

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления
Кафедра Интеллектуальных информационных технологий

ОТЧЁТ
по ознакомительной практике

Выполнил:

Е. В. Грибанов

Студент группы
321703

Проверил:

В. Н. Тищенко

Минск 2024

СОДЕРЖАНИЕ

Введение	3
1 Постановка задачи	4
2 Формализованные фрагменты теории реализации интерпретатора sc-моделей пользовательских интерфейсов	5
3 Формальная семантическая спецификация библиографических ис- точников	9
Заключение	10
Список использованных источников	11

ВВЕДЕНИЕ

Цель:

Закрепить практические навыки формализации информации в интеллектуальных системах с использованием семантических сетей.

Задачи:

- Построение формализованных фрагментов теории интеллектуальных компьютерных систем и технологий их разработки.
- Построение формальной семантической спецификации библиографических источников, соответствующих указанным выше фрагментам.
- Оформление конкретных предложений по развитию текущей версии Стандарта интеллектуальных компьютерных систем и технологий их разработки.

1 ПОСТАНОВКА ЗАДАЧИ

Часть 2 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"

⇒ библиографическая ссылка*:

- Стандарт OSTIS
- Толковый словарь по Искусственному интеллекту

⇒ URL*:

[<http://raai.org/library/tolk/aivoc.html>]

- ...

⇒ аттестационные вопросы*:

- < • Вопрос 1 по Части 2 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"
- >

Вопрос 1 по Части 2 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"

:= [Реализация интерпретатора sc-моделей пользовательских интерфейсов]

⇒ библиографическая ссылка*:

- В.В. Голенков, Н.А. Гулякина, И.Т. Давыденко, Д.В. Шункевич.
СеманТПИСиСАК-2019ст
- В.В. Голенков, Н.А. Гулякина. ПроектОСТКПИС-2014ст

2 ФОРМАЛИЗОВАННЫЕ ФРАГМЕНТЫ ТЕОРИИ РЕАЛИЗАЦИЯ ИНТЕРПРЕТАТОРА SC-МОДЕЛЕЙ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ

многоагентная система

- :=** [система агентов, которая занимается решением задач]
- ⇒** *пояснение**:
- [агенты первого уровня должны обладать одинаковым уровнем знаний, но иметь разные алгоритмы, чтобы на выходе изменялся результат]
- ⇒** *пояснение**:
- [важно, что каждый из агентов работает на четко определенной для него области знаний и по четко определенным для него алгоритмам принятия решения]
- ⇒** *свойства агентов первого уровня**:
- {** • [агенты могут общаться между собой]
 - [агенты могут обмениваться алгоритмами]
 - [агенты могут содержать подагентов]
 - [агенты имеют главного агента, который выдает им общее
 - [агенты могут обмениваться информацией]задание, а потом собирает результаты]
- ⇒** *пояснение**:
- [важно, что главный агент выбирает на конкурсной основе, то есть выбирает те, которые наиболее соответствуют заданию и его тематике]
- }**
- ⇒** *пояснение**:
- [для хранения и обработки данных интеллектуальных агентов, могут быть использованы миварные технологии]
- ⇒** *определение**:
- [Мивары – это более общий формализм, чем семантические сети, онтологии и модель «сущность-связь»]

*следует отличать**

- ⊃** **{** • [миварные технологии]
- [семантические сети]
- }**
- ⇒** *библиографический источник**:
- Белоусова А.И..ПодходФМММСсИМ-2011ст

интерпретатор знаний и навыков

- :=** [коллектив агентов, осуществляющих обработку базы знаний и управляемых ситуациями и событиями в этой базе знаний.]

платформенная независимость системы

- :=** [способность системы децентрализованно работать на различных платформах]
- ⇒** *пояснение**:
- платформы*
- =** **{** • Windows

- *Linux*
- *Mac*

}

⇒ *пояснение**:
способы достижения

= { • *четкое разделение унифицированной логико-семантической модели такой системы (sc-модели компьютерной системы)*

- *универсальный интерпретатор sc-моделей компьютерных систем.*

⇒ *разбиение**:

- { • *программно обеспеченный интерпретатор*
- *аппаратно обеспеченный интерпретатор*

}

интерпретатор языка SCP

:= [Semantic Code Programming]

:= [базовый процедурный язык программирования, ориентированный на обработку текстов SC-кода, хранимых в смысловой графовой ассоциативной памяти.]

