УДК 004.67

ИСПОЛЬЗОВАНИЕ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ОБРАБОТКИ И КАТЕГОРИЗАЦИИ ИНФОРМАЦИИ

Счастливый И.М., Тарасов И.Е.

МИРЭА – Российский технологический университет, 119454, Россия, г. Москва, проспект Вернадского, 78, e-mail: ilijadavincy@gmail.com, tarasov_i@mirea.ru.

Современные системы на базе машинного обучения способны улучшать свои показатели на основе опыта, полученного благодаря решению множества однотипных задач, что позволяет избавиться от необходимости в дополнительной доработке и оптимизации продукта. Сейчас это активно используется для исследования и дальнейшего распределения информации по интересующим пользователя темам.
В данной статье исследуется возможность создания системы обработки информации с использованием алгоритмов машинного обучения. В качестве главных задач выступают определение основных этапов системы, привидение и описание решений.

Ключевые слова: анализ данных, проектирование системы, машинные алгоритмы, Программируемая логическая интегральная схема, Vivado HLS.

USING MACHINE LEARNING ALGORITHMS TO PROCESS AND CATEGORIZE INFORMATION

Schastliviy I.M., Tarasov I.E.

MIREA - Russian Technological University, 119454, Moscow, 78 Vernadskogo Avenue, Russia, e-mail: ilijadavincy@gmail.com, tarasov_i@mirea.ru.

Modern systems based on machine learning are able to improve their performance based on the experience gained through solving many similar problems, which eliminates the need for additional refinement and product optimization. Now it is actively used for research and further distribution of information on topics of interest to the user. This article explores the possibility of creating an information processing system using machine-learning algorithms. The main tasks are the definition of the main stages of the system, the ghost and the description of solutions.

Key words: data analysis, system design, machine algorithms, field-programmable gate array, Vivado HLS.

Введение

Современные системы на базе машинного обучения способны улучшать свои показатели на основе опыта, полученного благодаря решению множества однотипных задач, что позволяет избавиться от необходимости в дополнительной доработке и оптимизации продукта. Сейчас это активно используется для исследования и дальнейшего распределения информации по интересующим пользователя темам.

В данной статье исследуется возможность создания системы обработки информации с использованием алгоритмов машинного обучения, которая отличалась бы от своих конкурентов большей доступностью в наши дни сложно обнаружить сферу жизни общества, в которой не используются информационные системы. Базы данных, серверы – всё это уже давно активно используется повсеместно.

Уже не новость, что отдельные компании-гиганты, например, Google, Baidu и IBM, уже долгое время работают над использованием машинных алгоритмов во всех сферах человеческой жизнедеятельности - системы поиска заболеваний и их дальнейшего лечения, улучшение поиска информации, системы, которые могут спать и видеть сны, боты в социальных сетях, поведение которых практически не отличимое от человеческого и многое другое. Основным недостатком таких разработок является дороговизна: найм исследователей, покупка необходимого технического и программного обеспечения — всё это стоит очень больших денег. В связи с этим лишь малая часть компаний может позволить себе исследования подобного типа.

Из них стоит отметить Pinterest. Этот социальный интернет-сервис отличается тем, что может самостоятельно группировать содержащуюся в нём информацию по отдельным категориям и рекомендовать пользователю те

публикации, которые с наибольшей вероятностью смогут заинтересовать его. Подобные алгоритмы интересны для любой компании. Целью данной статьи будет исследование использования алгоритмов машинного обучения для обеспечения доступности систем фильтрации, классификации и анализа поступающих данных.

Материальная составляющая

Рассмотрим предполагаемый процесс обработки информации. Компания направляет полученные от пользователей данные на вход в систему. После отделения не интересующей нас информации, включающей в себя спам, словесные распри и ссылки на небезопасные источники, материал поступает на распределение по определённым группам интересов, заданным владельцем.

Рис. 1. Примерная схема взаимодействия составляющих системы.

