*На правах рукописи*

**Успенский Михаил Борисович**

**ПРИМЕНЕНИЕ ОНТОЛОГИЧЕСКИХ МОДЕЛЕЙ И МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ОБНАРУЖЕНИЯ НЕИСПРАВНОСТЕЙ В СИСТЕМЕ ХРАНЕНИЯ ДАННЫХ**

Специальность 05.13.01 – Системный анализ, управление и обработка

информации (технические системы)

**АВТОРЕФЕРАТ**

диссертации на соискание ученой степени

кандидата технических наук

Санкт-Петербург – 2019

Работа выполнена в федеральном государственном автономном образовательном учреждении высшего образования «Санкт-Петербургский политехнический университет Петра Великого».

|  |  |
| --- | --- |
| Научный руководитель: | кандидат технических наук, доцент |
|  | **Ицыксон Владимир Михайлович** |
|  | СПбПУ, |
|  | директор Высшей школы интеллектуальных систем и суперкомпьютерных технологий |
|  |  |
| Официальные оппоненты: |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Ведущая организация: | **Акционерное общество «Концерн «Научно-производственное объединение «Аврора»** |

Защита состоится \_\_\_ \_\_\_\_\_\_\_ 20\_\_ г. в \_\_\_ часов на заседании диссертационного совета по защите диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук Д 002.199.01, созданного на базе Федерального государственного бюджетного учреждения науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН) по адресу: 199178, Санкт-Петербург, 14-я линия В.О., 39.

Факс: (812)-328-44-50, тел.: (812)-328-34-11.

С диссертацией можно ознакомиться в библиотеке Федерального государственного бюджетного учреждения науки Санкт-Петербургского института информатики и автоматизации Российской академии наук по адресу: 199178, Санкт-Петербург, В.О., 14 линия, д. 39 и на сайте <http://www.spiiras.nw.ru/dissovet/>

Автореферат разослан «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 20\_\_ года.

|  |  |  |
| --- | --- | --- |
| Ученый секретарь совета  диссертационного совета Д 002.199.01,  кандидат технических наук |  | Зайцева  Александра  Алексеевна |

Общая характеристика работы

**Актуальность темы исследования.** Проблема своевременного обнаружения неисправностей в вычислительных системах вообще и системах хранения данных (СХД) в частности имеет на данный момент большое значение в связи с повсеместным распространением таких систем во всех сферах человеческой деятельности. Так, СХД данных активно применяются в банковской сфере, сфере телекоммуникаций, государственных органах, оборонно-промышленном комплексе, топливно-энергетическом комплексе, образовании и науке. СХД применяются для хранения персональных данных, финансовых документов, проектной документации, обучающих материалов.

Современные СХД – это сложные аппаратно-программные комплексы, включающие в себя множество взаимосвязанных подсистем, модулей и компонентов. Ключевыми требованиями к характеристикам СХД являются:

- отказоустойчивость;

- обеспечение высокой доступности данных;

- обеспечение сохранности данных;

- производительность;

- масштабируемость.

Своевременное обнаружение сбоев в СХД необходимо для обеспечения сохранности данных, отказоустойчивости СХД, обеспечения бесперебойного доступа к данным. Неспособность обеспечить какое-либо из этих требований может привести к простою оборудования, материальным убыткам, частичной или полной потере данных. При этом, сложность диагностики таких систем постоянно возрастает, потому что по мере увеличения потребности в емкости хранимых данных, растет и сложность самих СХД, так как возрастают требования к аппаратным и программным компонентам, обеспечивающим скорость доступа к данным, различные варианты размещения данных и резервирование основных блоков системы. Вследствие этого всё большее и большое число сбоев в СХД возникает при взаимодействии различных компонентов, не являющихся непосредственными носителями информации. Выход из строя контроллера хранения, хотя и не затронет диски напрямую, но приведёт к деградации производительности системы, что, в свою очередь, может привести к недоступности данных в текущий момент времени.

Актуальность применяемых в настоящем исследовании подходов к обнаружению сбоев в СХД обуславливается возрастающей популярностью применения методов машинного обучения в задачах диагностики и прогнозирования сбоев в вычислительных системах, что подтверждается проведенным анализом предметной области.

**Степень разработанности темы исследования.** Несмотря на то, что задача обнаружения неисправностей в вычислительных системах хорошо проработана в многочисленных публикациях, исследования в области обнаружения неисправностей в СХД как правило в большей степени направлены на диагностику носителей информации, как наиболее уязвимой и хорошо изученной части системы. При этом отсутствует необходимый для решения задачи обнаружения проблем взаимодействия различных компонентов СХД комплексный подход, основанный на совместном использовании в рамках одной диагностической модели входных данных разного рода, в том числе из журналов СПО СХД.

**Цель исследования** заключается в разработке эффективных методов обнаружения неисправностей в СХД с использованием инструментов онтологического моделирования и методов машинного обучения, обеспечивающих раннее детектирование нештатных ситуаций.

Для достижения поставленной цели были сформулированы и решены следующие задачи:

- выполнение анализа существующих решений и научных исследований в области диагностики вычислительных систем, определение их сравнительных характеристик, формулирование требований к методам обнаружения неисправностей в СХД и реализующему их программному обеспечению;

- разработка диагностической модели СХД, позволяющей определять отношение между параметрами и состояниями СХД путем использования детерминированных связей и связей, основанных на использовании предварительно обученных алгоритмов машинного обучения для установления соответствия между параметрами и состояниями СХД.

- разработка и обоснование преобразования диагностической модели к упрощенному графовому виду для её применения в составе программного обеспечения и формализовать подход к построению такого графа.

- разработка алгоритма обнаружения неисправностей в СХД на основании классификации текстов журналов СПО СХД.

