# ОБЗОР И СРАВНЕНИЕ JAVA-БИБЛИОТЕК ДЛЯ МАШИННОГО ОБУЧЕНИЯ

**Г.Г. Коновалов,** *студент*
**Научный руководитель:** *Н.О.К. Зарбалиева, доцент*
**Волгоградский государственный университет**
**(Россия, г. Волгоград)**

*DOI:10.24412/2500-1000-2025-10-2-45-49*

***Аннотация.*** *В статье представлен обзор и сравнительный анализ популярных Java-библиотек для машинного обучения. Рассмотрены особенности архитектуры, функциональные возможности, производительность, удобство использования и области применения следующих решений: Weka, Deeplearning4j, MOA, Smile, Encog и Tribuo. Отдельное внимание уделяется критериям выбора библиотеки в зависимости от типа задачи – от образовательных проектов до промышленных систем. На основе анализа сделан вывод о роли Java в современных задачах машинного обучения и перспективах его применения в этой области.*

***Ключевые слова:*** *машинное обучение, Java, библиотеки, Weka, Deeplearning4j, MOA, Smile, Encog, Tribuo, нейронные сети, обработка данных, искусственный интеллект.*

В настоящее время Java остаётся одним из наиболее широко используемых языков программирования в промышленной разработке благодаря своей платформенной независимости, зрелой экосистеме и хорошей поддержке многопоточности. Несмотря на то, что в области машинного обучения доминируют такие языки, как Python и R, интерес к Java в этой сфере сохраняется в тех случаях, когда требуется интеграция моделей в существующие корпоративные решения или когда критична производительность в среде JVM.

Машинное обучение всё чаще применяется в разнообразных отраслях – от финансов и медицины до телекоммуникаций и логистики. В таких условиях растёт потребность в инструментах, которые позволяют строить и внедрять интеллектуальные системы на уже существующих технологических стэках. Java-библиотеки для машинного обучения позволяют разрабатывать, обучать и внедрять модели без необходимости переключаться на другой язык программирования, что особенно важно для крупных компаний, чья инфраструктура уже построена на JVM.

**Выбор критериев сравнения**

Для того чтобы объективно сравнить Java-библиотеки для машинного обучения, необходимо опираться на ряд ключевых характеристик, определяющих их пригодность для различных задач. Одним из наиболее значимых аспектов является обхват реализованных алгоритмов. Важную роль играет и простота использования, качество документации, примеры кода и доступность обучающих материалов, поскольку даже самая функциональная библиотека может остаться невостребованной без удобного интерфейса и понятной структуры.

Не менее важным критерием становится производительность. В условиях, когда обработка больших объёмов данных или обучение сложных моделей требует значительных вычислительных ресурсов, наличие поддержки многопоточности и, при необходимости, GPU-ускорения, может существенно повлиять на выбор инструмента. Ещё одним фактором выступает совместимость с другими технологиями, такими как Apache Spark или Hadoop, что особенно актуально в проектах, связанных с обработкой распределённых данных.

Также стоит уделять внимание активности сообщества, частоте обновлений и наличию поддержки со стороны разработчиков, так как это определяет, насколько библиотека будет полезна и безопасна в долгосрочной перспективе. И наконец, возможность визуализации данных и результатов моделирования, а также совместимость с Python и другими языками, может стать решающим моментом в выборе библиотеки для тех, кто работает в многопрофильной команде или предпочитает гибкие инструменты анализа.

**Обзор популярных Java-библиотек для ML**

Современная экосистема Java предоставляет несколько мощных библиотек, ориентированных на задачи машинного обучения. Каждая из них имеет свою специфику, архитектурные особенности и области применения.

Одним из наиболее известных решений в этой области является Weka – библиотека, созданная в академической среде и получившая широкое распространение благодаря своему графическому интерфейсу и богатому набору алгоритмов. Её основное преимущество заключается в доступности: порог входа здесь значительно ниже по сравнению с более технически сложными библиотеками, что делает её удобной для обучения и быстрой прототипизации [1]. Однако, проблема с масштабируемостью и отсутствие полноценной поддержки современных фреймворков снижают её эффективность в промышленных задачах.

