**УДК 004.6**

**Исследование требований к вычислительной сети от методов анализа больших данных инфокоммуникационной системы**

**Иванов М.Т.1** (студент), **Карпов К.Д.1,2**

**Научный руководитель – к.т.н., доцент Карпов К.Д.1,2**

*1 – Университет ИТМО*

*2 – ЦНИИ Электроприбор*

*e-mail: \*контактная почта авторов\**

В работе проведен анализ наиболее распространенных методов обработки больших данных, приведены преимущества и недостатки каждого метода, а также исследованы требования к вычислительным сетям для реализации методов. Приведены показатели оценки эффективности вычислительных сетей. На основе существующих алгоритмов анализа данных предложены и реализованы в виде прототипов модели вычислительных сетей для анализа больших данных.

**Ключевые слова**

Вычислительная система, большие данные, анализ данных, распределенные вычисления, эффективность обработки.

Большие данные – это понятие, без которого в настоящее время не обходится практически ни одна отрасль науки или промышленности. И на увеличение эффективности анализа больших данных направлены многие исследования и разработки. Все работы в этом направлении можно разделить на два направления: разработка более совершенных устройств и вычислительных мощностей, которые позволят физически ускорить вычисления над большими данными, и усовершенствование самих алгоритмов обработки больших данных, подразумевающее сокращения числа или упрощение операций обработки данных.

Для анализа методов кластеризации информации с разработаны метрики точности для измерения качества кластеризации, которые делятся на внутренние сравнивающие данные внутри кластеров и различия кластеров между собой, и внешние, определяющие соответствие практических кластеров эталонным, то есть заранее известным. В зависимости от устанавливаемой задачи по анализу больших данных может применяться тот или иной из описанных методов кластеризации [1, 2].

При проведении анализа эффективности алгоритмов и вычислительных сетей в первую очередь внимание обращается на размер оперативной памяти на ядро и на скорость работы процессора [3].

При определении средств анализа данных необходимо четко представлять цели задач и ожидаемые результаты. «Алгоритм обработки должен иметь минимальную зависимость от объема поступающих данных и должен позволять оперативно обрабатывать постоянно увеличивающиеся объемы данных, либо должен подразумевать возможность масштабирования в зависимости от желания пользователя» [4].

Многие исследования направлены на сравнение алгоритмов анализа больших данных. Особенно внимание акцентируется на такие алгоритмы как GraphLab, Galois, GraphX. В табл. 1 для них приводятся сведения о требовательности и гибкости при реализации.

Таблица 1

**Сравнение программных средств работы с большими сетями**

|  |  |  |  |
| --- | --- | --- | --- |
| **Программа** | **GraphLab** | **Galois** | **GraphX** |
| Каркас | MapReduce | MapReduce | ApacheSpark |
| Производительность | Высокая при числе вершин до 100000, далее значительно падает | Высокая при числе вершин до 100000, далее значительнопадает | Высокая |
| Требовательность кресурсам | Не требовательна | Не требовательна | Очень требовательна |
| Реализация алгоритмов | Гибкая настройка, поддержка разных языковпрограммирования | Использует толькособственные структуры данных | Гибкая настройка,поддержка разныхязыков программирования |

Для оценки эффективности вычислительных сетей применяются такие параметры, как:

1. Общая производительность смоделированной вычислительной сети.

2. Время проведения анализа в разной конфигурации модели – относительный показатель, при котором сравнивается время анализа при разных конфигурациях модели с целью выявления оптимальной.

3. Уровень использования компонентов сети на протяжении выполнения эксперимента.

4. Отношение используемых и простаивающих вычислительных узлов.

5. Отношение выполняющихся и ожидающих заданий – рассчитывается на основе объемов данных, находящихся внутри сети во время их обработки и во время ожидания очереди на обработку.

В рамках текущей работы под общей производительностью понимается характеристика скорости обработки данных сетью. Для расчета скорости необходимо замерить время эксперимента и объем данных, проходящий через вычислительную сеть, что легко реализуемо как при проверке моделей, так и на практике.

Вводными параметрами являются:

* количество узлов;
* объем данных для анализа;
* параметры узлов:
	+ скорость обработки;
	+ объем загружаемых данных;
	+ надежность узла (1-p, где p – вероятность отказа).

Для реализации выбранных алгоритмов используется ПО AnyLogic, общий вид модели представлен на рис. 1.



Рис. 1. Позитив [1]

Главными выходным значениями являются скорость обработки и объем обработанных данных на момент времени t. Расчет времени обработки данных t проводится по формуле (1).

, (1)

где – объем данных на одном узле, – скорость обработки на одном узле. В общем случае объем данных на одном узле рассчитывается по формуле (2).

, (2)

где *D* – значение общего объема данных в сети. Однако, возможны доработки алгоритма, позволяющие нагружать узлы пропорционально их производительности. Скорость обработки зависит от технических характеристик каждого из узлов.

Для моделирования режима ApacheSpark при расчете времени работы каждого узла вводится параметр объема оперативной памяти M. Таким образом, время обработки данных рассчитывается по формуле (3).

, (3)

где – время работы узла, – скорость обработки данных, – объем данных.

Дальнейшие исследования в этом направлении должны быть направлены на расширение описательной базы моделей с целью их более точной и гибкой настройки. Особенное внимание должно быть уделено проектированию граф-ориентированных архитектур вычислительных сетей, так как от алгоритма построения сети могут зависеть её качественные характеристики – надежность, безопасность и эффективность её работы.

**Литература**

1. Сюй Д., Тянь Ю. Всесторонний обзор алгоритмов кластеризации //Анналы науки о данных. – 2015. – Т. 2. – №. 2. – С. 165-193.
2. Лутов А., Хаяти М., Кудре-Мору П. Оценка точности алгоритмов кластеризации с перекрытием и множественным разрешением на больших наборах данных // Международная конференция IEEE 2019 по большим данным и интеллектуальным вычислениям (BigComp). – IEEE, 2019. – С. 1-8.
3. Кушнеревич А. Г. Оценка эффективности платформы распределенного анализа данных hadoop в сравнении с платформой SPARK для целей сетевой безопасности //Региональная информатика и информационная безопасность. – 2017. – С. 228-230.
4. Аксютина Е. М., Белов Ю. С. Обзор архитектур и методов машинного обучения для анализа больших данных //Электронный журнал: наука, техника и образование. – 2016. – №. 1. – С. 134–141.