- экспериментальное обоснование использования параметрического описания СХД на основании текстов журналов СПО СХД и выбор алгоритма классификации.

- проведение экспериментальной проверки эффективности применения разработанных методов и средств.

**Объектом исследования** являются СХД, возникающие в них неисправности и взаимосвязи между состоянием отдельных компонентов СХД.

**Предметом исследования** являются модели и методы обнаружения неисправностей в СХД.

**Научная новизна** положений, выносимых на защиту, заключается в следующем:

- В настоящей работе впервые используется подход к построению диагностической модели СХД, основанный на построении онтологической модели с возможностью задания условных связей между объектами онтологии путем использования моделей машинного обучения.

- Используемый алгоритм обнаружения неисправностей на основании классификации текстов журналов СПО СХД позволяет выполнять обнаружение неисправностей в СХД без детального анализа структуры журналов и сообщений.

- В настоящей работе впервые используется подход к обнаружению сбоев в СХД на основании классификации текстов журналов СПО СХД с применением алгоритмов машинного обучения и методов онтологического моделирования.

- Используемый в настоящей работе переход от онтологической модели к графовому виду позволяет обеспечить её эффективное использование в составе диагностического программного обеспечения.

**Теоретическая и практическая значимость исследования.**

Предложенный в настоящей работе подход развивает научные основы построения диагностических моделей для автоматического и автоматизированного обнаружения неисправностей в СХД на основании онтологического подхода с расширением аппарата методов онтологического моделирования за счёт применения связей между объектами онтологии, задаваемых при помощи внешних процедур, реализующих применение алгоритмов машинного обучения.

Практическая значимость исследования заключается в разработке и практической реализации в виде программного обеспечения метода обнаружения сбоев в СХД, использование которого в процессе функционирования СХД позволит повысить надежность хранения данных и обеспечить к ним бесперебойный доступ.

**Методология и методы диссертационного исследования** базируются на междисциплинарном подходе с применением методов теории графов, теории распознавания образов, методах онтологического моделирования и семантических сетей, методов обработки естественного языка, в том числе с использованием средств машинного обучения.

**Положения, выносимые на защиту:**

1) Онтологическая модель СХД, определяющая отношение между значениями параметров СХД, состояниями компонентов СХД и состоянием СХД в целом при помощи структурных связей между объектами и условных связей, определяемых внешними процедурами, позволяющая диагностировать большее количество неисправностей относительно существующих решений.

2) Метод обнаружения неисправностей в СХД, основанный на классификации предобработанных текстов журналов СПО СХД, позволяющий, в отличие от существующих решений, обнаруживать неисправности без детального анализа структуры журналов.

3) Комплексный метод обнаружения неисправностей в СХД, основанный на совместном использовании онтологической модели СХД и алгоритма классификации текстов журналов СПО СХД с использованием машинного обучения позволяет повысить количество обнаруживаемых неисправностей относительно существующих решений.

4) Методика преобразования онтологической модели СХД в упрощенную графовую модель, позволяющая снизить сложность онтологической модели и сократить её размер путём применения правил преобразования классов, объектов и свойств онтологии в узлы и дуги ориентированного графа.

**Обоснованность и достоверность** научных результатов достигается за счёт использования апробированного математического аппарата, соответствием экспериментальных данных теоретическим предположениям, а также успешным применением практических результатов в экспериментальном образце аппаратно-программного комплекса обнаружения сбоев в СХД.

**Реализация результатов работы.** Результаты, полученные в настоящем исследовании, были реализованы в ходе выполнения работ по разработке программно-аппаратного комплекса прогнозирования сбоев, выполненных при финансовой поддержке Министерства образования и науки Российской Федерации в рамках Федеральной целевой программы «Исследования и разработки по приоритетным направлениям развития научно-технического комплекса России на 2014-2020 годы», соглашение о предоставлении субсидии от 03.10.2017 г. № RFMEFI58117Х0023.

**Апробация полученных результатов.** Полученные в ходе диссертационного исследования результаты были представлены и обсуждены на следующих российских и международных конференциях:

1) Topical Problems of Architecture, Civil Engineering and Environmental Economics, St. Petersburg, 2018.

2) XXIII Международная научно-практическая конференция. «Системный анализ в проектировании и управлении». Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, 2018.

3) Международная конференция по мягким вычислениям и измерениям, Санкт-Петербург, 2018.

4) International Conference Cyber-Physical Systems and Control, St. Petersburg, 2019.

5) 17th IEEE International Symposium on Intelligent Systems and Informatics, Serbia, Subotica, 2019.

6) 33rd International Business Information Management Association Conference (IBIMA), Spain, Madrid, 2019.

**Публикации.** Основные результаты по теме диссертационной работы было опубликовано ХХ научных работ и приравненных к ним публикаций, в том числе 3 в журналах из списка рекомендуемых ВАК и 9 в журналах, индексируемых в SCOPUS и Web of Science. Кроме того, было зарегистрировано 9 программ для ЭВМ.

**Личный вклад.** Все результаты, выносимые на защиту в настоящей диссертации, получены автором лично.

**Структура и объем диссертационной работы.** Диссертация состоит из введения, четырех глав, заключения, и ХХ приложений. Объем основной части диссертации составляет ХХХ страниц, полный объем диссертационной работы – ХХХ страниц, и включает в том числе ХХ таблиц и ХХ рисунков. Список литературы содержит ХХ наименований.

# СОДЕРЖАНИЕ РАБОТЫ

**Во введении** обосновывается актуальность темы исследования, определяется цель и решаемые задачи, объекты и предмет исследования, формулируются положения, выносимые на защиту, их теоретическая и практическая значимость и научная новизна.