В противоположность Weka, библиотека Deeplearning4j ориентирована на 1глубокое обучение и использование в высоконагруженных системах. Она построена на стеке, включающем ND4J для работы с числовыми массивами и интеграцию с Apache Spark, что делает её мощным инструментом для распределённой обработки данных. Особое внимание здесь уделено производительности, в том числе за счёт возможности использования GPU [4]. Тем не менее, освоение этой библиотеки требует более глубоких технических знаний и подготовки.

Ещё одним интересным инструментом является MOA – библиотека, созданная для обработки потоковых данных в реальном времени. Её сильной стороной является специализация на адаптивных алгоритмах, способных обучаться на непрерывных потоках информации. Это делает её особенно актуальной для задач, связанных с мониторингом, интернетом вещей или детектированием аномалий. Важным фактором является и её тесная интеграция с Weka, что позволяет расширять возможности обеих платформ [3].

Наиболее универсальным решением можно назвать Smile – современную библиотеку, сочетающую высокую производительность с широким спектром алгоритмов. Она привлекает внимание за счёт чистой архитектуры и наличия средств визуализации, что упрощает анализ данных на всех этапах. Несмотря на отсутствие инструментов для глубокого обучения, Smile остаётся востребованным выбором в задачах, где ключевую роль играет интерпретируемость и скорость работы.

Существуют также более компактные и специфичные решения, такие как Encog, ориентированный на разработку нейросетевых моделей [6]. Он привлекает разработчиков своей лаконичностью и возможностью быстрой интеграции, но в то же время ограничен в плане масштабируемости и поддержки современных подходов. Среди новейших разработок стоит отметить библиотеку Tribuo, созданную Oracle и спроектированную с учётом современных требований. Её архитектура делает акцент на модульности, надёжности и возможности сериализации моделей, а поддержка формата ONNX открывает путь к межъязыковой совместимости [5].

Таблица. Сравнительная таблица Java-библиотек для машинного обучения

| Библиотека | Deep learning | GUI | GPU | Поддержка потоков | Интеграция | Документация |
|---|---|---|---|---|---|---|
| Weka | Нет | Да | Нет | Частично | MOA | Отличная |
| DL4J | Да | Нет | Да | Да | Spark | Хорошая |
| MOA | Нет | Частично | Нет | Да | Weka | Средняя |
| Smile | Нет | Нет | Нет | Нет | Нет | Хорошая |
| Encog | Частично | Нет | Нет | Нет | Нет | Умеренная |
| Tribuo | Ограничено | Нет | Нет | Да | ONNX | Отличная |

Сравнение Java-библиотек для машинного обучения позволяет более чётко представить их сильные и слабые стороны, особенно если рассматривать их через призму практического применения. Различия между ними становятся особенно заметными, когда речь заходит о поддержке конкретных алгоритмов, наличии инструментов для глубокого обучения, графического интерфейса, а также возможности ускорения вычислений с помощью GPU. Некоторые библиотеки сосредоточены на удобстве и наглядности, облегчая обучение и эксперименты, в то время как другие ориентированы на производительность и промышленную интеграцию.

Так, например, Weka, несмотря на свои недостатки, выделяется наличием графического интерфейса и большим числом классических алгоритмов, что делает её удобной для быстрого анализа данных и обучения студентов. В то же время, библиотеки, такие как Deeplearning4j, фокусируются на более глубокой интеграции с современными технологиями и предоставляют средства для обучения нейронных сетей, включая использование графических процессоров и работу в распределённых вычислительных средах. Это делает их подходящими для сложных задач, где масштабируемость и скорость имеют решающее значение.

Библиотеки, ориентированные на потоковую обработку (MOA), позволяют работать с непрерывными данными, что важно для систем реального времени, например, в мониторинге или анализе поведения пользователей. Такие решения значительно отличаются от более традиционных библиотек, так как требуют другой подход к обучению моделей и обработке информации. В свою очередь, Smile выделяется как сбалансированная платформа, способная обеспечивать высокую производительность при работе с классическими задачами машинного обучения, но без акцента на глубокое обучение.

