Куракин А.В.

Сравнение библиотек машинного обучения FANN и Neuroph для работы из Java-программ с многослойными полносвязными нейронными сетями

Московский технологический университет

(Россия, Москва)

doi:10.18411/lj-05-2018-90

idsp: 000001:lj-05-2018-90

Аннотация

Сравниваются библиотеки FANN (версии 2.2.0, FANNj) и Neuroph (версии 2.94) по функционалу, простоте подключения и надёжности для работы из Java программ. Рассмотрены особенности библиотек при подключении к Java программам. Библиотеки сравниваются по доступным функциям из java и скорости тренировки сетей.

Ключевые слова: Neuroph, FANN, Java и машинное обучение, скорость обучения сетей, библиотеки машинного обучения, ИНС

Abstract

The FANN libraries (version 2.2.0, FANNj) and Neuroph (version 2.94) are compared in terms of functionality, ease of connection and reliability for working from Java programs. Listed features of libraries are considered when connecting to Java programs. Libraries are compared by available functions from Java and network training speed.

Keywords: Neuroph, FANN, Java and machine learning, time of machine learning, library of ANN, ANN

Введение

В этой работе было проведено сравнение библиотек для работы с полносвязными многослойными нейронными сетями, работающих с программами, написанными на Java. Требовалось найти надёжную библиотеку для использования с полносвязными сетями прямого распространения в программах, написанных на Java 1.8.

Существует много библиотек (фреймворков) машинного обучения, имитирующих искусственные нейронные сети для разных языков программирования и архитектуру компьютеров и вычислительных сетей. Моделирование многослойных нейронных сетей на компьютере («нейроэмитаторы» [1]) обычно подразумевает операции над большими векторами чисел с плавающей запятой. Поэтому одной из важных характеристик библиотек является скорость работы и оптимальное использование аппаратных ресурсов.

Для сравнения были выбраны библиотеки FANN и Neuroph, так как они подходили под поставленную задачу: применение искусственной нейронной сети (ИНС) для демонстрации средствами, написанными на Java 1.8. Библиотеки сравнивались по техническим характеристикам, доступным возможностям, архитектуре представления данных в памяти и файле. Были проведены эксперименты для сравнения производительности библиотек по скорости обучения сетей.

Обзор библиотек машинного обучения для Java

Были рассмотрены некоторые библиотеки, подключаемые к Java программ: Neuroph, deeplearning4j, JavaCV (интерфейс для OpenCV), Encog, FANN (fannj интерфейс). Эти библиотеки — бесплатные (open-source).

Требовалось выбрать библиотеку с возможностью работы в 32- и 64-битными версиями операционных систем Linux (для демонстрации на разных компьютерах), и Java 1.8. Желательна возможность демонстрировать программу и на других ПК с ОС Windows. Требовалась поддержка сигмоидной функцией активации, структуры многослойной сети прямого распространения, методов обучения с учителем.

Из перечисленных библиотек были выбраны две: FANN и Neuroph. Они обе простые для подключения и работы из Java, требуют мало дополнительных файлов (меньше зависимостей от других компонентов).

Описание библиотек

Исполняемые файлы библиотеки FANN [2] требуется компилировать под каждую ОС. Имеется возможность указать конкретную модель центрального процессора (ЦП). Это мешает переносимости, но за счёт этого исполняемые файлы могут получиться эффективными. Компилируются исполняемые файлы библиотеки командой `stake`. Применяется компилятор GCC, в котором можно указать разный уровень оптимизации (-O1, -O2, -O3) и тип ЦП. Скомпилировать исполняемые файлы можно другими средствами, например VisualStudio. Библиотека FANN подключается к Java через интерфейс `fannj`, использующий технологию JNI. Так как в Java-интерфейсе (файл `fannj-0.7.jar`) библиотеки FANN представлены не все функции, реализованные в C++ версии FANN, то в этой статье будут рассматриваться только доступные из Java функции.

Имеется больше 5 отдельных программ для FANN, пригодных для визуализации сетей, но они давно не обновлялись и написаны на C++, что делает сложной их переносимость между разными платформами.