**В первой главе** приводится описание системы хранения данных (СХД) как объекта диагностики с рассмотрением её укрупненной архитектуры, представленной на рисунке 1, в том числе с учётом особенностей реализации системы хранения данных, анализировавшийся в рамках настоящего исследования. Рассмотрены основные компоненты СХД, их взаимосвязь, причины возникновения неисправностей и особенности их распространения по топологии системы.

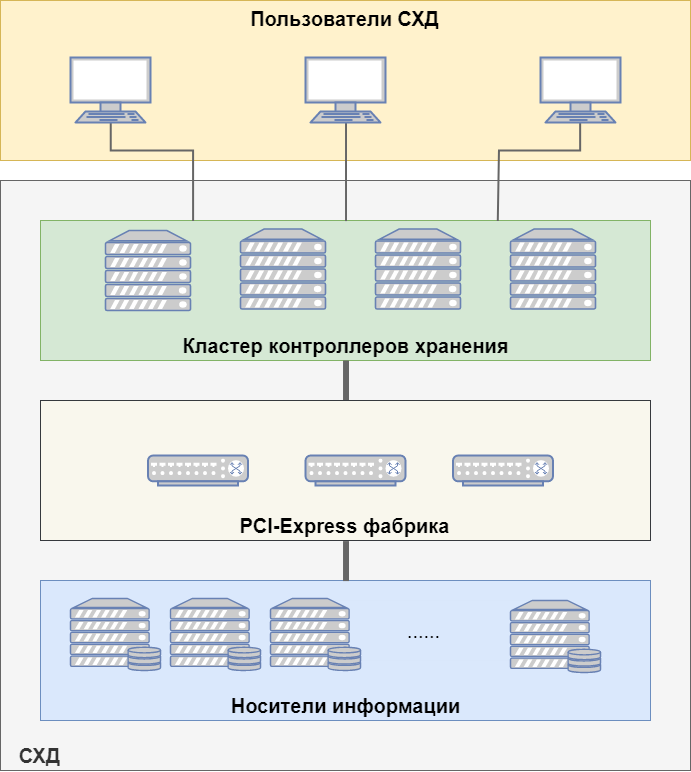


Рисунок 1 – Архитектура объекта диагностики

Выполнен системный анализ наиболее актуальных на текущий момент программных и программно-аппаратных средств, предназначенных для обнаружения и предотвращения возникновения неисправностей в вычислительных системах, которые применяются или могут быть применены для диагностики СХД. Предложена их классификация в соответствии с областью применения, принципом сбора диагностических данных и типами данных, получаемых в результате процесса мониторинга и используемых в процессе диагностики.

Определен и обоснован набор критериев, характеризующих основные преимущества и недостатки данных инструментов и проведено их сравнение в соответствии с данными критериями, с определением перечня подходов, применяющихся для обнаружения неисправностей.

На основании выявленного перечня подходов к обнаружению неисправностей в рассмотренных средствах, проведен анализ современных научных публикаций, направленный на поиск наиболее актуальных методов реализации данных подходов, в том числе перспективных алгоритмов обнаружения аномалий в диагностических данных, алгоритмов классификации и кластеризации.

В результате анализа определена потребность в разработке новых инструментов обнаружения сбоев, направленных на более широкое использование журналов программного обеспечения СХД (ПО СХД), в том числе с использованием диагностической модели СХД.

**Во второй главе** предложен подход к разработке диагностической модели СХД на основании онтологии, позволяющей сопоставить вектор наблюдаемых значений входных параметров ***{Pi}*** и оценку текущего состояния СХД, определяемую как:

* набор состояний (из списка работоспособное, предотказное, неисправность, полный отказ) компонентов СХД, подсистем СХД и СХД в целом ***{Sc, Ss, S}****;*
* вектор обнаруженных событий в СХД - ***{F}****.*

Разработанная в соответствии с указанным подходом онтологическая модель включает в себя следующие основные разделы:

* раздел описания топологии СХД;
* раздел описания параметров СХД;
* раздел описания неисправностей;
* вспомогательные разделы, например, раздел, включающий, классы и свойства для описания процесса обработки журналов СПО СХД.

Параметр СХД описывается следующим образом:

*Pi =* (1)

*где N – число связей i-ого параметра СХД, <R,V> - пара связь/значение.*

Компонент СХД:

*Ci = <* (2)

*где N – число параметров, а M – число возможных событий i-ого компонента СХД, <F, D, R> - триплет событие/состояние/связь с параметрами, P – набор параметров компонента.*

Подсистема СХД:

*Si = <*(3)

*где N – число компонентов, M – вложенных подсистем, а L – событий- i-ой подсистемы СХД, <F, D, {R, R}}> - триплет <событие/состояние/связь, определяющая компонент или подсистему – источник события>.*

Тогда система в целом:

*Storage = <* (4)

*где N – число параметров, M – число возможных событий, <F, D, R> - триплет событие/состояние/ связь, определяющая подсистему – источник события.*

Под **компонентом СХД *{C}*** понимается объект, считающийся на уровне модели атомарным и неделимым, его состояние описывается при помощи соответствующих ему параметров СХД. Для систематизации знаний в онтологической модели принята иерархическая структура классов-компонентов, в соответствии с типом устройств. Так, класс EthernetController (контроллер ethernet) имеет дерево родительских классов: Component -> Network Device -> Network Interface Card -> EthernetController. При этом, в процессе создания именованных сущностей создаётся экземпляр только класса EthernetController. Компонент СХД может быть, как аппаратным, так и программным, например, одним из сервисов СПО СХД.

**Подсистема СХД *{S}*** представляет собой набор из компонентов СХД, объединенных по какому-либо функциональному признаку. При этом, подсистема может состоять из подсистем, включая подсистемы такого же класса (например, пул хранения StoragePool может включать другие пулы хранения), то есть цепочка предков конкретного класса-подсистемы может включать как классы, для которых создаются именованные сущности, так и абстрактные классы. Для отношения между подсистемами и компонентами применяется набор объектных свойств «depends\_on» разного веса.

