

рост когнитивного сопротивления [5] разрабатываемого программного обеспечения, что, в свою очередь, приводит к эпизодическим непродуктивным тратам времени при работе с программой, повреждению данных и росту раздражения пользователей.

Основные причины роста когнитивного сопротивления [5]:

- неадекватная оценка уровня квалификации пользователя;
- перегрузка пользовательского интерфейса элементами управления;
- неверное истолкование целей, к достижению которых стремится пользователь.

В той же работе, посвящённой преодолению когнитивного сопротивления, предложены следующие возможные решения:

- чёткое определение ролей пользователей и требуемого им функционала;
- максимальное упрощение интерфейса программ;
- проектирование взаимодействия должно осуществляться на ранних этапах разработки;
- все предположения разработчиков должны проверяться в тестовой эксплуатации.

В работе [6] был описан опыт разработки и сопровождения систем электронного документооборота в государственных учреждениях Республики Беларусь, обобщение которого полностью подтверждает выдвинутые тезисы. В настоящее время в разработке находится система автоматизации документооборота для Государственных инспекций по семеноводству, карантину и защите растений, которая проектируется и реализуется с использованием описанных принципов.

Список использованных источников:

1. O'Leary, Daniel L. Enterprise resource planning systems. — Cambridge University Press, 2000. — 232 с. — ISBN 0-521-79152-9.
2. Стандарт ИСО 9004-1-94. Управление качеством и элементы системы качества (п.5.1.1).
3. Мировой рынок систем электронного документооборота [Электронный ресурс] / Мировой рынок систем электронного документооборота. — Режим доступа: <http://citforum.ru/consulting/docflow/market/article1.8.200222.html>. — Дата доступа: 02.10.2015.
4. СанПиН 9-131 РБ 2000. "Гигиенические требования к видеодисплейным терминалам, электронно-вычислительным машинам и организации работы".
5. Alan Cooper. The inmates are running the asylum – Sams Publishing, USA, 2004.
6. Бычко, А. А. Автоматизированная система электронного документооборота с низким когнитивным сопротивлением / А. А. Бычко // Молодёжь в науке - 2015, Минск, Национальная Академия Наук Беларуси, Совет молодых ученых НАН РБ. — Минск, 2015.

РЕАЛИЗАЦИЯ ПАРАЛЛЕЛЬНОГО АЛГОРИТМА ОДНОСЛОЙНОГО ПЕРСЕПТРОНА ДЛЯ ПЛАТФОРМЫ OPENCL

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Жабинский С. В.

Татур М. М. - д-р техн. наук, профессор

С увеличением объёма обрабатываемых современными информационными системами данных возникает необходимость в их эффективном интеллектуальном анализе. В докладе рассматривается реализация параллельного алгоритма однослойного персептрона для платформы OpenCL, обеспечивающей выполнение программы на графическом процессоре, обладающем высокой степенью параллелизма.

Одним из простейших алгоритмов классификации является однослойный персептрон [1]. Он относится к алгоритмам с обучением с учителем и позволяет производить линейную классификацию. Принцип работы однослойного персептрона основан на модели функционирования нервной клетки — искусственного нейрона. Однослойный персептрон является нейронной сетью с одним скрытым слоем нейронов. Схема однослойного персептрона представлена на рисунке 1.

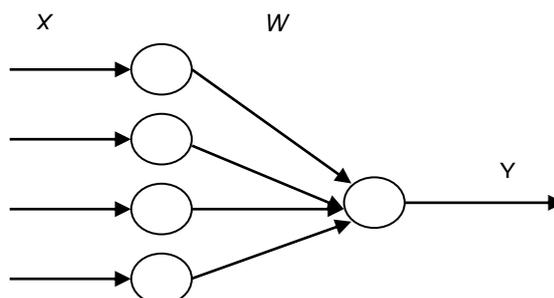


Рис. 1. Схема однослойного персептрона.

X — вектор входных значений, W — вектор весовых коэффициентов, Y — выходное значение. Выходной нейрон производит взвешенное суммирование входных значений и принятие решения о выходе персептрона. Как правило, если полученная сумма больше, либо равна 0, выход принимается равным 1, в противном случае — 0. Обучение нейрона заключается в поиске весовых коэффициентов W .

