ДОКЛАД

ПО ДИССЕРТАЦИИ ТОЩЕВА А. С. «ИНТЕЛЛЕКТУАЛЬНАЯ СИСТЕМА ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ИТ-СЛУЖБЫ ПРЕДПРИЯТИЯ» ПО СПЕЦИАЛЬНОСТИ 05.13.11 — МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ МАШИН, КОМПЛЕКСОВ И КОМПЬЮТЕРНЫХ НА СОИСКАНИЕ УЧЕНОЙ СТЕПЕНИ КАНДИДАТА ТЕХНИЧЕСКИХ НАУК

**Слайды в Приложении 1.**

**Слайд 1**

Уважаемые члены диссертационного совета, оппоненты, присутствующие;

Я представляю Вам свою диссертацию, на слайде указаны ее данные.

**Слайд 2**

На этом слайде приведены обозначения, которые используются в докладе и диссертации.

Например, инцидент – это проблема, которая возникла в результате работы программного обеспечения и приведшая к полной или частичной невозможности работы.

На Слайде приведены обозначение инцидента

**Слайд 3-4**

Вот содержание диссертации. Она состоит из 4 глаз, введения и заключения.

**Слайд 5**

Начну с введения. Здесь представлены цели, задачи и предмет диссертации.

Актуальность подтверждается тем, что подобные системы сейчас создаются по всему миру в крупнейших компаниях

**Слайд 6**

Часть из них, включая IBM указаны на данной слайде. Здесь также указаны лидеры групп и их научные исследования

**Слайд 7**

Среди подобных систем, ни одна из подобных систем не способна использовать логическое рассуждение. Большинство из них используют исключительно для регистрации проблем, но никак не для их решения. Кроме того, все эти системы являются закрытыми и не имеют открытого исходного кода, чтобы результаты этих исследований могли использоваться другими. В отличии от моей диссертации, которая сейчас выложена в открытом доступе.

**Слайд 8**

 Во время своей работы я использовал следующие теоретические, специальные и экспериментальные методы.

**Слайд 9**

Работа находится в строгом соответствии с паспортом специальности. Здесь слева приведены данные с паспорта специальности, а справа результаты работы.

На следующем слайде идет продолжение.

**Слайд 10-11**

Здесь приведены дополнительные направления исследования в соответствии с паспортом специальности.

Основные результаты опубликованы в 10 печатных изданиях. На следующих слайдах я их привел. Если это будет интересно, мы позже сможем к ним вернуться.

**Слайд 12 - 16**

Результаты также опубликованы в 2 изданиях, входящих в перечень ВАК.

**Слайд 17-19**

Результаты также выкладывались на ряде международных и российских конференциях, конференции собраны по годам

**Слайд 20-21**

Перехожу к описанию содержания диссертации

**Слайд 22**

Задача состояла в том, чтобы оптимизировать эффективность службы поддержки предприятия. Используются результаты работы компании ICL, где существуют проекты, которые занимаются удаленной поддержкой пользователей, если у них случилась какая-то проблема, инцидент, так называемый, то они его должны решить.

**Слайд 23**

Был проанализирован период в месяц, в него входило 2920 инцидентов, и они были категоризированы. Самая большая по численности категория – проблемы с программным обеспечением. Это конечно же достаточно абстрактная категория. Ну а на втором месте по частоте идет категория – установить новое программное обеспечение.  Что, в принципе, казалось бы, может быть автоматизировано. Потому что ничего сложного, на первый взгляд, в этом нет. На третьем месте идет категория – нет доступа. Которая также может быть автоматизирована.

**Слайд 24**

И по результатам анализа выяснилось, что очень много существует повторяющихся инцидентов, которые могут быть автоматизированы.

Но я столкнулся с проблемами, что, во-первых, бывают достаточно неоднозначные запросы, где приведено очень много лишней информации, которая не имеет смысла, и, конечно же, запросы с грамматическими ошибками.

**Слайд 25**

Сама модель этих проектов системы представляет собой модель из теории массы обслуживания, где также существует поток заявок, очередь, агенты, которые это обрабатывают и имеется время нахождения заявки в очереди.

**Слайд 26-27**

На следующих слайдах приведены обозначения из теории массового обслуживания.