Исходя из этого, каждый элемент иерархии *i-*ого уровня вложенности *Ei* может включать в себя набор из *{E(i+1, j); E(i,k)}*, для *j = 0..N, k = 1..M* (*N* – число дочерних компонентов *i+1* уровня вложенности компонента, *M –* число дочерних компонентов i-ого уровня вложенности). Онтологическое описание такой структуры помимо свойства объектов «depends\_on» задаётся при помощи свойства объектов «consists\_of», задающего отношение «родитель»-«потомок» между классами.

**Параметры** СХД ***{P}*** также, как и компоненты, атомарны и имеют набор абстрактных классов-родителей, позиционирующий их в структуре онтологии, но, в отличие от компонентов, не имеют соответствующих им именованных сущностей. Наблюдаемое значение параметра Pi классифицируется как одно из заданных событий при помощи пары <связь, значение> ***{Ldi, Vi}****,* для *i=1,N,* где N – число определенных в модели вариантов значений параметра. Данная пара фактически определяет правило, по которому наблюдаемое значение параметра должно оцениваться с точки зрения его соответствия штатному или нештатному состоянию. Например, состояние параметра R-W-V-ErrCnt определяется единственной парой*{«is\_normal\_below», 2}.*

**События** вмодели СХД ***{F}*** представляют собой совокупность значений параметров СХД, состояний компонентов или подсистем СХД, характеризующие наступление какой-либо неисправности. Каждое событие при этом классифицируется как одно из состояний ***{D}***, в зависимости от уровня его критичности, который в свою очередь, задаётся при помощи типа свойства объектов. При этом работоспособное состояние компонента СХД подразумевает отсутствие диагностированных неисправностей этого компонента, подсистемы – отсутствие диагностированных неисправностей в дочерних компонентах и подсистемах.

Для определения состояния D*Ei* в элементе *Ei* определяются его дочерние элементы и проверяется наличие событий в них, начиная с элементов самого низкого уровня вложенности – параметров и далее вверх по иерархии элементов от компонентов к подсистемам.

Состояние *Dci* компонента *Сi* определяется при помощи вектора пар <связь, параметр> ***{Loj, Pj}****,* для *j=1,…,N,* где N – число параметров, определяющих события в *i*-м компоненте. Связь определяет, каким именно образом событие зависит от параметра, например, объектное свойство «shows\_in» просто обозначает, что состояние S1 определяется как P1 ˄ PN.

Для обеспечения возможности применения онтологической модели в том случае, когда параметрами являются журналы СПО СХД, предложена реализация условных связей, основанных на применении алгоритмов машинного обучения, применяемых для анализа естественных языков. Применение такого решения позволяет определить связь между значением параметра и событиями в системе даже в том случае, когда нет возможности установить такую связь путём анализа сообщений журналов.

Решение этой задачи обеспечивается добавлением дополнительного типа связи, определяемого как «условная связь», и реализуемого с использованием двух классов объектных связей «solves\_with» и «described\_by», и класса-посредника «ExternalProc», который представляет собой ссылку на внешнюю процедуру, обеспечивающую определение соответствия между вектором входных параметров *{****P****}* и одним из вектора *{****S****}* вариантов событий в компоненте СХД, см. рисунок 2:

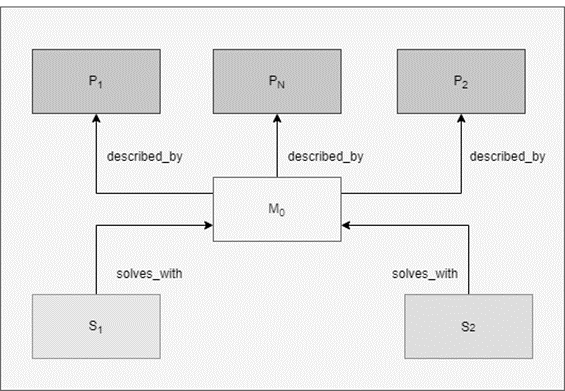


Рисунок 2 – Связь между параметрами и состояниями СХД

Таким образом, с учетом того, что определение событий в подсистемах и СХД в целом выполняется аналогичным образом, функцию построения событий в СХД можно сформулировать как ***S = <La, <Lb, <Lc...<Lz, {Pi}>>>***. При этом, на уровне связей определяется, какими именно состояниям соответствуют определенные события.

Далее во второй главе выполнена оценка журналов СПО СХД как возможного источника диагностической информации для онтологической модели, в том числе предъявлены требования к допустимым типам журналов и форматам сообщений.

Для практического применения онтологической модели в составе программного обеспечения предложено её приведение к упрощенному графовому виду, предполагающему преобразование традиционного rdf к формату rdf-nquad и описание модели в формате [узел]-<связь> [узел] [контекст]. Такое преобразование позволяет устранить громоздкие конструкции наследования, применяемые в формате построения онтологии OWL и свести их к упрощенному виду.

**В третьей главе** рассматривается алгоритм по динамическому обнаружению неисправностей в СХД, предназначенный для реализации онтологической условной связи, рассматривающий в качестве входных параметров блоки сообщений из журналов СПО СХД. Диагностическая процедура включает в себя повторяющееся во времени последовательное выполнение следующих основных этапов:

1. Запуск процесса обнаружения неисправностей
2. Обнаружение изменившихся данных и перестроение входных последовательностей.
3. Классификация текущего состояния.
4. Локализация неисправностей.