Библиотека Neuroph [3] написана целиком на Java 1.8 и использует только стандартные возможности вычислений платформы Java (JRE). Основной структурной единицей является искусственный нейрон (персептрон [1]).

Для Neuroph имеется удобная программа визуализации и работы с сетями NeurophStudio, сделанная на базе NetBeans, который тоже работает на Java. Есть краткое руководство пользователя NeurophStudio и руководство программиста. Инструкция содержит описание базовых функций создания, обучения, применения сети, а также сохранение результатов в файл. Имеется автосгенерированная средствами JavaDoc документация API.

Сравнение по характеристикам

Обе библиотеки не умеют работать в многопоточном режиме (используют 1 вычислительное ядро ЦП на одну ИНС), не используют дополнительные вычислительные устройства (видеокарту), простые в использовании. Сравнение библиотек по техническим параметрам представлены в следующей таблице 1.

Таблица 1

Сравнение по техническим параметрам

	FANN 2.2.0	Neuroph 1.94
Дата выпуска версии	24.01.2012	19.08.2017
Язык программирования	C++, Java 1.7	Java 1.8
Кроссплатформенность	частично (требуется компиляция dll/so)	да (Java 1.8)
Точность вычислений	Float (одинарная)	Double (двойная)
Формат хранения сетей и данных	текстовый	бинарный (сериализованный Object)
Функции активации	фиксированные	настраиваемые

Подключение библиотеки Neuroph простое — нужно добавить минимальный набор `jar` файлов в проект. Если нужны дополнительные функции, добавляются другие `jar` файлы с дополнительными классами (для новых типов сетей, источников данных) [4].

Для подключения FANN требуется добавить `jar` файлы к проекту, затем скомпилировать SO/DLL файлы из исходников (на C++) и разместить их в папке с программой, либо указать путь в параметре `CLASSPATH` на эти исполняемые файлы. Исполняемые библиотеки подключаются к Java по технологии JNI [5].

В таблице 2 библиотеки сравниваются по функциональности и поддерживаемым типам сетей, методам обучения ИНС.

Таблица 2

Сравнение по функциям и типам моделируемых сетей

Сравнение	FANN 2.2.0	Neuroph 1.94
Элемент архитектуры	Слой (вектор)	Нейрон (перцептрон)
Типы сетей	полносвязная многослойная сеть прямого распространения	Adaline, AutoencoderNetwork, BAM, CompetitiveNetwork, ConvolutionalNetwork, ElmanNetwork, Hopfield, JordanNetwork, Kohonen, MaxNet, MultiLayerPerceptron, NeuroFuzzyPerceptron, RBFNetwork, SupervisedHebbianNetwork, и др.
Функции активации нейрона	LINEAR, THRESHOLD, SIGMOID, GAUSSIAN, ELLIOT, SIN, COS	Gaussian, Linear, Log, Ramp, Sgn, Sigmoid, Sin, Step, Tanh, Trapezoid
Функция входа нейрона	сумма	And, Difference, EuclideanRBF, InputFunction, Max, Min, Or, Product, Sum, SumSqr, WeightedSum
Методы обучения с учителем	INCREMENT, BATCH, PROP, QUICKPROP	BackPropagation, ManhattanPropagation, MomentumBackpropagation, QuickPropagation, ResilientPropagation, SigmoidDeltaRule, и др.
Функция ошибки сети	MSE (средняя квадратичная ошибка)	MeanSquaredError (средняя квадратичная ошибка)
Компоненты (основные)	fannj-0.7.jar, jna-4.1.0.jar, ibfloatfann.so	neuroph-core-2.94.jar, slf4j-api-1.7.5.jar, библиотека logback
Сайт	www.leenissen.dk/fann/wp	www.neuroph.sourceforge.net

В таблице приведены названия функций и методов, даны из кода интерфейсов соответствующих библиотек. Их расшифровку можно получить из документации соответствующих библиотек. Для поставленной задачи (работы с полносвязными сетями прямого распространения) подходят обе библиотеки.