Компания сама выбирает способ сбора информации в соответствии со своими целями. В качестве примера будет рассматриваться самый распространённый на данный момент способ. Сейчас даже малый бизнес владеет собственным сайтом с системами авторизации. В среднем каждый пользователь предоставляет фамилию и имя, данные о своей электронной почте, телефон и другие контактные данные. После прохождения авторизации пользователь может оставить отзыв на продукт в целом или определённую его составляющую. Учитывая то, что необходимость в использовании алгоритмов машинного обучения возникает только при большом объёме данных, при категоризации которых возникает сложность, нам потребуется большое хранилище для них. В связи с этим нам потребуется выделенный сервер. Полученная на сервер информация поступает в выделенную базу данных, где и будет храниться информация, полученная от пользователя. Следующим шагом уже является выполнение алгоритмов машинного обучения. Для машинного обучения самыми важными критериями являются количество ядер и объём памяти. Графические процессоры, рассчитанные на большое количество параллельных вычислений, превосходят центральные процессоры, но их высокая стоимость не подходит в рамках поставленной задачи. Если клиент готов пожертвовать скоростью обучения ради повышения быстродействия и уменьшения энергопотребления, идеальным выбором будут программируемые логические интегральные схемы (ПЛИС). ПЛИС – это сеть из нескольких миллионов программируемых блоков, связь между которыми программным образом настраивается пользователем.

Благодаря отсутствию кэшей загрузки при низкой тактовой частоте производительность заметно улучшается. Отдельным плюсом стоит отметить сравнительно низкую себестоимость. При желании можно компенсировать низкую скорость обучения с помощью систем автоматического проектирования(САПР), которые синтезируют параллельно работающие аппаратные структуры в ПЛИС из высокоуровневого кода на C/C++/SystemC . В частности, одной из самых выдающихся САПР является система Vivado High Level Synthesis. Её новые инструменты на базе синтеза высокого уровня позволяют разработчикам эффективно создавать и проверять оборудование, что обеспечивает контроль над оптимизацией проектной архитектуры.

Дальнейшие действия с информацией завися уже от нужд компании. При желании отфильтрованные данные можно направить на вход другого машинного алгоритма, задачей которого будет уже предсказание будущих тенденций в определённой сфере. Также отфильтрованные данные с техническими отзывами по продукту могут быть направлены в департамент разработок и исследований компании, в котором на основе этих выводов откорректируют существующие огрехи, добавят дополнительные функции, повышающие удобство пользования продуктом, или же обеспечат доступность продукта на новой платформе.

Первичная обработка информации

При отделении лишней информации стоит учитывать её составляющие: спам и хам. Оба термина означают нежелательную рассылку, однако “спам” относиться к рассылкам, на получение которых пользователь не подписывался, тогда как хам относится уже к рассылке, на которую пользователь дал своё разрешение, пусть даже не осознавая этого. Пользователь непроизвольно соглашается на получение рассылки при установке или же обновлении программного обеспечения; оформлении подписки на новый сервис. Каждый из этих подтипов нежелательной информации обладает своим набором часто используемых слов и фраз. Некоторые из них человек может обнаружить самостоятельно, без помощи системы. Обычно для этого используются инструменты

визуализации. Например, при использовании Wordcloud пользователь увидит изображение со словами, размер каждого из которых зависит от частоты его использование в исследуемом материале. Это помогает сразу выделить такие явные слова, как “бесплатно”, “продукт”, “скидка” и тому подобное.