Правила разбора заголовка сообщения определяются онтологической моделью, при этом сама процедура разбора базируется на методе комбинаторного парсинга, причем используется единственный вариант комбинатора – bind:

*bind :: Parser a -> (a -> Parser b) -> Parser b*

*p ‘bind‘ f = \inp -> concat [f v inp’ | (v,inp’) <- p inp]*

В таком случае разбор строки *S* набором парсеров *Pi*, производится следующим образом: на вход *Pi* примитивного парсера подаётся строка, остаток строки подаётся на вход *Pi+1* примитивного парсера, и т.д., после выполнения *PN*операция будет считаться завершенной. В онтологической модели предусмотрена двухуровневая система классов, где «MessageHeaderFormatField» определяет свойства поля как части заголовка конкретного журнала, а «MessageHeaderFieldDatatype» определяет свойства объекта, который может быть связан с полями различных журналов, в котором описываются свойства, необходимые для разбора таких полей. Для разбора конкретного типа журнала строится комбинаторный парсер, обладающий информацией о перечне необходимых примитивных парсеров (связи «parser\_name» и «lowlevel\_type»), наименованиях соответствующих им полей (именованные сущности класса «MessageHeaderFormatField») и порядке расположения примитивных парсеров (отношение «parsing\_order»).

Предварительная обработка текстовой части сообщения включает в себя следующие действия:

1. Очистка и разбиение текста.
2. Преобразование текста в стандартную форму.
3. Фильтрация текста.

В рамках данной процедуры выполняется токенизация текста, приведение его к единому регистру, удаление всех небуквенных токенов, удаление стоп-слов, стемминг и фильтрация сообщений, где признаки фильтрации задаются для каждого журнала на уровне онтологической модели.

Процедура обнаружения неисправностей в обработанном тексте представляет собой применение алгоритма классификации к вектору Xu = {Vt, Xm}, представляющему собой объединение векторного представления текста журнала Vt и вектора дополнительных признаков Xm. Для повышения эффективности построения векторного представления текста совместно применялись два типа коэффициентов:

* весовой коэффициент Wtf-idf, характеризующий важность слова в рамках контекста и представляющий собой отношение частоты появления слова в документе к частоте употребления слова во всех документах из выборки;
* весовой коэффициент Werr, характеризующий важность слова для идентификации неисправности.

Коэффициент Werr вводится для того, чтобы компенсировать дискретность размера временного окна, используемого для построения векторного представления сообщений, так как сообщения, зарегистрированные в начале временного окна, имеют меньшую важность для идентификации возникающих неисправностей чем сообщения, зафиксированные ближе к концу временного окна.

Таким образом, итоговый вектор Vw, описывающий каждое слово, имеет следующий вид (4):

И, соответственно, вектор Vt, описывающий текущее временное окно, должен быть определен как (5):

где n – общее число слов в выборке.

Дополнительные параметры Xm описывают количественные характеристики текста журнала, их перечень приведен в таблице 1.

Таблица 1 – Описание дополнительных параметров текста

|  |  |  |
| --- | --- | --- |
| **Название** | **Значение признака** | **Описание параметра** |
| Количество слов в i-ом журнале | Целочисленное | Подсчитывается общее количество слов во временном окне, соответствующем каждому журналу |
| Количество сообщений в i-ом журнале | Целочисленное | Подсчитывается общее количество сообщений во временном окне, соответствующем каждому журналу |
| Средняя длина сообщения в i-ом журнале | С плавающей точкой | Подсчитывается средняя длина по каждому сообщению во временном окне, соответствующем каждому журналу |
| Наличие сообщений в i-ом журнале | Целочисленное | Фиксируется наличие сообщений по каждому журналу в процессе построения временного окна, наличию соответствует 0, отсутствию – 1. |
| Среднее количество слов в секунду в i-ом журнале | С плавающей точкой | Фиксируется количество слов в секунду для каждого журнала |
| Среднее количество сообщений в секунду в i-ом журнале | С плавающей точкой | Фиксируется среднее количество сообщений за одну секунду временного окна |

Оценка важности данных параметров выполнялась двумя способами, на основании применения алгоритма Random Forest и расчёта на каждом шаге критерия Джини:

(1)

и альтернативным способом, путём расчёта F-критерия на множестве признаков, представляющего собой отношение межгрупповой дисперсии к внутригрупповой:

F (4)

Проверка выполнялась на имеющихся размеченных данных, полученных по результатам тестирования ПО СХД. Общая выборка содержит 5904 пакетов журналов (что составляет порядка 350Гб файлов журналов), размещаемых на контроллерах хранения СХД, в том числе 1574 соответствующих возникавшим когда-либо в процессе эксплуатации СХД неисправностям, с указанием примерного временного диапазона возникновения неисправностей. Всего в пакетах был идентифицирован 41 тип различных неисправностей.

Сравнительная частота отдельных неисправностей от общего количества пакетов с неисправностями представлена на рисунке 3. В качестве меры сбалансированности ***B*** использовалось отношение энтропии по Шэннону(2) к двоичному логарифму числа классов:

; (2)

где *k* – число различных классов, *сi* – размер класса, *n* – размер выборки.

Данная мера принимает значение 0, когда набор полностью не сбалансирован (0, когда есть всего один большой класс) и 1, когда набор состоит из классов одинакового размера. Значение меры Шэнннона для данного набора классов составляет 0,974, что означает, что набор данных достаточно хорошо сбалансирован.