Encog и Tribuo занимают особое место в этом ландшафте. Первый привлекает своей простотой и нацеленностью на нейросетевые структуры, тогда как второй предлагает современную архитектуру с акцентом на модульность, интероперабельность и строгую организацию данных. Поддержка формата ONNX в Tribuo делает возможной интеграцию моделей с решениями, разработанными на других языках, что особенно важно в мультиплатформенных проектах.

**Примеры использования**

Выбор подходящей Java-библиотеки для машинного обучения во многом зависит от характера задачи, с которой сталкивается разработчик или команда. В образовательной среде, где важна наглядность и лёгкость освоения, предпочтение часто отдаётся инструментам, предлагающим интуитивно понятный интерфейс и богатую визуализацию. Такие библиотеки позволяют быстро построить модель, провести эксперименты и увидеть результаты без необходимости глубоко погружаться в технические детали. Это особенно актуально для студентов, преподавателей и начинающих специалистов, которым важна не столько производительность, сколько доступность и простота.

В корпоративной среде, где машинное обучение становится частью больших систем обработки данных, основной акцент смещается в сторону масштабируемости и совместимости с промышленными инструментами. Здесь на первый план выходят библиотеки, способные эффективно работать в распределённых средах, обеспечивать интеграцию с системами хранения и передачи данных, а также использовать вычислительные ресурсы максимально эффективно. Учитывая требования к производительности, в таких условиях ценится поддержка многопоточности и возможность ускорения с использованием графических процессоров.

Задачи, связанные с обработкой потоков данных в реальном времени, требуют особого подхода к выбору инструментов. Не каждая библиотека способна справляться с изменяющейся во времени структурой данных и необходимостью оперативной адаптации моделей. Поэтому при разработке систем, реагирующих на данные на лету – например, в области интернет вещей, финансового мониторинга или кибербезопасности – особенно важно использовать специализированные решения, разработанные с учётом этих особенностей.

Когда речь идёт о промышленном применении, где устойчивость, надёжность и поддержка долгосрочной эксплуатации выходят на первый план, важны не только технические характеристики библиотеки, но и такие факторы, как активность сообщества, регулярность обновлений, качество документации и наличие корпоративной поддержки. В этом контексте предпочтение часто отдают зрелым, хорошо документированным платформам, на которые можно опереться при реализации критически важных компонентов [2].

При разработке решений на основе глубокого обучения особое внимание уделяется возможностям построения и обучения сложных нейросетевых архитектур. В таких проектах важна не только гибкость настройки, но и производительность обучения, особенно при работе с большими объёмами данных. Здесь ключевыми становятся поддержка современ-

ных алгоритмов, возможность работы с GPU, а также совместимость с форматами, обеспечивающими переносимость моделей между различными платформами.

**Заключение**

Сравнительный анализ Java-библиотек для машинного обучения показывает, что каждая из них ориентирована на определённый тип задач и целевую аудиторию. Универсального решения, которое одинаково хорошо справлялось бы с любыми сценариями, не существует, и выбор инструмента всегда должен основываться на контексте применения. Некоторые библиотеки демонстрируют силу в простоте и удобстве, другие – в производительности и глубокой интеграции с инфраструктурными решениями. Есть решения, которые отлично подходят для образовательных целей и быстрой прототипизации, тогда как другие рассчитаны на использование в высоконагруженных и распределённых системах.

Java, несмотря на доминирование Python в области машинного обучения, по-прежнему сохраняет своё значение, особенно там, где важна стабильность, безопасность и интеграция с уже существующими системами. Промышленные проекты, особенно в крупных компаниях с устоявшейся архитектурой на базе JVM, часто выигрывают от использования Java-библиотек, так как это позволяет избежать лишней сложности, связанной с внедрением решений на других языках. Кроме того, развитие таких проектов, как Tribuo, говорит о том, что интерес к машинному обучению в среде Java не только сохраняется, но и получает новое дыхание.

