным коллегой по работе Себастианом Траном является организатором первых изветных онлайн-курсов по искусственному интеллекту и сооснователем образовательной организации Udacity.

Книга снабжена большим количеством псевдокода и доступных для понимания математических выкладок. Поэтому неудивительно то, что «Искусственный интеллект. Современный подход» используется во многих профильных учебных заведениях в качестве готового учебного пособия.

1. **А. А. Ежов, С. А. Шумский. Нейрокомпьютинг и его применения в экономике и бизнесе. 1998 – 216с.**

В то время, как книга Саймона Хайкина является полным математическим экскурсом в область нейронных сетей, в книге Ежова и Шумского рассмотрены, скорее, вопросы практического применения нейронных сетей в более популярной форме. Прочитать данную книгу можно рекомендовать, прежде всего, начинающему исследователю, чтобы получить первичное представление о возможностях, проблемах и методах выбранной области практической науки.

Интересно и то, что материал написан именно для лучшего понимания основ нейросетевой обработки данных преимущественно в области финансов. То есть все выкладки объясняются максимально простым языком, понятным неподготовленным читателям. Для повышения интереса к области книга снабжена наиболее интересными примерами применения нейронных сетей из реальной жизни, которые все еще не потеряли своей актуальности, несмотря на достаточно долгий срок, прошедший с момента издания.

1. **Марк Лутц. Изучаем Python. 2010 – 1280с.**

В данной книги изложены основополагающие принципы программирования на языке Python. Данная книга является весьма основательным учебником, написанным, что для студентов немаловажно, сравнительно доступным языком и основанным на материалах курсов, которые ведет автор Марк Лутц вот уже более десяти лет. Это позволяет быстро овладеть основными принципами программирования на Python за сравнительно небольшой промежуток времени независимо от уровня начальной подготовки читающего. Хотя базовые знания программирования точно помогут процессу усвоения читателем философии языка.

Особенность четвертого издания, выпущенного в 2010 году, состоит в особом акцентировании отличий версии языка 3.0 от версии 2.6. Но при этом и та, и другая версия языка рассматриваются с одинаковым вниманием, что немаловажно для нас, так как в работе над приложением использовалась, преимущественно, версия 2.7, причиной чего явились весьма специфические требования библиотек.

В целом, книга является достаточным для большинства задач настольным руководством для языка Python.

1. **Саймон Хайкин. Нейронные сети. Полный курс. 2006 – 1103с.**

«Нейронные сети. Полный курс» Саймона Хайкина – известная всем исследователям в области искусственного интеллекта книга об основных парадигмах нейронных сетей. Интересно в ней в первую очередь то, что весь материал снабжен не только строгим математическим обоснованием, но и иллюстрируется наглядными примерами и содержит множество решений встречающихся в реальной жизни задач. В частности в книге на должном уровне рассмотрены задачи распознавания образов, управления и обработки сигналов, столь типичные для этой области.

В нейронных сетях, да и в машинном обучении в целом, невозможно никуда деться от необходимости понимания математических предпосылок используемых методов. Для нашей работы материала, изложенного в книге, хватает с огромным запасом. Рассмотрены там даже методы, которые не будут использоваться по причине их излишней сложности в реализации и понимании их математического аппарата студентом типичной инженерной специальности с курсом высшей математики в три семестра.

Так же, как и книга Марка Лутца является настольным руководством для программистов на языке Python, труд Саймона Хайкина является библией исследователя программных нейронных сетей. Книга обязательна к прочтению.

* 1. **Обзор схожих программных средств**

Зачастую при изучении и оценке эффективности нейронных сетей и других алгоритмов машинного обучения используются программные пакеты, обладающие огромным избыточным функционалом, что можно рассматривать и как преимущество, в случае достаточно хорошей теоретической подготовки и необходимости в экспериментах, и как недостаток, в случае обладания лишь базовым представлением о работе нейронных сетей и используемых алгоритмов. Зачастую, чтобы приступить к углубленному изучению темы, необходимо сначала получить начальное представление о ней, обойдясь сравнительно малыми затратами. Тем не менее, существует ряд аналогов для нашей разработки, которые используются уже достаточно широко и предоставляют широкий спектр возможностей для работы с алгоритмами машинного обучения.