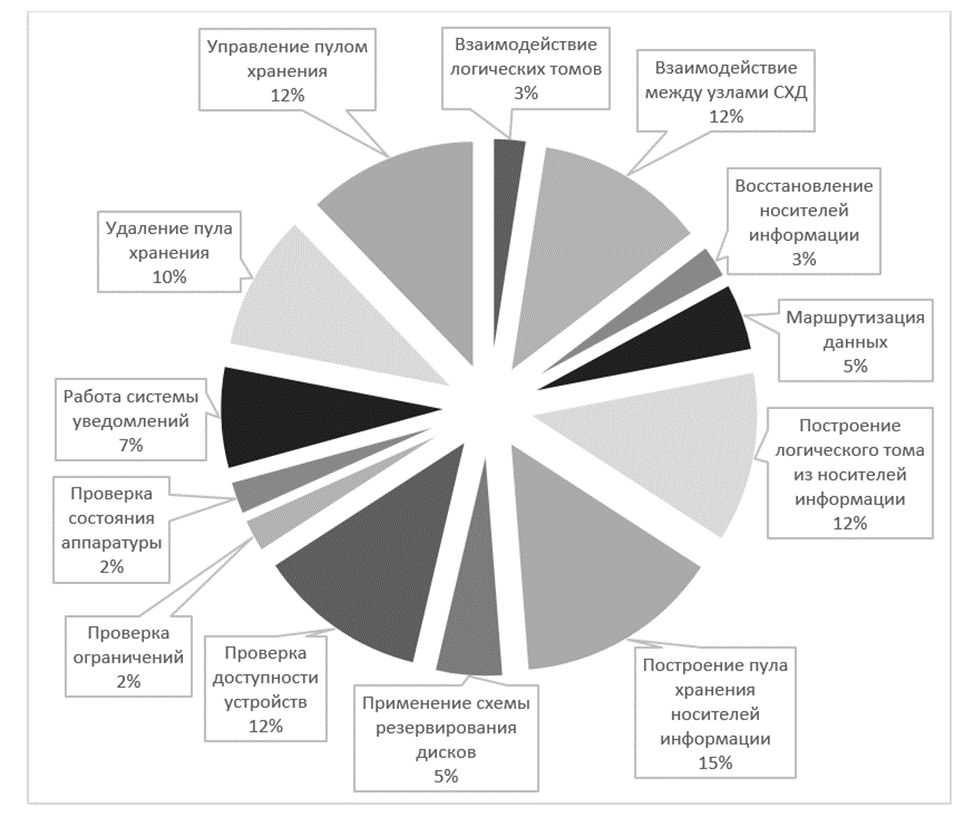
****

Рисунок 3 – сравнительная частота возникновения сбоев

Выполненная оценка важности признаков позволила исключить признак «наличие сообщений в i-ом журнале», прочие показали свою важность для классификации.

**В четвертой главе** приводится описание формирования и применения онтологической модели в составе СПО СХД, схема применения указана на рисунке 4.

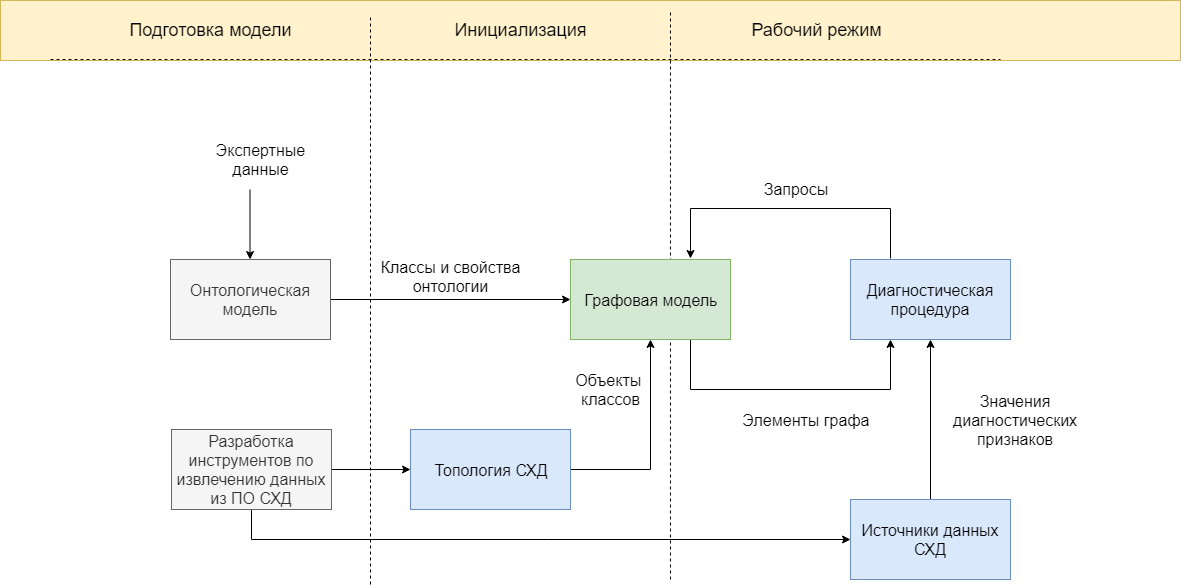


Рисунок 4 – Схема применения моделей

На первом этапе выполняется подготовка онтологической модели, путём наполнения её экспертными данными в редакторе онтологий (Stanford Protégé).

Далее, на этапе инициализации ПО СХД, модель преобразуется к графовому виду, причём структура модели, то есть её классы и свойства (classes, properties), получается путём преобразования онтологической модели, а объекты классов (named individuals) – путём обращения к системному программному обеспечению СХД. Сформированная таким образом модель системы помещается в графовую базу данных и используется в рамках диагностической процедуры для получения диагностических сведений о взаимосвязи значений диагностических признаков со сбоями в СХД.

Архитектура реализующего диагностическую процедуру ПО предполагает получение текущего значения параметров СХД, в том числе текстов временных окон из журналов СПО СХД из базы данных, заполняемой модулем мониторинга и сбора данных. Экземпляр такого диагностического ПО запускается менеджером кластера на одном из контроллеров хранения, входящих в кластер, и определяет состояние системы в целом и её отдельных подсистем, и компонентов.