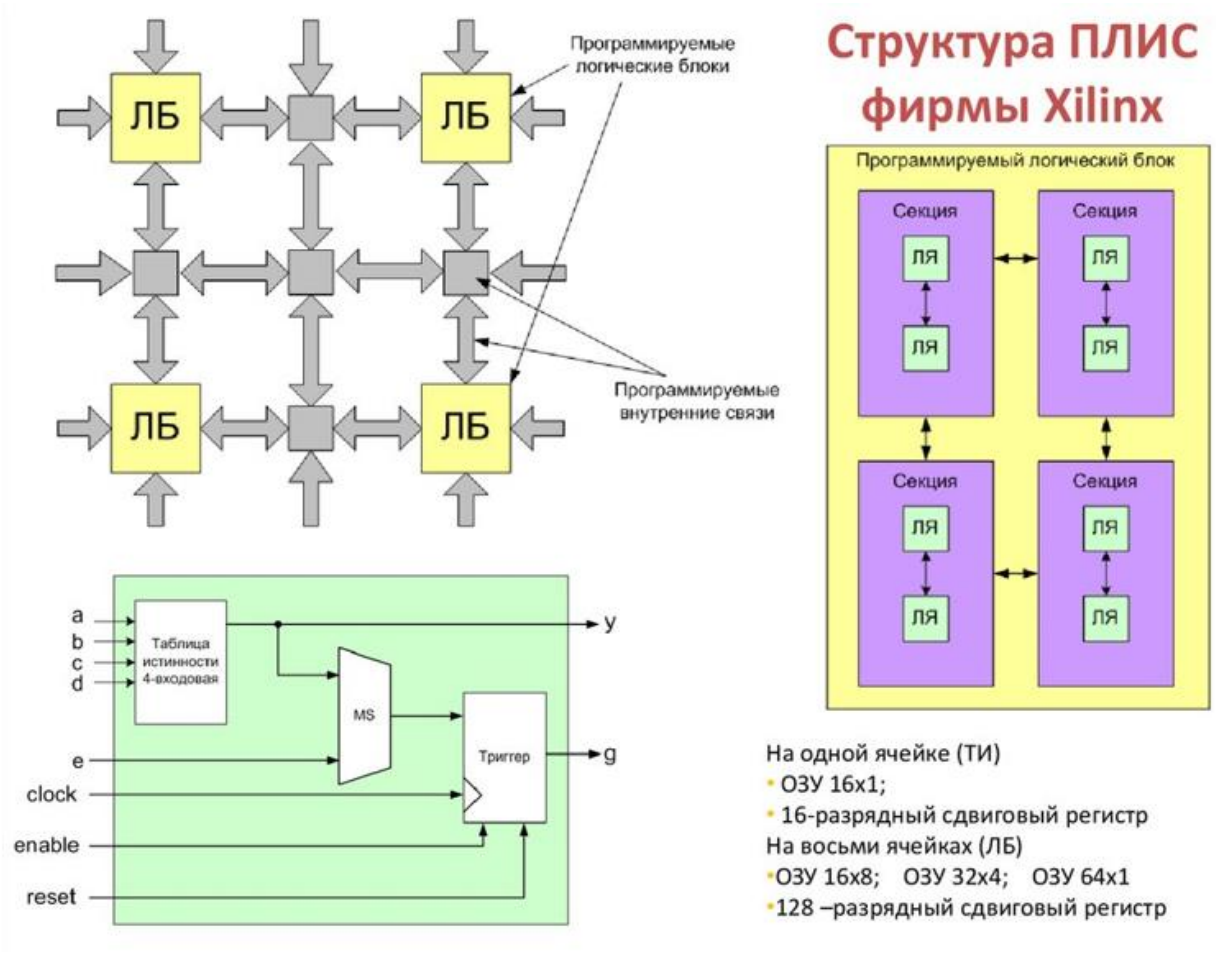

Рисунок 2 Структура ПЛИС фирмы Xilinx

Рис. 3. Визуализация спам почты с помощью Wordcloud

После этого следует перейти к подготовке информации для дальнейшей обработки. Чаще всего ограничиваются удалением пунктуации, пробелов, ссылок, номеров; конвертацией всех символов в строчные или же заглавные буквы. Иногда происходит дальнейшая фильтрация, которая включает в себя удаление аффиксов и конвертации слов в их базовую форму.

```
# Результат работы алгоритма конвертации слов в их базовую форму
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
dirty_text = "He studies in the house yesterday, unluckily, the fans breaks down"
def word_lemmatizer(words):
lemma_words = [lemmatizer.lemmatize(o) for o in words]
return " ".join(lemma_words)
clean_text = word_lemmatizer(dirty_text.split(" "))
clean_text
#Результат
'I study in the house yesterday, unluckily, the fan break down'
```

В связи с тем, что алгоритму для работы требуются числа, необходимо создать прослойку, трансформирующую материал в числовой формат. Это происходит с помощью ряда мер, направленных на решение отдельных подзадач. Первым делам благодаря TfidfVectorizer происходит обычная запись слов с подсчётом частоты их использования в тексте. При желании можно использовать TfidfVectorizer, что позволит искусственно уменьшить учёт предлогов и других часто попадающихся слов. Наконец, Word Embedding подразумевает создание векторных связей между связанными словами для организации определённого порядка хранения слов. Чем ближе два слова по смыслу, тем меньше будет дистанция между векторами.