Принимая во внимание всё, что было рассмотрено ранее, можно сделать вывод, что Java остаётся жизнеспособной платформой для разработки систем машинного обучения, несмотря на доминирующее положение других языков в этой области. Каждая из исследованных библиотек предлагает уникальный подход к решению задач, связанных с анализом данных, обучением моделей и их внедрением в реальные приложения. Одни решения обеспечивают высокую доступность и лёгкость освоения, что делает их полезными для обучения и научных экспериментов, другие ориентированы на производственные среды, где важна масштабируемость, скорость и устойчивость.

Сильные стороны Java-библиотек проявляются там, где необходима тесная интеграция с корпоративными системами, а также в тех случаях, когда предпочтительна работа в едином технологическом стеке. Многие из этих инструментов обладают зрелой архитектурой, поддержкой современных форматов и возможностями оптимизации производительности. Однако нельзя не отметить и существующие ограничения: некоторая отсталость в области глубокого обучения, менее развитое сообщество по сравнению с Python, а также ограниченное число актуальных учебных материалов и примеров.

Тем не менее, выбор Java в качестве основы для систем машинного обучения может быть вполне обоснован, особенно когда ключевыми становятся надёжность, производственная готовность и долгосрочная поддержка решений. Продолжающееся развитие библиотек, открытость к современным стандартам и стремление к улучшению совместимости с другими платформами позволяют с уверенностью говорить о потенциале Java в будущем. Главный вывод заключается в том, что при грамотном подходе и учёте особенностей проекта Java может быть не только допустимым, но и эффективным инструментом в арсенале специалиста по машинному обучению.

**Библиографический список**

1. Kaur H. Comparative Weka Analysis of Clustering Algorithm's / H. Kaur, P. Verma // International Journal of Information Technology and Computer Science. – 2017. – Vol. 9, № 8. – P. 56-67. – DOI 10.5815/ijitcs.2017.08.07. – EDN LAFTHE.

2. Petrov N. Java and machine learning: opportunities and limitations / N. Petrov // Recent scientific investigation: Proceedings of LXVI International Multidisciplinary Conference, Shawnee, USA, 03 марта 2025 года. – Shawnee, USA: ООО «Интернаука», 2025. – P. 59-63.

3. Various Frameworks and Libraries of Machine Learning and Deep Learning: A Survey / Z. Wang, K. Liu, J. Li [et al.] // Archives of Computational Methods in Engineering. – 2019.

4. Анализ программных библиотек для разработки встраиваемых нейросетевых приложений / Б.В. Артемьев, А.И. Власов, Ж.О. Исроилов, С. Мулатола // Нейрокомпьютеры: разработка, применение. – 2023. – Т. 25, № 6. – С. 5-12.

5. Асылова М.А. Обзор программных библиотек для машинного обучения / М.А. Асылова // Вопросы науки 2024: потенциал науки и современные аспекты. – Анапа: Общество с ограниченной ответственностью «Научно-исследовательский центр экономических и социальных процессов» в Южном Федеральном округе, 2024. – С. 4-8.

6. Крылов А.С. Анализ средств для реализации нейронных сетей на языке программирования Java / А.С. Крылов // Молодой ученый. – 2017. – № 18(152). – С. 36-39.

7. Передерий В.А. Сравнительная характеристика библиотек машинного обучения для внедрения искусственного интеллекта в CRM-систему / В.А. Передерий, М.Л. Рысин // Инженерный вестник Дона. – 2024. – № 4(112). – С. 141-155.

## REVIEW AND COMPARISON OF JAVA LIBRARIES FOR MACHINE LEARNING

**G.G. Konovalov,** *Student*
**Supervisor:** *N.O.K. Zarbalieva, Associate Professor*
**Volgograd State University**
**(Russia, Volgograd)**

***Abstract.*** *This article provides an overview and comparative analysis of popular Java libraries for machine learning. The architectural features, functionality, performance, usability, and application areas of the following solutions are discussed: Weka, Deeplearning4j, MOA, Smile, Encog, and Tribuo. Special attention is given to library selection criteria depending on the type of task – from educational projects to industrial systems. Based on this analysis, conclusions are drawn about the role of Java in modern machine learning tasks and the prospects for its application in this field.*

***Keywords:*** *machine learning, Java, libraries, Weka, Deeplearning4j, MOA, Smile, Encog, Tribuo, neural networks, data processing, artificial intelligence.*