**В пятой главе** выполняется экспериментальная проверка работы диагностического ПО на массиве текстов журналов со встроенной моделью и обученными алгоритмами классификации текста. Целью проведения эксперимента являлись:

1. выбор оптимальных элементов процедуры классификации, в том числе собственно алгоритма-классификатора (рассмотрены классические алгоритмы Random Forest, наивный байесовский классификатор, KNN, логичтическая регрессия, метод опорных векторов и нейронные сети глубокого обучения LSTM, RNN и LSTM с механизмом внимания), способа векторного представления текста и способа объединения текстов в журналах;

2. оценка эффективности всего подхода к обнаружению неисправностей в СХД.

Результаты эксперимента 1 показали, что наиболее эффективным является раздельное построение векторного пространства для каждого журнала и объединение их вместе с векторами дополнительных признаков, определенных для каждого журнала, в единое признаковое пространство. Результаты сравнения качества классификации представлены в таблице 2, наибольшую эффективность показал алгоритм Random Forest.

Таблица 2 – Сравнительная оценка эффективности классификации

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Классификатор | Средняя точность | Средняя полнота | Средняя f-мера | Среднее время обучения модели |
| Random Forest | 0,74 | 0.71 | 0.71 | 183 с |
| Наивный Байес | 0,48 | 0,27 | 0,28 | 205 c |
| KNN | 0,38 | 0,39 | 0,37 | 166 c |
| Логистическая регрессия | 0,28 | 0,33 | 0,28 | 498 c |
| Метод опорных векторов | 0,23 | 0,26 | 0,20 | 430 c |
| LSTM | 0,47 | 0,44 | 0,45 | ~4 часа |
| GRU | 0,24 | 0,22 | 0,23 | ~7 часов |
| LSTM с механизмом внимания | 0, 42 | 0,39 | 0,40 | ~ 6 часов |

При этом, по графикам обучения нейронных сетей, см. рисунок 5, видно, что нейронные сети LSTM демонстрируют признаки недообученности, а GRU с задачей классификации не справились.

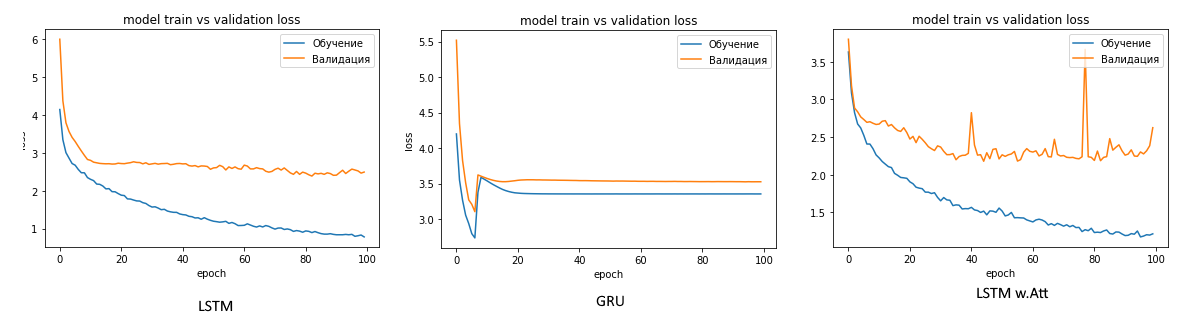


Рисунок 5 – График обучения нейронных сетей

Оценка эффективности всего подхода выполнялась с использованием имитации неисправностей, так как частота их возникновения в процессе штатного функционирования СХД слишком мала, из-за чего нет возможности наблюдать достаточное количество неисправностей за время проведения эксперимента. По результатам эксперимента получены следующие результаты:

* для неисправностей, определяемых в результате применения правил обнаружения, заданных детерминированными онтологическими связями, вероятность идентификации сбоя составляет 0.99 на массиве данных в 10000 неисправностей;
* для неисправностей, определяемых в результате применения правил, заданных условными онтологическими связями, вероятность идентификации сбоя составляет 0.71 на массиве данных в 10000 неисправностей.

**Заключение**

В диссертационной работе разработано решение актуальной научно-технической задачи по повышению эффективности обнаружения неисправностей в СХД за счёт совместного использования онтологической модели, её упрощенного графового представления и алгоритма обнаружения неисправностей на основании классификации текстов журналов СПО СХД.

Получены следующие результаты:

1. Выполнен анализ существующих решений и научных исследований в области диагностики вычислительных систем, определены их сравнительные характеристики, сформулирована задача разработки нового метода обнаружения неисправностей.
2. Разработана онтологическая модель СХД, определяющая отношение между параметрами и состояниями СХД путем использования объектных связей, связей данных и связей, задаваемых при помощи внешних объектов, основанных на использовании предварительно обученных алгоритмов машинного обучения для установления соответствия между параметрами и состояниями СХД.
3. Предложено и обосновано использование графового вида онтологической модели для её применения в составе программного обеспечения и формализован подход к построению такого графа.
4. Разработка алгоритм обнаружения неисправностей в СХД на основании классификации текстов журналов СПО СХД с использованием алгоритмов машинного обучения.
5. Экспериментально обосновано использование параметрического описания СХД на основании текстов журналов СПО СХД и обоснован выбор алгоритмов классификации.
6. Проведена экспериментальная проверка эффективности применения разработанных методов и средств.

В качестве **рекомендаций и перспектив дальнейшей разработки темы** можно указать исследование моделей векторного представления текстов журналов СПО СХД, основанных на применении нейронных сетей и определение возможности совместного использования разных алгоритмов для решения задачи классификации.

Полученные результаты соответствуют пунктам 2 «Формализация и постановка задач системного анализа, оптимизации, управления, принятия решений и обработки информации.», 4 «Разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации.» и 11 «Методы и алгоритмы прогнозирования и оценки эффективности, качества и надежности сложных систем» паспорта специальности 05.13.01 Системный анализ, управление и обработка информации (по отраслям).