Благодаря обширному количеству библиотек для обучения системы обнаружению нежелательной информации, самым логичным решением для обработки данных будет выступать использование обучения с учителем. В этом случае каждый объект описывается парой показателей данные - целевое значение. В результате прохождения этого алгоритма будет выявлена определённая зависимость между указанными показателями. Среди конкурентов самым подходящим для задачи обнаружения нежелательной информации показал себя наивный байесовский классификатор, основанный на применении теоремы Байеса со строгим предположением независимости. Наивный Байес получил своё название из-за того, что алгоритм предполагает, будто каждая входная переменная независимая, что является не соответствующим реальным данным предположением. Этот метод выгоден тем, что для его обучения требуется сравнительно малое количество информации. Благодаря этому методу достигается точность, близкая к 95%. В итоге данного процесса мы получаем данные, которые практически лишены спама.

Категоризация информации

Теперь нам остаётся только распределить полученные данные по интересующим нас категориям. В случае со спамом мы могли предоставить машине конкретные примеры как нежелательных сообщений, так и сообщений, несущих смысловую нагрузку. В случае же с категоризацией по темам все немного иначе. Мы не можем в полной мере предсказать, каких тем будет касаться пользователь при написании своего отзыва. Если мы хотим получить действительно полезный алгоритм, который можно использовать вне зависимости от области нашей работы, нам нужно использовать классификацию с применением обучения без учителя. В таком случае алгоритмы будут пытаться найти определённые зависимости и структуры отдельных наборов данных без использования каких-либо внешних указателей. Стоит отметить, что для человека эти зависимости могут казаться не самыми логичными. Однако у этого подхода есть определённый недостаток. Его эффективность растёт вместе с ростом поступающей в него информации, из чего следует, что компания, желающая воспользоваться данным решением, должна будет обладать определённой клиентской базой. Для выполнения задачи распределения по группам необходимо рассмотреть кластеризацию. Наш выбор конкретного подвида кластеризации ограничивает отсутствие предварительных данных по итоговому количеству кластеров.

Самой подходящей является иерархическая кластеризация, где с каждой итерацией последовательно объединяются самые похожие объекты до получения итоговой дендрограммы. Это идеально подходит для ситуаций, в которых текст содержит информацию сразу по нескольким направлением. Рассмотрим работу этого алгоритма на конкретном примере.

Таблица 1.

Работа алгоритма иерархическая кластеризация на тестовом наборе данных

Итерация	Совмещённые кластеры	Все кластеры	Общее число кластеров
0	0	2,3,6,8,12,16,34,35,50	9
1	(2 -3), (34-35)	(2 -3), (34-35),6,8,12,16,50	7
2	(6-8)	(2 -3), (34-35),(6-8),12,16,50	6
3	[(2 -3)-(6-8)],(12-16)	[(2 -3) - (6-8)],(12-16),(34-35),50	4
4	[(2 -3)-(6-8)-(12-16)]	[(2 -3)-(6-8)-(12-16)] ,(34-35),50	3
5	[(2 -3)-(6-8)-(12-16)-(34-35)]	[(2 -3)-(6-8)-(12-16)-(34-35)], 50	2
6	Все	Все	1