Алгоритм обучения однослойного персептрона выглядит следующим образом:

1. Вектор весовых коэффициентов W инициализируется небольшими случайными величинами.
2. Для каждого значения x_j из обучающей выборки:
 - 2.1. Определяется выход нейрона y :
 - а) Вычисляется сумма s произведения каждого признака из входного вектора признаков на соответствующий весовой коэффициент w_i .
 - б) К сумме применяется функция активации. В качестве функции активации, как правило, применяют ступенчатую функцию, принимающую значение 1, если $s \geq 0$, и значение 0, если $s < 0$.
 - 2.2. Каждый весовой коэффициент обновляется по формуле: $w_j = w_j + (l_j - y) * x_{ij}$, где l_j — класс, к которому относится данное наблюдение.
3. Вычисляется ошибка ϵ обучения на текущем шаге:
 - 3.1. Для каждого наблюдения из обучающей выборки вычисляется выход y .
 - 3.2. Разница между выходом персептрона y и истинным значением из обучающей выборки добавляется в сумму S .
 - 3.3. Ошибка ϵ вычисляется как S/N , где N — количество элементов в обучающей выборке.
4. Если ϵ меньше выбранного порогового значения, обучение прекращается. В противном случае повторяются шаги 2 — 4.

Заметим, что данный алгоритм может быть эффективно реализован параллельно. Для параллельной реализации будем использовать аппаратно-программную платформу OpenCL [2]. Данная платформа позволяет производить вычисления общего назначения на графическом процессоре [3].

Рассмотрим особенности реализации алгоритма однослойного персептрона для платформы OpenCL.

Платформа OpenCL относится к гетерогенным вычислительным системам. Управляющим элементом системы является центральный процессор (ЦПУ). В качестве ускорителя вычислений используется графический процессор (ГПУ). Общая схема такой системы представлена на рисунке 2. ЦПУ передаёт данные и команды в память ГПУ и получает обратно результат вычислений. Как правило, «бутылочным горлышком» таких систем является передача данных из основной памяти компьютера в память ГПУ. Поэтому следует избегать большого количества подобных операций. Также по этой причине, при малых объёмах обрабатываемых данных скорость выполнения параллельного алгоритма может быть меньше, чем скорость последовательного алгоритма, выполняющегося на ЦПУ.

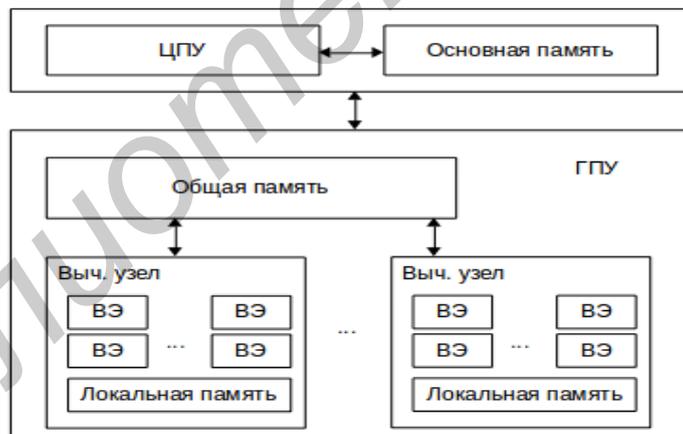


Рис. 2. Общая схема гетерогенной системы для вычислений общего назначения на ГПУ.

Для проверки эффективности параллельной реализации для платформы OpenCL были реализованы последовательный алгоритм (выполняющийся на ЦПУ), и параллельный (для OpenCL). Графики с временем обучения для данных реализаций представлены на рисунке 3.

На скорость обучения однослойного персептрона влияют размер обучающей выборки и размер пространства признаков. Тестирование производилось с различными значениями данных параметров для определения масштабируемости реализации, то есть изменение времени обучения от объёма вычислений.

Как видно из графиков, с ростом объёма вычислений, время обучения растёт практически линейно. Однако, при использовании параллельного алгоритма наклон прямой более пологий. Также следует обратить внимание на точку пересечения графиков. Данная точка показывает пороговое значение объёма данных, при котором эффективность использования реализации для OpenCL становится выше классической реализации для выполнения на ЦПУ.