**Список публикаций по теме диссертации**

**В журналах, рекомендованных ВАК:**

1. Успенский М.Б. Обзор подходов к обнаружению сбоев в системах хранения данных / Успенский М.Б., Смирнов С.В. // Научно-технические ведомости СПбГПУ. Информатика.Телекоммуникации.Управление – 2019. - №4 (в печати).

2. Успенский М.Б. Применение онтологической модели и алгоритмов классификации текста в задачах обнаружения сбоев систем хранения данных / Успенский М.Б. // Проблемы информационной безопасности. Компьютерные системы – 2019. – №4 (в печати)

3. Успенский М.Б. Применение методов классификации текста для анализа журналов программного обеспечения // Проблемы информационной безопасности. Компьютерные системы – 2019. – №4 (в печати)

**В зарубежных изданиях, индексированных в Web of Science и Scopus**

4. Shirokova S.V., Bolsunovskaiya M.V., Loginova A.V., Uspenskiy M.B. Developing a procedure for conducting a security audit of a software package for predicting storage system failures // MATEC Web Conf, 2018. International Scientific Conference on Energy, Environmental and Construction Engineering (EECE-2018). Volume 245, № 10007.

5. Uspenskij M., Makarov A., Sochnev A., Shirokova S., Petrov V. Development of a software structure for monitoring the working capacity of the data storage system for predicting failures and preventing critical situations // Proceedings of the 33rd International Business Information Management Association Conference, IBIMA 2019: Education Excellence and Innovation Management through Vision 2020, 2019. Pp. 8508-8514.

6. Mamoutova O.V., Shirokova S.V., Uspenskij M.B., Loginova A.V. The ontology-based approach to data storage systems technical diagnostics // E3S Web of Conferences, Vol. 91, № 08018.

7. Mamoutova O.V, Uspenskiy M.B., Sochnev A.V., Smirnov S.V. Bolsunovskaya M.V. Knowledge Based Diagnostic Approach for Enterprise Storage Systems // Proceeding of IEEE 17th International Symposium on Intelligent Systems and Informatics, 2019

**В прочих изданиях**

8. Макаров А.С., Болсуновская М.В., Широкова С.В., Успенский М.Б., Кузьмичев А.А. Анализ  подходов к диагностике систем хранения данных // Мягкие вычисления и измерения: сборник трудов XXI Международной конференции, 2018. Т. 2. С. 61-64

9. Приданова Е.В., Успенский М.Б., Ицыксон В.М. Мониторинг и анализ параметров системы хранения данных для оценки ее состояния // Сборник научных трудов XXIII Международной научно-практической конференции “Системный анализ в проектировании и управлении”, 2019. С. 170-177.

10. Иванов О.И., Михайлов Е.А., Пустоветов В.И., Успенский М.Б. Подсистема подготовки тестовых проектов для контрольно-диагностических комплексов // Вопросы радиоэлектроники. 2013. Т. 1. № 1. С. 99-105.

11. Берлик С.А., Иванов О.И., Успенский М.Б., Пустоветов В.И. Архитектура графической среды аппаратно-программного комплекса КДК // Вопросы радиоэлектроники. 2013. Т. 1. № 1. С. 73-80.

12. Михайлов А.Н., Иванов О.И., Успенский М.Б. Сигнатурный анализатор для многофункционального аппаратно-программного комплекса КДК // Вопросы радиоэлектроники. 2014. Т. 1. № 2. С. 106-112.

**Свидетельства о государственной регистрации программ для ЭВМ**

1. Успенский М.Б. Программа для сбора параметров системы хранения данных / М.Б. Успенский, В.Д. Петров, А.В. Сочнев, В.И Пустоветов. – Свидетельство о государственной регистрации программы для ЭВМ № 2018660284 от 21.08.2018.

2. Успенский М.Б. Программа имитации функционирования аппаратной компоненты системы хранения данных - носителя информации / М.Б. Успенский, М.Е. Карпов. – Свидетельство о государственной регистрации программы для ЭВМ № 2018665078 от 30.11.2018.

3. Успенский М.Б. Программа имитации функционирования аппаратной компоненты системы хранения данных - контроллера фабрики PCI-Express / М.Б. Успенский, К. Арзыматов. – Свидетельство о государственной регистрации программы для ЭВМ № 2018665160 от 03.12.2018.

4. Успенский М.Б. Программа имитации функционирования аппаратной компоненты системы хранения данных - контроллера хранения данных / М.Б. Успенский, В.С. Белавин. – Свидетельство о государственной регистрации программы для ЭВМ № 2018665676 от 06.12.2018.

5. Успенский М.Б. Программа для сбора и отображения климатических параметров систем хранения данных / М.Б. Успенский, С.В. Смирнов. – Свидетельство о государственной регистрации программы для ЭВМ № 2019614476 от 05.04.2019.

6. Успенский М.Б. Программа для диагностирования системы хранения данных / М.Б. Успенский, М.И. Гущин. – Свидетельство о государственной регистрации программы для ЭВМ № 2019618328 от 27.06.2019.

7. Успенский М.Б. Программа настройки параметров имитационной модели функционирования системы хранения данных / М.Б. Успенский, В.С. Белавин. – Свидетельство о государственной регистрации программы для ЭВМ № 2019618010 от 25.06.2019.

8. Иванов О.И. Программа тестирования и диагностики цифровой радиоэлектронной аппаратуры с использованием многоканального сигнатурного анализа / О.И. Иванов, В.И. Пустоветов, М.Б. Успенский. – Свидетельство о государственной регистрации программы для ЭВМ № 2015661325 от 23.10.2015.

9. Иванов О.И. Программа подготовки комбинированных тестовых проектов / О.И. Иванов, В.И. Пустоветов, М.Б. Успенский. – Свидетельство о государственной регистрации программы для ЭВМ № 2015661326 от 23.10.2015.