на инструктаж, проведение психодиагностики и обработку результатов; повышение точности результатов экспериментов, за счет машинной обработки исходных данных; возможность анализа накопленных данных об испытуемых и результатах диагностических исследований; снижение трудозатрат на проведение тестирования.

Список использованных источников:

1. Эндрю Троелсен. Язык программирования C# 5.0 и платформа .NET 4.5 – Москва, 2013. – 1312 с.
2. Немов, Р. С. Психология. Кн. 3: Психодиагностика. Введение в научное психологическое исследование с элементами математической статистики. – Москва, 2001. – 640 с.

ПОСТРОЕНИЕ ГРАФА СЛОВ С ПРИМЕНЕНИЕМ ТОМИТА-ПАРСЕРА

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Огородник Р.В.

Серебряная Л.В. – кандидат техн. наук, доцент

Для более эффективной обработки текстовой информации необходимо всячески структурировать её. Существует множество способов структурирования, но наиболее доступным и наглядным является построение графа слов. Для этого удобно использовать Томита-парсер.

Томита-парсер представляет собой программное средство для анализа текстов. Он использует GLR-алгоритм — расширенную версию алгоритма LR-парсинга. Но LR-алгоритм предназначен для анализа текстов, написанных на достаточно строго детерминированных языках программирования, и с естественными языками работать не может. Томита решил эту проблему путём параллелизации стеков, что позволило рассматривать различные трактовки одних и тех же участков текста: как только возникает возможность различной трактовки, стек разветвляется. В процессе анализа ошибочные ветви отбрасываются, а результатом работы становится наиболее длинная цепочка. Плюсом алгоритма также является то, что он выдаёт результаты по мере продвижения по тексту, в отличии от других парсеров.

Для создания программы, которая будет реализовывать разбор текста на слова, с последующим построением графа с помощью Томита-парсера необходимы следующие входные файлы конфигурации:

- Конфигурационный файл
- Корневой словарь
- Файл грамматики
- Файлы описания типов фактов и ключевых слов (опционально)

Конфигурационный файл описывает основные параметры подключаемого парсера. Корневой словарь содержит слова и статьи, из которых производится разбор – словарь языка. Файл грамматики определяет список правил, на основе которых будут извлекаться слова из предложений. Грамматика представляет собой цепочки, описанные на специальном формальном языке. Структурно правила разделяются символом «->» на левую и правую части. В левой части располагается один нетерминал, а правая состоит как из терминалов, так и нетерминалов. Терминалом в данном случае называется некоторый объект, имеющий неизменное значение. Множество терминалов представляет собой алфавит языка Томита, из которого выстраиваются все остальные слова. Терминалами в Томита-парсере выступают «леммы» - слова в начальной форме, записанные в одинарных кавычках части речи (Noun, Verb, Abj), знаки пунктуации (Comma, Punct, Hyphen), другие спецсимволы (Percent, Dollar). После создания грамматики с помощью словаря создаётся статья. Статья словаря описывает способ выделения цепочки слов в анализируемом тексте. Цепочку можно составлять при помощи списка ключевых слов, упомянутого выше, встроенного в парсер алгоритма, и других способов, которыми можно расширить парсер на уровне исходного кода. Статья словаря состоит из типа, названия и содержания.

Следующим шагом является введение процедуры интерпретации, то есть преобразования извлечённых цепочек в факты. Сначала необходимо создать структуру того факта, который необходимо извлечь, а именно, описать, из каких полей он состоит. Для этого создаётся файл описания типов фактов, который после импорта базовых типов дополняется специальными фактами, описание которых состоит из названия, базового типа факта, от которого будет производиться наследование и перечислены поля факта.

Также с помощью цепочки правил формального Томита-языка можно составлять специфические правила, с использованием в цепочках подобия регулярных выражений для выделения более сложных структур из текстов. В качестве специальных структур в Томита-парсере существуют пометы-ограничения. Это уточняющие свойства терминалов и нетерминалов стркрутуры, которые накладывают ограничения на цепочки, описываемые терминалом или нетерминалом. Они записываются после терминалов/нетерминалов и, в случае нетерминалов применяются к синтаксически главному слову группы. Некоторые пометы представляют собой унарный оператор, некоторые имеют поле, которое может быть заполнено различными значениями.

Ещё одним свойством Томита-парсера является возможность комбинировать словари и грамматики

и пропускать обрабатываемый текст последовательно через несколько грамматик и словарей в заданном порядке, таким образом добиваясь на каждой итерации своих результатов. Результатом работы парсера являются выделенные с помощью грамматик и словарей слова и словоцепочки, а также граф предложений, который впоследствии можно преобразовать в единый текстовый граф слов, и с помощью этого графа можно будет выделить знания и данные, которые относятся к ключевому или выделяемому слову, соотнести факты относящиеся к одному понятию и произвести сооружение дополнительных связей на основе этого графа.