Для исходных данных, которые я анализировал, я провел замер и получил следующие цифры:

Среднее время прохождения – 47 часов , при количестве агентов -6

И важный показатель – количество инцидентов, которые были выполнены в срок, здесь составляет 82%.

**Слайд 28-29**

Для решения этой проблемы была создана интеллектуальная система. Работа над ней велась с 2010 года. Рассматривались различные модели. На этом слайде приведена их эволюция. И в конце я остановился на модели мышления, которая основывается на концепции, предложенной Марвином Мински. Здесь приведены модели и ссылки на публикации, где подробно они описываются.

**Слайд 30**

Основой модели является Т3 – это некий вероятностный критикан, который благодаря логическим правилам активируется, с помощью селектора получает путь мышления. Это то, как будет решен тот или иной инцидент.

**Слайд 31-32**

Сама система в ее реализации представляет собой совокупность слабо связанных вероятностных машин, которые управляются основной машиной и взаимодействуют между собой при помощи паттерна. И процесс называется слабосвязанным, потому что результатом работы критика является вероятность, которая подсчитывается после зачисления результата работы правил, которые находятся внутри него. Тем самым может активироваться несколько критиков. Некоторые будут более вероятные, некоторые менее вероятные. Это дает возможность придать системе гибкость, придать ей возможность применять менее вероятные решения, но в данный момент оно поможет решить проблему.

**Слайд 33-34**

Общая диаграмма действий верхнего уровня состоит сначала из классической обработки запроса, классификации его (понять, что от нас хотят, либо предоставить доступ, либо установить программное обеспечение), если же это проблема с желаемом состоянии, то происходит его переформулировка, для того, чтобы потом искать решение непосредственно уже как по запросу на прямые инструкции. Далее идет поиск решений и применение решений.

**Слайд 35-39**

Вот, например, обработка запроса «Please install Firefox», выставляется цель – обработать, идет обработка и на выходе мы получаем текст, который преобразовывается в сематические сети. Здесь основные концепции, здесь ветви, которые описывают эту концепцию.

Дальше система подвергает анализу, и после обработки она использует только эти 5 связей. Тем самым после обработки граф представляет собой такой вид.

Построенный граф ищет соответствия во внутренней базе знаний и прикрепляется к уже известным концепциям. Тем самым на выходе мы получаем, что в базе уже есть эта концепция и она используется.

Во время поиска решения идет сравнение двух графов.

**Слайд 40**

Пример обработки сложного запроса:

Видно, что система отбрасывает все лишние концепции, потому что не использует несущественные для него связи.

**Слайд 41**

Дальше пример обработки запроса с желаемым состоянием. Есть офис 2010, а нужен 2016. Тем самым граф приобретает два состояния и ищется решение, которое может к этому желаемому состоянию привести.

**Слайд 42**

Здесь приведена полная схема работы системы. И вот небольшой фрагмент. Все описано в формате UML, чтобы легко это было читать.

**Слайд 43**

В системе, безусловно есть ограничения: длина запроса, наличие только 5 семантических связей, логически противоречивые запросы не обрабатываются, поддерживается пока только английский язык, не более 8 логических правил на человека.

**Слайд 44**

Одними из особенностей являются расширяемость, масштабируемость.

**Слайд 45-46**

После экспериментальных исследований выяснилось, что на короткие запросы система обрабатывает быстрее, на длинных запросах ей нужно чуть больше времени, чем специалисту.

Здесь данные ровные, потому что, к сожалению, с той системой, с которой я брал данные, они округлялись до секунд.

**Слайд 47-49**

После обработки исходных инцидентов, выяснилось, что 64% проблем с ПО требуют вмешательства специалиста для их разрешения.

Удалось улучшить показатели, путем добавления в цикл системы. То есть наряду со специалистами работала система и результативность выросла до 96%. Но нужно учесть, что это работа уже не специалиста, а логической системы.

**Слайд 50-51**

На следующем слайде представлены основные результаты работы: на основе модели, разработанной в диссертации, создана архитектура системы и ее прототип. И все эти результаты находятся в строгом соответствии со специальностью.

**Слайд 52-53**

Программный комплекс зарегистрирован. Сейчас идет оформление патента на эту разработку. Также имеется акт о внедрении результата в компании.

**Слайд 54**

Спасибо за внимание.

На слайдах 55-68 приведены вопросы оппонентов и дополнительная информация.

Дата защиты

25 мая 2017 года Тощев А.С.