Одним из наиболее мощных пакетов для разработки и редактирования нейронных сетей обладает продукт компании американской The MathWorks – пакет прикладных программ MATLAB. Графический интерфейс для построения нейронных сетей Neural Network Toolbox впервые появился в MATLAB версии 6.0. Преимущество использования именно продукта MATLAB заключается в возможности комбинировать нейронные сети с многочисленными алгоритмами и функциями для инженерных расчетов, а также наличие продвинутых возможностей по оценке полученных алгоритмов. Также преимуществом MATLAB является наличие высокоуровневого интерпретируемого языка программирования, включающего основанные на матрицах структуры данных и широкий спектр функций, а также объектно-ориентированные возможности и интерфейсы к программам, написанным на других языках. И, безусловно, немаловажным положительным фактором является наличие очень большого сообщества пользователей, которое позволяет очень быстро получить ответы на возникающие вопросы и квалифицированную помощь.

Недостатками пакета программ MATLAB является, все же, относительно небольшое число реализованных специфичных структур нейронных сетей, хотя надо заметить, что все основные архитектуры в нем присутствуют. Также реализовано крайне небольшое число функций активации нейронов – целых три штуки. Рассматривать ли это как недостаток данного программного пакета – зависит от поставленной задачи, однако в графическом интерфейсе отсутствует возможность задания произвольной функции активации нейрона, что могло бы быть полезным для исследований конкретных нейросетевых алгоритмов.

Единственным в мире крупным нейросетевым программным продуктом, полностью переведенным на русский язык (по заявлению компании-разработчика), является программный пакет Statistica, разработанный, как это ни странно, компанией Statsoft, тоже американской, а точнее, его модуль Statistica Automated Neural Networks. Как и MATLAB, Statistica представляет большой пакет программ для инженерных расчетов, а значит, нейронные сети можно успешно комбинировать с другими видами алгоритмов в сложных законченных системах. В системе реализовано большое количество нейросетевых архитектур, достаточное для решения многих практических задач и, что немаловажно, мастер решений, который автоматически анализирует задачу и выбирает несколько подходящих для реализации архитектур. Также в пакете программ представлено большое число различных алгоритмов обучения сети, что позволяет иметь большую свободу в выборе оптимальных алгоритмов для решения определенного типа задач. Рекламируются также возможность создания гибридных нейронных сетей, состоящих из нескольких различных архитектур и база готовых решений [4].

Несмотря на все богатство функциональных возможностей и хорошую русификацию, критике можно подвергнуть и продукт Statistica Automated Neural Networks. Так, благодаря впечатляющим функциональным возможностям, мы имеем достаточно сложный интерфейс самой программы. В итоге задача изучения нейронных сетей и экспериментирования с их различными структурами и типами плавно превращается в задачу изучения интерфейса одной отдельно взятой программы. В целом, программа рассчитана на практическое применение в реальных областях специально обученными людьми, что не соответствует задаче академического обучения. Но продукт компании Statsoft можно смело рекомендовать пользователям, достаточно глубоко изучившим нейронные сети и желающим применить полученные знания в какой-либо прикладной задаче [5].

Существует достаточно большое количество мелких продуктов для конструирования нейронных сетей. Многие программисты реализуют их в процессе обучения, так как это позволяет лучше отточить необходимые в работе навыки.

Одним из примеров может служить немецкая программа MemBrain. Отличается от вышеперечисленных крупных продуктов она тем, что предлагает бесплатную лицензию для некоммерческого и академического использования. Однако данный продукт, все же, не совсем подходит для начальных этапов работы с нейросетями по причине отсутствия поддержки какого-либо языка кроме английского, что, впрочем, не должно стать для студентов проблемой, но дополнительным раздражающим элементом стать вполне может [6].

В целом, во всех приведенных программных средствах присутствует возможность оценки качества используемых алгоритмов, но ни один из этих программных пакетов не является достаточно простым, чтобы обеспечить первичное ознакомление с алгоритмами машинного обучения и быстрый выбор наиболее эффективного способа обработки конкретного набора данных. Эту задачу мы и попытаемся решить в своей работе.

