Министерство образования и науки Российской Федерации

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (ТГУ)

Факультет информатики

Кафедра теоретических основ информатики (ТОИ)

**КУРСОВАЯ РАБОТА**

ИССЛЕДОВАНИЕ ХАРАКТЕРИСТИК ГРАФА, ПОСТРОЕННОГО НА ОСНОВЕ ДАННЫХ СВОБОДНОЙ БИБЛИОТЕКИ WORDNET

Озерова Александра Игоревна

Руководитель,

канд. тех. наук

\_\_\_\_\_\_\_\_\_\_\_\_М.С.Пожидаев

*подпись*

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_2014г.

Студент группы №\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_А.И.Озерова

*подпись*

Томск 2014

**Оглавление**

Введение.

Глава 1. Структура тезауруса WordNet и библиотеки JWNL

* 1. Структура WordNet
  2. Библиотека JWNL

Глава 2. Описание реализации

2.1 Объекты класса WordnetHandler

2.2 Объекты класса Graph

2.3 Объекты класса Node

Глава 3. Анализ полученных данных

Заключение.

Список литературы.

Введение

Вслед за появлением вычислительной техники, исследователей в области искусственного интеллекта уже на протяжении долгого времени волнуют проблемы машинной обработки текста, но принимая во внимание скачок в развитии информационных технологий и смежных дисциплин, удовлетворительного решения таких проблем пока не найдено.

Курсовая работа посвящена исследованиям тесно связанным с задачами обработки естественного языка. Как известно, крайне затруднительно проводить какие-либо работы в этой сфере без задействования словарей и онтологий любых типов.

Онтология — это попытка всеобъемлющей и детальной формализации некоторой области знаний с помощью концептуальной схемы. Обычно такая схема состоит из структуры данных, содержащей все релевантные(адекватные, понятные, практически применимые) классы объектов, их связи и правила (теоремы, ограничения), принятые в этой области.

Онтологии используются, как форма представления знаний о реальном мире или его части. Основные области применения онтологий: ядерные методы обучения, задачи интерпретации текста, задачи извлечения данных, моделирование бизнес-процессов.

Доступными вариантами онтологий, которые могли быть использованы в ходе выполнения курсовой работы, являются граф Wordnet, граф связей Wikipedia, комбинация Wordnet и Wikipedia. Но выбор был остановлен на тезаурусе английских слов WordNet.

Целью курсовой работы является определение степени пригодности Wordnet при решении практически значимых задач. Для достижения этой цели необходимо оценить характеристики графа, построенного на основе данных WordNet.

Соответственно имеем следующие задачи:

- представление данных тезауруса WordNet в виде графа

- исследование характеристик полученного графа

- изучение применимости полученных результатов для реализации алгоритма метода опорных векторов

Языком для написания программной части был выбран язык Java, а также использована библиотека JWNL (Java WordNet Library) для упрощения работы с данными WordNet.

**Глава 1. Структура тезауруса WordNet и библиотеки JWNL**

* 1. **Структура WordNet**

WordNet — это электронный тезаурус/семантическая сеть слов английского языка, был разработан и выпущен в Принстонском университете.

WordNet можно свободно использовать в коммерческих и научных целях. Для работы с ним существует несколько программ, множество интерфейсов и API, реализуемых как на большинстве возможных языков, так и с помощью протокола DICT, программы GoldenDict и других. Также, пакеты WordNet присутствуют в некоторых репозиториях программного обеспечения для GNU и Linux и их дистрибутивов.

Словарь состоит из 4 сетей для основных знаменательных частей речи: существительных, глаголов, прилагательных и наречий. Базовой словарной единицей в WordNet является не отдельное слово, а так называемый синонимический ряд - синсет, объединяющий слова со схожим значением. Каждый синсет представляет узел сети. Для удобства использования словаря каждый синсет дополнен описанием и небольшими примерами употребления слов в контексте. Слово или словосочетание может встречаться более чем в одном синсете и иметь более одной категории части речи. Каждый синсет содержит список синонимов или синонимичных словосочетаний и указатели, которые описывают отношения между ним и другими синсетами. Слова, имеющие несколько значений, включаются в несколько синсетов и могут относиться к различным синтаксическим и лексическим классам.

Синсеты в WordNet могут быть связаны между собой различными семантическими отношениями:

гипероним (breakfast (завтрак) → meal (прием пищи)) (термин «meal» является гиперонимом по отношению к термину «breakfast», поскольку «meal» - слово с более широким значением, выражающее общее, родовое понятие, название класса(множества), в которое входит слово «breakfast»);

гипоним (meal (прием пищи) → lunch (обед) (термин «lunch» является гипонимом по отношению к термину «meal», поскольку «lunch» - понятие, выражающее частную сущность по отношению к понятию «meal»);

has-member (faculty (факультет) → professor (профессор)) (понятие «professor» является участником или членом понятия «faculty»);

member-of (pilot (пилот)→ crew (экипаж)) (понятие «crew» подразумевает сложную сущность, участником или членом которой является понятие «pilot»);

мероним: has-part (table (стол) → leg (ножка)) (понятие «leg» является составной частью понятия «table»);

антоним (truth(правда) → lie(ложь)) (понятие «lie» имеет прямо противоположное понятию «truth» лексическое значение).

Также, существуют различные другие связи: лексические, антонимические, контекстные (слово 'x' имеет отношение к слову 'y') и другие. Особую роль среди них играет гипонимия: она позволяет организовывать синсеты в виде семантических сетей. Для разных частей речи родовидовые отношения могут иметь дополнительные характеристики и различаться областью охвата.

Поскольку WordNet может рассматриваться как онтология, его структура представляет собой граф. Заполнение и дальнейшее исследование графа невозможно без получения исходных данных из WordNet. И в силу того, что языком программной части стал язык Java, то в качестве упрощения работы с WordNet взята библиотека JWNL.

**1.2 Библиотека JWNL**

Библиотека JWNL позволяет не только получать слова и словосочетания из WordNet, но и определять отношения между ними, а также проводить морфологический анализ слов, то есть определять принадлежность заданного слова к той или иной части речи. Перечислим некоторые характерные особенности библиотеки: архитектура, позволяющая конфигурировать JWNL с помощью файла настроек, локализация посредством использования пакета свойств, ведение журнала, универсальность кода, для облегчения работы с другими WordNet подобными словарями, исправление ошибок.

Установка и использование библиотеки JWNL настолько просты и требуют так мало дополнительного кода, насколько это возможно.

Конфигурирование библиотеки файлом настроек привело к модульной архитектуре. Например, в файле настроек, можно указать, какой тип интерфейса словаря вы хотите использовать через тег <dictionary>. Класс, который вы укажите, загружается при инициализации и устанавливается как единственный экземпляр класса Dictionary.

Возможность работы с различными WordNet подобными словарями обеспечивается тем, что ответственность за создание объектов делегируется фабричному классу, который, как и класс Dictionary, может быть легко изменён с помощью файла настроек.

Использовать JWNL очень просто. Во-первых, необходимо вызвать JWNL.initialize() где-то в коде инициализации программы. Затем, просто прописать Dictionary.getInstance (), чтобы получить установленный словарь.

Единственные методы словаря, которые вам действительно могут понадобиться это - lookupIndexWord(), lookupAllIndexWords(), и getIndexWordIterator(). Другие методы которые могут быть интересны это - PointerUtils.Relationship.findRelationships(). Relationship.findRelationships() позволяет выявить связи определённого типа между двумя словами. Другой способ понимания отношения - это путь от синсета первого слова к синсету второго слова.