На основе вышеизложенного можно сделать вывод, что Томита-парсер, с такими произведенными улучшениями, как агрегатор графов предложений в общий граф слов и грамматики, описывающие наиболее характерные для данного текста цепочки является мощным инструментом в обработке текстовой информации.

Список использованных источников:

1. [Электронный источник] Технологии Яндекса — Томита-парсер. https://tech.yandex.ru/tomita/ Дата доступа 10.03.2015 г.
2. [Электронный источник] GitHub. GLR-парсер. https://github.com/vas3k/python-glr-parser Дата доступа 10.03.2015 г.

АЛГАРЫТМ АЎТАМАТЫЧНАГА ВЫЗНАЧЭННЯ МЕСЦА НАЦІСКУ Ў НЕВЯДОМЫХ СЛОВАХ У ЛІНГВІСТЫЧНАЙ ІНФАРМАЦЫЙНА-ПОШУКАВАЙ СІСТЭМЕ

Беларускі дзяржаўны ўніверсітэт інфарматыкі і радыёэлектронікі
г. Мінск, Рэспубліка Беларусь

Філіпчык А. В.

Сярэбраная Л. В. — к. т. н., дацэнт

Разгледжаныя асаблівасці вызначэння націску. Прапанаваны алгарытм, які падыходзіць для рашэння пастаўленай задачы вызначэння націскнога складу ў невядомых словах.

Для многіх сістэм, якія працуюць з тэкставымі дадзенымі, неабходна ведаць не толькі правільнае напісанне слова, але яшчэ і правільнае яго вымаўленне, у прыватнасці, трэба ведаць месца націску. Напрыклад, такія дадзеныя вельмі важныя для сістэм сінтэзу маўлення, альбо для камерцыйных прадуктаў тыпу Soundex, што прымяняецца авіякампаніямі для захоўвання імён і прозвішчаў пасажыраў у фанетычнай форме для прадухілення канфліктных сітуацый з няправільным напісаннем ідэнтыфікацыйных дадзеных у розных мовах і алфавітах. Звычайна такія сістэмы аперыруюць слоўнікамі, у якіх захоўваецца ўся неабходная інфармацыя пра кожную лексічную адзінку, але натуральныя мовы імкліва развіваюцца, і ні адзін слоўнік не можа ўмясціць абсалютна ўсе славаформы, якія могуць сустрэцца сістэме. Для вызначэння націску ў невядомых словах неабходны адмысловыя алгарытмы, гэтаму і прысвечана дадзеная праца.

На ўваход распрацаванага алгарытма могуць паступаць як асобныя словы, так і цэлыя тэксты.

Алгарытм вызначэння націску можна падзяліць на наступныя этапы:

1. Марфалагічны аналіз і пошук слова ў базе дадзеных.
2. Вызначэнне стандартных прэфіксаў.
3. Вызначэнне стандартных суфіксаў і канчаткаў.
4. Прадказанне націску на аснове статыстыкі.

На першым этапе алгарытм праводзіць марфалагічны аналіз кожнай лексічнай адзінкі для выяўлення выпадкаў амаграфіі. Відавочна, што зварот да слоўніка не заўсёды дазваляе адназначна вызначыць прыналежнасць славаформы да той ці іншай лексемы. Амаграфы – гэта славаформы з аднолькавым напісаннем, якія, тым не менш, належаць да розных лексем і могуць адрознівацца націскам. Метад кантэкстнага аналіза ўлічвае славаформы ў левым і правым кантэксце і падлічвае імавернасць з'яўлення ў дадзеным кантэксце той ці іншай граматычнай формы. Пасля марфалагічнага аналізу алгарытм шукае слова ў базе славаформ па вызначаных характарыстыках. База славаформ пабудавана на аснове слоўнікаў беларускай мовы, узятых з адкрытых крыніц (усяго больш за 500.000 уваходжанняў). Аднак у гэтыя слоўнікі ўключаныя далёка не ўсе існыя словы: так, у іх адсутнічаюць шматлікія імёны, назвы, рэгіяналізмы, аўтарскія словы, неалагізмы. Між тым, алгарытм павінны з высокай імавернасцю правільна вызначаць націск ва ўсіх словах.

Другі этап - вызначэнне стандартных прэфіксаў. У аснове метада вызначэння націска - формула $p = (n+1)/2$, дзе n – колькасць складоў у слове. Гэты алгарытм дае добрыя вынікі ў кароткіх словах, але ў доўгіх словах, асабліва складаных, дае даволі вялікую колькасць памылак. Для паляпшэння вынікаў выкарыстоўваецца механізм вылучэння стандартных прэфіксаў. У спіс прэфіксаў уваходзяць як прыстаўкі, так і першыя часткі складаных словаў (такія як пяцьсот-, трактара-, стара-, электра- і г.д). Слова дзеліцца на дзве часткі: прэфікс і аснову, якая зноў шукаецца ў слоўніку. Некаторыя прэфіксы маюць уласны націск,