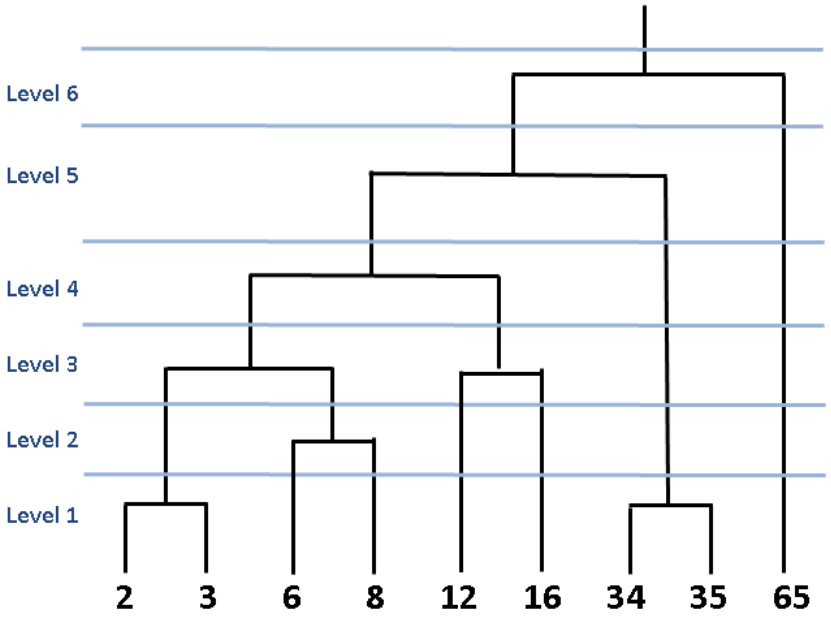

Рисунок 4 Результат работы алгоритма иерархическая кластеризация на тестовом наборе данных

Тем не менее, стоит упомянуть недостатки:

1) Сам по себе алгоритм простой, однако является вычислительно сложным. Для работы с Big Data он не подходит.

2) У алгоритма возникают проблемы при отображении дендрограммы с большим числом объектов.

Стоит отметить, что ввиду рассматриваемой задачи эти недостатки не являются критичными: для малого и большей части среднего бизнеса объёмы данных. Соразмерные Big Data не требуются, а от алгоритма нам нужно именно распределение по темам. Его отображение будет вызываться редко. Тем не менее, в ситуациях, когда эти недостатки являются критичными, можно использовать кластеризацию DBSCAN. Этот алгоритм группирует вместе плотные области данных и может находить кластеры даже очень сложных форм.

Недостатки:

1) Часть точек не относятся к каким-либо кластерам и считаются шумом.

2) При низкой плотности данных возникают проблемы с кластеризацией.

Как можно было заметить, этот алгоритм требует большего количества данных от пользователя, в связи с чем и не является приоритетным в решении данной задачи.

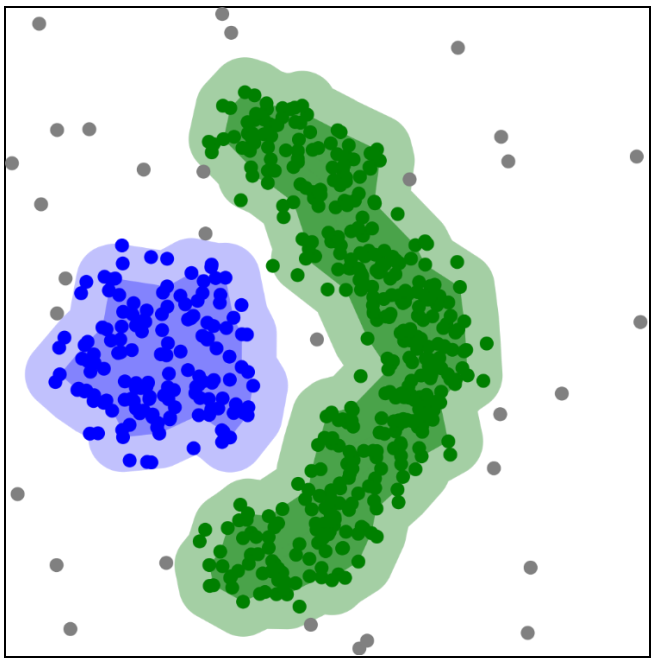

Рис. 5. Результат работы DBSCAN.