⇒ *пояснение**:
требуется наличие универсального интерпретатора

фон-неймановская архитектура системы

:= [аппаратная реализация описанной ранее программной модели с сохранением традиционных принципов хранения и обработки информации.]

⇒ *пояснение**:
Такой вариант реализации будет обладать существенно более высокой производительностью по сравнению с программной моделью, однако реализация семантических моделей представления и обработки информации на основе традиционной фон-неймановской архитектуры все еще будет обладать значительно более низкой производительностью по сравнению с реализацией интерпретатора в виде семантического ассоциативного компьютера.

⇒ *разбиение**:
причины низкой производительности

= { • *последовательная обработка, ограничивающая эффективность компьютеров физическими возможностями элементной базы*

- *низкий уровень доступа к памяти, т. е. сложность и громоздкость выполнения процедуры ассоциативного поиска нужного фрагмента знаний. Ускорить процесс доступа можно путем создания специализированной ассоциативной памяти, обеспечивающей ассоциативный доступ к произвольным фрагментам хранимых знаний (имеющим произвольные размеры и структуру). Кроме того, актуальной становится задача реализации обеспечения параллелизма при выполнении поисковых операций*
- *представление информации в памяти современных компьютеров практически не апеллирует к семантике представляемой информации, что сильно затрудняет переработку знаний за счет необходимости учета большого количества деталей, касающихся способа представления информации в памяти, а не ее смысла*

}

семантический ассоциативный компьютер

- :=** [САК]
- :=** [компьютер с нелинейной структурно перестраиваемой (графодинамической) ассоциативной памятью, переработка информации в которой сводится не к изменению состояния элементов памяти, а к изменению конфигурации связей между ними]
- ⇒ разбиение*:**
 - принципы САК**
 - =** { • *в качестве внутреннего способа кодирования знаний, хранимых в памяти семантического ассоциативного компьютера, используется универсальный способ нелинейного (графоподобного) смыслового представления знаний (SC-код);*
 - *есть базовые агенты, которые не могут быть реализованы программно*
 - *все агенты работают над общей памятью одновременно*
 - ⇒ примечание*:**

Более того, если для какого либо агента в некоторый момент времени в различных частях памяти возникает сразу несколько условий его применения, разные акты указанного агента в разных частях памяти могут выполняться одновременно (акт агента – это неделимый, целостный процесс деятельности агента)
 - }**

нелинейная память

- :=** [память, в которой каждый элементарный фрагмент хранимого в памяти текста может быть инцидентен неограниченному числу других элементарных фрагментов этого текста]
- ⇒ примечание*:**

в данный момент тяжело создать подходящий тип нелинейной памяти, который подходил бы для специфических задач

sc-машина

- :=** [sc-модель обработки знаний]
- :=** [абстрактная модель, в основе которой лежит SC-код]
- ⇒ пояснение*:**

Каждая такая модель (sc-машина) представляет собой многоагентную систему
- ⇒ разбиение*:**
 - составляющие модели**
 - =** { • *графодинамическая память*
 - :=** [память, в которой хранятся и обрабатываются тексты SC-кода]
 - :=** [sc-память]
 - *коллектив агентов*
 - :=** [sc-агент]
 - ⇒ пояснение*:**

такие агенты должны работать над общей для них sc-памятью
 - }**
- ⇒ разбиение*:**
 - языки программирования**
 - =** { • *Семейство sc-языков программирования высокого и сверхвысокого уровня (как процедурных, так и непроцедурных).*

⇒ *пояснение**:
Тексты программ этих языков хранятся в базе знаний ИС и описывают способы решения различных классов задач в соответствующих предметных областях.

- *Базовый sc-язык программирования (язык SCP), на котором описываются sc-агенты и интерпретации sc-языков программирования высокого и сверхвысокого уровня*
- *Язык программирования, на котором описывается интерпретатор базового sc-языка программирования.*

⇒ *примечание**:
Этот язык, в частности, может использоваться как язык микропрограмм для sc-компьютера, обеспечивающего аппаратную интерпретацию базового sc-языка программирования (языка SCP).