Преимуществом FANN является быстрая скорость обучения (эффективное использование ресурсов ЦП). FANN оптимально использует аппаратные возможности ЦП. Недостатком FANN являются платформу-зависимые скомпилированные файлы, ошибки в реализации FANN JNI (.jar), приводящие к нестабильной работе со множеством сетей. В процессе работы с FANN версии 2.2.0 обнаружили проблемы в интеграции с Java: нестабильная работа с несколькими сетями одновременно. Обучающую выборку нельзя передавать через оперативную память, только через текстовые файлы. Нельзя изменять норму обучения (англ. learning rate), по умолчанию равна 0,7 [6]. При запуске Java программы требуется указывать путь к исполняемому DLL/SO файлам, которые требуется компилировать для каждой ОС и архитектуры ЦП.

Преимуществом Neuroph является большой выбор видов нейронных сетей и их настройка, контроль процесса обучения, полная интеграция с Java. Недостатками Neuroph является медленное обучение и большое потребление памяти. Neuroph работает только с числами двойной точности (double) для многослойных сетей прямого распространения.

Сравнение скорости обучения ИНС

Сравнение скорости обучения проводилось на примере реальной задачи обучения сети на фотографиях для классификации 4-х видов поверхностей. Обучение производилось методом обратного распространения ошибки (обучения с учителем). Значения эталонных векторов для обучения сети были в диапазоне от 0 до 1. В

обучающей выборке было 90600 эталонных пар векторов (на вход и эталон выхода сети). Размер обучающей выборки составил 90 Мб.

Структура полносвязанной сети прямого распространения: входной слой состоит из 256 нейронов, скрытый слой — из 131 нейрона, выходной — из 4-х.

В FANN использовалась сигмоидная функция активации (FANN_SIGMOID) и модель многослойной полносвязной сети (по умолчанию). Точность чисел с плавающей запятой — одинарная (float). Процесс сохранения обучающей выборки в текстовый файл и дальнейший парсинг (обработка текста) был включён в оценку времени обучения, так как других вариантов передачи данных в FANNj не предусмотрено.

В Neuroph использовалась сигмоидная функция активации (TransferFunctionType.SIGMOID по умолчанию) и модель многослойной полносвязной сети (MultiLayerPerceptron). Использовались числа с плавающей запятой двойной точности (double). Обучающая выборка для Neuroph, в отличие от FANN, не нуждалась в передаче через текстовый файл.

Тестирование библиотек проводилось на персональных компьютерах (ПК), с 64 битной ОС Linux, OpenJDK version "1.8.0_162". У первого ПК был процессор AMD FX(tm)-4170 с частотой 4 ГГц и 8 Гб оперативной памяти (ОЗУ) типа DDR3, у второго компьютера — процессор Intel i5-7200U с частотой 2,7 ГГц и 6 Гб ОЗУ DDR3.

Библиотека FANN была скомпилирована в режиме generic O2 (подходит для 64-битных Intel, AMD) и O3 native (для процессора AMD). Были проведены серии испытаний на двух ПК с разными вариантами библиотек. В ходе работы выяснилось, что сборка FANN, скомпилированная с указанным типом оптимизации процессора (native), не смогла работать на другом ПК.

Скорости обучения и другие параметры, измеряемые при разных условиях и с разными сборками библиотек, представлены в таблице 3.

Таблица 3

Скорость обучения в экспериментах.

ЦП	Библиотека (сборка)	№	Подготовка выборки, мс	Время обучения, мс	Эпох	MSE (на последней эпохе)
AMD	fann o2	1	1018	75516	10	0,09484
AMD	fann o2	2	988	72946	10	0,17207
AMD	fann o2	3	913	73712	10	0,16928
AMD	fann o2	4	888	75293	10	0,11454
Intel	fann o2	1	1348	35356	4	0,03128
Intel	fann o2	2	637	60278	10	0,23660
Intel	fann o2	3	639	59589	10	0,27834
Intel	fann o2	4	625	60501	10	0,31527
AMD	fann o3	1	837	77874	10	0,08697
AMD	fann o3	2	887	77323	10	0,08169
AMD	fann o3	3	881	78241	10	0,09675
AMD	fann o3	4	864	78861	10	0,15060
Intel	fann o3	1	-	не работает	-	-
AMD	Neuroph	1	926	1272132	10	0,09768
AMD	Neuroph	2	878	1194749	10	0,09810
AMD	Neuroph	3	917	1201188	10	0,09737
AMD	Neuroph	4	847	1198684	10	0,09795
AMD	Neuroph	5	830	1256533	10	0,09802
Intel	Neuroph	1	894	749477	10	0,09939
Intel	Neuroph	2	639	984184	10	0,09900
Intel	Neuroph	3	631	1024248	10	0,09826
Intel	Neuroph	4	641	1023560	10	0,09915

Подготовка обучающей выборки заняла менее секунды. В среднем, подготовка обучающей выборки на Intel выполнялась быстрее на 20%.