Заключение

Таким образом, спроектированное решение отвечает всем требованиям, поставленным к системам кластеризации, оставаясь при этом материально доступным. Себестоимость этих разработок ограничивается зарплатой сотрудников, арендой сервера и покупкой дополнительных производственных мощностей. Во многих случаях компания, рассчитывающая на внедрение технологии машинного обучения в свои технологические процессы, уже будет обладать всеми необходимыми для этого ресурсами. После внедрения же компания сможет вернуть себе деньги, потраченные на разработку, удешевить дальнейшие исследования, связанные с машинными алгоритмами.

Список литературы:

1. 10 Companies Using Machine Learning in Cool Ways // [Электронный ресурс]: https://www.wordstream.com/blog/ws/2017/07/28/machine-learning-applications (Дата обращения 12.04.2021)
2. Mipsology: The Future of FPGA-Based Machine Learning // [Электронный ресурс]: https://www.xilinx.com/support/documentation/product-briefs/mipsology-aws-f1.pdf (Дата обращения 13.04.2021)
3. Ham v Spam: what's the difference? // [Электронный ресурс]: https://blog.barracuda.com/2013/10/03/ham-v-spam-whats-the-difference/ (Дата обращения 12.04.2021)
4. How To Design A Spam Filtering System with Machine Learning Algorithm // [Электронный ресурс]: https://towardsdatascience.com/email-spam-detection-1-2-b0e06a5c0472 (Дата обращения 12.04.2021)
5. Чудесный мир Word Embeddings: какие они бывают и зачем нужны? // [Электронный ресурс]: https://habr.com/ru/company/ods/blog/329410/ (Дата обращения 11.04.2021)
6. Де Прадо. «Машинное обучение. Алгоритмы для бизнеса» М. Прогресс книга 2019
7. Pedro Domingos. «The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World» Манн 2016
8. Флах Петер. «Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных» ДМК Пресс 2015
9. Машинное обучение: рекомендательные системы // [Электронный ресурс]: https://vc.ru/ml/132779-mashinnoe-obuchenie-rekomendatelnye-sistemy (Дата обращения 11.04.2021)
10. Clustering in Machine Learning // [Электронный ресурс]: https://www.geeksforgeeks.org/clustering-in-machine-learning/ (Дата обращения 11.04.2021)
11. Unsupervised Machine Learning: Clustering Analysis // [Электронный ресурс]: https://towardsdatascience.com/unsupervised-machine-learning-clustering-analysis-d40f2b34ae7e (Дата обращения 11.04.2021)

References

1. 10 Companies Using Machine Learning in Cool Ways // [Electronic resource]: https://www.wordstream.com/blog/ws/2017/07/28/machine-learning-applications (Date of treatment 04/12/2021)

2. Mipsology: The Future of FPGA-Based Machine Learning // [Electronic resource]: https://www.xilinx.com/support/documentation/product-briefs/mipsology-aws-f1.pdf (Date of access 13.04.2021)

3. Ham v Spam: what's the difference? // [Electronic resource]: https://blog.barracuda.com/2013/10/03/ham-v-spam-whats-the-difference/ (Date of treatment 04/12/2021)

4. How To Design A Spam Filtering System with Machine Learning Algorithm // [Electronic resource]: https://towardsdatascience.com/email-spam-detection-1-2-b0e06a5c0472 (Date of treatment 04/12/2021)

5. The wonderful world of Word Embeddings: what are they like and why are they needed? // [Electronic resource]: https://habr.com/ru/company/ods/blog/329410/ (Date of treatment 04/11/2021)

6. De Prado. Machine Learning. Algorithms for business "M. Progress book 2019

7. Pedro Domingos. "The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World" Mann 2016

8. Flach Peter. Machine Learning. The science and art of building algorithms that extract knowledge from data "DMK Press 2015

9. Machine learning: recommender systems // [Electronic resource]: https://vc.ru/ml/132779-mashinnoe-obuchenie-rekomendatelnye-sistemy (Date of access 11.04.2021)

10. Clustering in Machine Learning // [Electronic resource]: https://www.geeksforgeeks.org/clustering-in-machine-learning/ (Date of access 11.04.2021)

11. Unsupervised Machine Learning: Clustering Analysis // [Electronic resource]: https://towardsdatascience.com/unsupervised-machine-learning-clustering-analysis-d40f2b34ae7e (Date of access 11.04.2021)