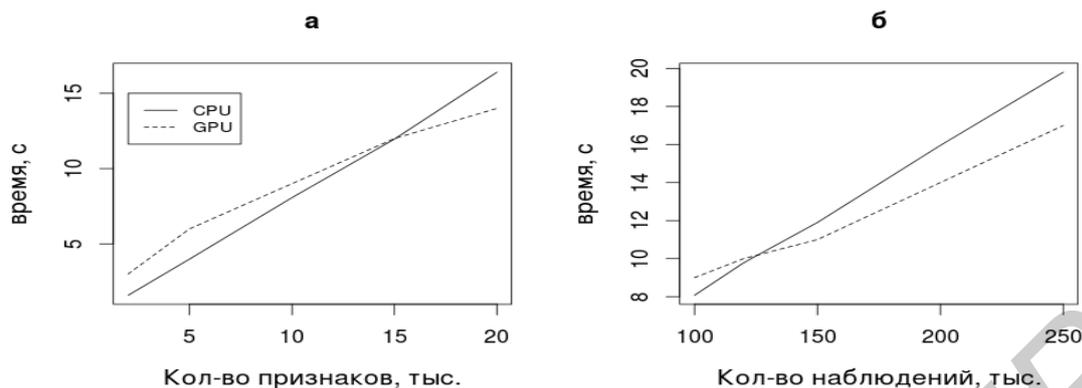


Рис. 3. Время обучения однослойного перцептрона.

Список использованных источников:

1. Беркинблит М. Б. Нейронные сети. / М. Б. Беркинблит. - Москва: МИРОС и ВЗМШ РАО, 1993. - 96 с.
2. Stone, J. E. OpenCL: A Parallel Programming Standard for Heterogeneous Computing Systems / J. E. Stone, D. Gohara, G. Shi // Computing in Science & Engineering. - 2010. - Vol. 12, iss. 3. - p. 66-73.
3. Keckler, S. GPUs and the Future of Parallel Computing. / S. W. Keckler, W. J. Dally, B. Khailany, M. Garland, D. Glasco // IEEE 2011. - Vol. 31, № 5 - p. 7-17.

МЕТОДЫ СИНТЕЗА ЗНАНИЙ, ОПЫТА И РЕСУРСОВ ПОЛЬЗОВАТЕЛЕЙ В КОНТЕКСТЕ «ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СОВМЕСТНОЙ РАБОТЫ»

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Краснов А.Ю.

Воронов А. А. – к-т. техн. наук, доцент

Тема доклада раскрывается в рамках магистерской диссертации: «Алгоритмы объединения и визуализации ресурсов пользователей на примере музыкальной социальной сети». Цель магистерской работы - разработка метода объединения ресурсов пользователей и интеграция данного метода в веб-приложение «социальная музыкальная сеть».

Актуальность исследования обусловлена развитием проектных форм организации работ (проектного менеджмента), которые являются ответом на такие вызовы времени как ускорение темпов научно-технического прогресса, и высокая скорость социально-экономических изменений [1]. Таким образом, возникает необходимость оперативного группового обсуждения ситуации и проблем, в условиях отсутствия экспертов извне, а также недостатка времени. Это обуславливает использование знаний, опыта и ресурсов ограниченного числа людей в рамках решения совместных задач. Исходя из этого можно сделать вывод, что разработка технологии совместной работы и объединения ресурсов пользователей является актуальной.

Наиболее близким аналогом нашего программного продукта будет класс «программного обеспечения для совместной работы» (groupware) – тип программного обеспечения (далее – ПО), созданного с целью поддержки взаимодействия между людьми, коллективно работающими над решением общих задач [2]. К данному типу программного обеспечения применим закон Меткалфа, который гласит, что полезность сети пропорциональна квадрату численности пользователей этой сети. Таким образом, чем больше людей используют что-либо, тем более ценным оно становится.

Ниже приведена классификация «программного обеспечения для совместной работы» (далее – ПОСР):

- совместная работа над документами (Google Docs, Office Online);
- вики технологии [3] (Wikipedia);
- системы управления проектами (Jira, eGroupWare, Citadel, Microsoft Project);
- аудио, видео конференции (Skype, Skype For Business, Google Hangouts,);
- облачные хранилища данных с возможностями совместной работы над файлами (Dropbox, OneDrive, Яндекс-диск);
- корпоративные социальные сети (Битрикс24, DaOffice, Jive, Yammer);
- базы знаний[4] (в особенности экспертные системы) (CLIPS, WolframAlpha, MYCIN);
- интернет-опросы и голосования (Examinare, Surveymonkey, Testograf);