* 1. **Постановка задачи**

Основной задачей при проектировании приложения для оценки качества алгоритмов машинного обучения является создание достаточно простого и удобного инструмента для быстрой проверки первичных гипотез относительно выбора оптимального алгоритма для конкретного набора данных.

По причине исключительно большой трудоемкости построения алгоритмов машинного обучения с нуля на языках системного программирования, подобных С++, в данной работе будет широко использоваться высокоуровневый язык программирования Python, который, к тому же, многие программисты предпочитают использовать в задачах машинного обучения наряду с языками Matlab и R.

Особо стоит отметить использование в приложении библиотек PyBrain и Scikit-learn, реализующих многие алгоритмы машинного обучения и содержащие большое количество доступных для использования метрик оценки данных алгоритмов.

PyBrain – одна из лучших библиотек в мире Python для реализации и изучения разнообразных алгоритмов, связанных с нейронными сетями. Scikit-learn реализует часть алгоритмов машинного обучения, которая лежит за пределами нейросетевых алгоритмов: бинарные деревья, метод опорных векторов, алгоритмы кластеризации. Кроме того, данная библиотека снабжена набором метрик оценки качества алгоритмов машинного обучения буквально на все случаи жизни.

Для построения графического интерфейса используется библиотека Glade, генерирующая описание графического интерфейса на основе XML. Потенциально есть возможность получить кроссплатформенную реализацию графического интерфейса, так как существует порт Glade для семейства ОС Windows. В пользу Glade также говорит наличие удобного инструмента для проектирования интерфейса – Glade Interface Designer.

Основными целевыми операционными системами будут являться Debian-based дистрибутивы Linux, что соответствует сложившейся ситуации в области систем машинного обучения и позволяет следовать в полной степени лучшим традициям проектирования свободного программного обеспечения. Принципиально возможно портирование приложения для использования в сочетании с семейством операционных систем Windows по причине кроссплатформенности абсолютного большинства используемых библиотек.

* 1. **Краткий обзор используемых алгоритмов**

Алгоритм обратного распространения ошибки – метод обучения многослойного перцептрона, основная идея которого состоит в распространении сигналов ошибки от выходов сети к ее входам, в противоположность прямому распространению сигналов в обычном режиме работы [4]. Алгоритм предполагает два прохода по всем слоям сети: прямого и обратного. При прямом проходе входной вектор подается на сенсорные узлы сети, после чего распространяется от слоя к слою. В результате генерируется набор выходных сигналов, который является фактической реакцией сети на данный входной образ. Во время обратного прохода все синаптические веса нейронов настраиваются в соответствии с правилом коррекции ошибок, в результате чего формируется сигнал ошибки. Этот сигнал распространяется по сети в направлении, обратном направлению синаптических связей [5].

Метод эластичного обратного распространения ошибки во многих случаях является более оптимальным выбором, чем обычный алгоритм обратного распространения ошибки. Плюс эластичного распространения в отсутствии необходимости подбирать параметры обучения, которые в случае с обратным распространением ошибки часто приводили к параличу сети.

Деревья принятия решений, также называемые деревьями классификации, используется для построения прогнозных моделей и классификации данных на основе разбиения исходных данных на подмножества, основанные на значениях атрибутов.

Основной идеей метода опорных векторов является перевод исходных данных в пространство большей размерности и поиска оптимальной разделяющей гиперплоскости в этом пространстве.

Метод К-средних – наиболее популярный из всех, используемых для кластеризации данных. Принцип его работы заключается в подборе положения центров кластеров таким образом, чтобы суммарное квадратичное отклонение точек от центров оказалось минимальным.

Метод Mean-shift основан на поиске максимумов функции плотности для наборов данных. Алгоритм автоматически устанавливает число кластеров и обычно используется в задачах компьютерного зрения и обработки изображений.

Плотностный алгоритм кластеризации, он же DBScan, решает проблемы разбиения данных на кластеры произвольной формы, причем было доказано, что в отличие от других алгоритмов кластеризации, которые обычно создают кластеры по форме близкие к сферическим, DBScan способен правильно распознать более большой спектр пространственных форм размещения данных.