Точность (репрезентативность) средних оценок была рассчитана на основе t-критерия Стьюдента с 95% уровнем достоверности [7]. Сводные результаты измерений времени представлены в таблице 4, учитывались только успешные испытания, прошедшие ровно 10 эпох обучения ИНС.

Таблица 4

Скорость библиотек, доверительные интервалы

Библиотека	ПК	Испытаний	σ (мс)	Время обучения (мс)
FANN o2	AMD	4	1242	74367 ± 1976
FANN o2	Intel	3	475	60123 ± 1182
FANN o3	AMD	4	646	78075 ± 1028
Neuroph	AMD	5	36708	1224657 ± 45579
Neuroph	Intel	4	131929	945367 ± 209929

Подготовка обучающей выборки и обучение сети на ПК с процессором Intel производились на 20%-30% быстрее, чем на ПК с AMD. Для расчёта времени при параллельной работе программ надо провести дополнительные эксперименты (результаты будут сильно отличаться).

Библиотека FANN, собранная с параметром O2 работала быстрее в 16 раз, чем Neuroph. Сборка FANN O3 работала медленнее на 5%, чем FANN O2 (однако, ожидалось, что будет работать либо быстрее, либо не повлияет на время), и на одном ПК не заработала.

Заключение

Neuroph является многофункциональной и кроссплатформенной библиотекой. Но Neuroph оказался очень медленной библиотекой для работы с искусственными нейронными сетями. Библиотека Neuroph обновлялась позже, чем последний выпуск FANN 2.2.0. FANN является быстрой библиотекой, быстрее Neuroph в 16 раз, по функциям предназначена только для многослойных сетей прямого распространения. Библиотека имеет ряд ограничений при работе из Java программ, в некоторых случаях, препятствующих достижению поставленной задачи.

Тип процессора мало влияет на скорость обучения сети, по сравнению с разницей в скорости библиотек машинного обучения. Дополнительные настройки оптимизации компиляции программы почти не повлияли на производительность, по сравнению с типом процессора. Оказалось, параметр сборки O3 замедлил работу библиотеки на 5%. Настройки компилятора повлияли на переносимость программы между разными ПК.

1. Бутусов О.Б., Редикульцева Н.И. Нейрокомпьютерное моделирование. // М.: Университет машиностроения, 2013. - 73с.
2. Fast Artificial Neural Network Library (FANN) // FANN. 2015. URL: www.leenissen.dk/fann/wp/ (дата обращения 8.04.2018)
3. Java Neural Network Framework Neuroph // Neuroph. 2017. URL: www.neuroph.sourceforge.net (дата обращения 8.04.2018)
4. J Steven Perry «Create an artificial neural network using the Neuroph Java framework» // IBM developerWorks. — Январь 2018. — 16. С. URL: www.ibm.com/developerworks/library/cc-artificial-neural-networks-neuroph-machine-learning/cc-artificial-neural-networks-neuroph-machine-learning-pdf.pdf (дата обращения 17.04.2018)
5. Sheng Liang. «The Java Native Interface: Programmer's Guide and Specification (The Java Series) 1st Edition». Amsterdam: Addison-Wesley Longman, 1999. 303 p.
6. FANN Training // FANN Training - Fast Artificial Neural Network Library (FANN). URL: http://leenissen.dk/fann/html/files/fann_train-h.html#fann_set_learning_rate (дата обращения 16.04.2018)
7. Носов В.В. Методы статистического анализа: компьютерный практикум. Учебное пособие. — 2-е изд., доп. и испр. — М.: изд-во Моск. гуманит. ун-та, 2015. — 96 с.