}

sc-агент

:= [коллектив агентов, взаимодействующие друг с другом через общую sc-память]

⇒ *разбиение**:
область применения

= { • *интерпретация программ различных sc-языков программирования высокого уровня*

- *информационный поиск*
- *реализация правил логического вывода, соответствующих самым различным логическим исчислениям*
- *сведение задач к подзадачам*
- *анализ качества хранимой базы знаний, в частности, ее корректности, полноты*
- *обнаружение и автоматическое склеивание синонимичных sc-элементов*

}

sc-модель интеллектуальной системы

:= [унифицированная логико-семантической моделб ИС]

:= [логическая система, построенная но основе абстрактной sc-машины]

⇒ *включает**:

= { • *интегрированную совокупность всех знаний, которые необходимы для функционирования ИС и которые представлены в виде интегрированного sc-текста*

⇒ *пояснение**:
такую семантическую модель базы знаний можно называть sc-моделью базы знаний или sc-текстом базы знаний

- *абстрактную sc-машину, в памяти которой хранится указанный sc-текст базы знаний*

}

3 ФОРМАЛЬНАЯ СЕМАНТИЧЕСКАЯ СПЕЦИФИКАЦИЯ БИБЛИОГРАФИЧЕСКИХ ИСТОЧНИКОВ

В.В. Голенков, Н.А. Гулякина, И.Т. Давыденко, Д.В. Шункевич.
СеманТПИСuCAK-2019ст

⇒ *ключевой знак**:

- *многоагентная система*
- *интерпретатор знаний и навыков*
- *платформенная независимость системы*
- *интерпретатор языка SCP*
- *фон-неймановская архитектура системы*
- *семантический ассоциативный компьютер*
- *нелинейная память*

⇒ *аннотация**:

[В статье проведен анализ проблемы обеспечения совместимости компьютерных систем, рассмотрены основные принципы, лежащие в основе технологии OSTIS, одной из задач которой является решение данной проблемы. Отдельное внимание уделено принципам построения семантических ассоциативных компьютеров, являющихся аппаратной реализацией интерпретатора логико-семантических моделей компьютерных систем, разрабатываемых по технологии OSTIS.]

В.В. Голенков , Н.А. Гулякина. ПроектОСТКПИС-2014ст

⇒ *ключевой знак**:

- *sc-машина*
- *sc-агент*
- *sc-модель интеллектуальной системы*

⇒ *аннотация**:

[Статья является второй в цикле статей, посвященных рассмотрению открытого проекта, направленного на создание и развитие технологии компонентного проектирования интеллектуальных систем. В работе рассматривается унификация семантических моделей обработки знаний - моделей информационного поиска, моделей интеграции знаний, моделей решения задач, моделей трансляции семантических сетей во внешнее представление и обратно. На основе унифицированных семантических моделей интеллектуальных систем рассмотрена модель их компонентного проектирования, основанная на выделении многократно используемых компонентов интеллектуальных систем и на обеспечении платформенной независимости их проектирования. Рассмотрены также средства обеспечения открытого характера технологии проектирования интеллектуальных систем, методика их эволюционного проектирования и принципы построения метасистемы, предназначенной для комплексной поддержки проектирования интеллектуальных систем.]

ЗАКЛЮЧЕНИЕ

В ходе ознакомительной практики были приобретены ценные компетенции в сфере формализации текстового материала. Проведена тщательная работа по подбору соответствующей литературы, детальному разбору источников и выявлению ключевых элементов.

Изучен теоретический базис Стандарта OSTIS с целью последующей интеграции собственной формализации. Кроме того, строго соблюдались синтаксические правила оформления формализованной теории.

В процессе практической деятельности были дополнены уже существующие формализованные концепты в монографии посредством добавления примечаний, разъяснений и конкретных примеров. Параллельно была проведена формализация дополнительной информации, касающейся формальной онтологии множеств, связей и отношений.

В результате выполнения ознакомительной практики были приобретены необходимые знания и умения в области формализации текстовых данных с соблюдением установленных стандартов и требований.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] В.В. Голенков, Н.А. Гулякина. Проект открытой семантической технологии компонентного проектирования интеллектуальных систем. Часть 2: унифицированные модели проектирования / Н.А. Гулякина В.В. Голенков. — Белорусский государственный университет информатики и радиоэлектроники, 2014. — С. 53.

[2] В.В. Голенков Н.А. Гулякина, И.Т. Давыденко Д.В. Шункевич. Семантические технологии проектирования интеллектуальных систем и семантических ассоциативных компьютеров / И.Т. Давыденко Д.В. Шункевич В.В. Голенков, Н.А. Гулякина. — Белорусский государственный университет информатики и радиоэлектроники, 2019. — С. 50.