**Семинар 13 июня**

**Ю. Шучалова «Разработка лексико-семантических шаблонов в информационных системах»**

При работе с корпусами текстов нам часто требуется выделить элементы текста, например, слова, имеющие определённое свойство или словосочетания, имеющие определённую структуру – т.е. нанести на корпус разметку. В частности, в нашем исследовании нам необходимо выделять конструкции, соответствующие разным маркерам академического стиля. Иногда нужно просто выделить нужные слова – предлоги или существительные, в других случаях, как с препозитивными и постпозитивными определениями, требуется выделить множество разных форм этого явления.

Как мы можем разметить корпус? Самое очевидное решение – сделать это вручную с помощью специалистов-экспертов. Но тогда мы потратим огромное количество времени и сил, даже если нужно будет найти только определённые слова. Следовательно, процесс нанесения разметки необходимо автоматизировать. Чтобы объяснить компьютеру, какие слова и словосочетания могут являться искомым нами, например, маркером стиля, мы можем воспользоваться лексико-синтаксическими шаблонами.

Лексико-синтаксический шаблон – декларативная структура, структурный образец языковой конструкции, который отображает её лексические и поверхностные синтаксические свойства [1]. Иными словами, это описание некоторого смыслового явления в виде чёткого шаблона. Синтаксис и семантика часто тесно связаны, поэтому такие шаблоны называют и лексико-синтаксическими, и лексико-семантическими. Я буду чаще использовать первый термин, т.к. он является более привычным в нашем исследовании.

Итак, рассмотрим основные области применения лексико-синтаксических шаблонов.

Во-первых, с их помощью можно автоматически выделять ключевые словосочетания в тексте. Самый элементарный сценарий использования - по некоторому шаблону, например, прилагательное + существительное, выделяются словосочетания, а потом исследуется частота их совместного появления в тексте.

Следующая область применения – извлечение онтологий и автоматическое построение связей между понятиями. Допустим, по шаблону «существительное 1»+ «работает в»+«существительное 2» можно из предложения «учитель работает в школе» провести связь «место работы» между понятиями «учитель» и «школа».

Третий способ применения лексико-синтаксических шаблонов я уже упоминала – это автоматическая разметка, аннотирование текста. В соответствии с заданным шаблоном ставится метка/тег определённого типа.

В принципе, применение лексико-синтаксических шаблонов не ограничено и они могут использоваться для извлечения разной информации, расширенного поиска и т.д.

Существует множество способов представления лексико-синтаксических шаблонов, и по большей части выбор способа зависит от используемого инструментария, программы или языка программирования. Рассмотрим три: регулярные выражения, специально разработанные структуры и специализированные языки.

Регулярные выражения сами по себе могут быть представлены в разных нотациях, но состоят они как правило из одинаковых типов элементов. При работе с лексико-синтаксическими шаблонами в регулярных выражениях можно выделить следующие элементы [2]:

- лексические характеристики, в том числе:

* конкретная форма слова
* лексическая категория, часть речи (например, «прилагательное»)
* корень слова
* концептуальная категория (например, «человек»)

- логические операторы – И, ИЛИ, НЕ

- специальные символы, обозначающие некоторое количество любых идущих подряд слов (токенов) – 0 или 1, 0 или более, 1 или более

- группирующие операторы (разные скобки - <>, [])

- символы, указывающие на количество повторений элемента: 0 или более раз, 1 или более раз

- диапазоны чисел

- необязательные конституенты

С помощью такого набора можно описать большинство конструкций языка. Регулярные выражения существуют во многих языках программирования, и в этом заключается удобство и универсальность данного способа представления. Часть лексико-синтаксических шаблонов в созданных для нашей программы плагинов использует регулярные выражения в нотации языка Java.

Следующий способ представления – специализированные структуры. В качестве примера можно привести шаблоны JAPE, которые используются в программе GATE Developer. В целом сценарии JAPE имеют много общего с регулярными выражениями, но они изначально направлены на использование существующей базовой морфологической разметки и используют появляющиеся в результате нанесения этой разметки характеристики, например, начальную форму слова или регистр входящих в него букв. Разметка текста на основе сценариев JAPE производится в программе GATE с помощью встроенного плагина.

И последний способ, о котором я расскажу – это специализированные языки, разработанные для описания и обработки лексико-синтаксических шаблонов. Для русского языка существует две разработки, обе по стечению обстоятельств называются LSPL – Lexico-Syntactic Pattern Language. Но разрабатывались они разными группами независимо друг от друга.

Первая разработка – группы Е. И. Большаковой [4]. Описываемый ими язык содержит следующие элементы:

- литералы («конкретные слова»)

- символьные обозначения частей речи

- символьные обозначения грамматических конструкций

- условия, уточняющие грамматические характеристики рассмотренных элементов.

Второй язык LSPL разработан Е.А. Рабчевским [5] и представляет собой XML-подобный язык.

Кто занимается собственно созданием шаблонов для выделения различных языковых явлений? Конечно же, эксперты – лингвисты. В отдельных случаях, при хорошем знании языка и наличии методических материалов, роль эксперта может выполнять сам разработчик информационной системы.

В нашем проекте лексико-синтаксические шаблоны выделяются с помощью плагинов для программы GATE Developer. Есть плагины для выделения определённых шаблонов, встроенный плагин для обработки сценариев JAPE, а также плагины для выделения произвольных шаблонов.

При работе с морфологической разметкой, которая наносится на корпус с помощью встроенных ресурсов GATE, мы, как и разработчики программы GATE, пользуемся условными обозначениями частей речи, с которыми можно ознакомиться в списке Penn Treebank Part-of-speech tags. Так, например, здесь есть разные обозначения для существительных в зависимости от того, нарицательное оно или собственное, и находится ли оно в единственном или во множественном числе.

С помощью плагинов для выделения определённых шаблонов мы ищем: глаголы разного времени, глаголы широкой абстрактной семантики, десемантизированные глаголы, усилительные наречия. Эти шаблоны имеют одну из следующих особенностей:

- конструкция состоит из нескольких слов;

- состав конструкции довольно неопределённый;

- есть несколько условий определения той или иной категории выражения;

- использование изменяемого списка слов, который может пополняться экспертом.

Пример: конструкция [<will>{<наречие>}]<глагол> выделяет глагол в будущем времени. Состав конструкции неопределённый в том смысле, что между will и глаголом может быть любое количество наречий, которые, тем не менее, не должны разрывать две части этой глагольной конструкции и поэтому выделяются вместе.

Пример выражения: I will absolutely definitely do that.

Сценарии JAPE в нашем проекте используются для выделения устойчивых конструкций, например, всех существительных, существительных с суффиксом –or, слов I/We, глаголов в пассивном залоге.

Пример сценария для выделения существительных с суффиксом –or:

Input: Token

Options: control = appelt debug = true

Rule: OrSuffix1

(

(

{Token.category==~"NN.\*", Token.string=~"\\w+[Oo][Rr]\\b"}

)

)

:potential

-->

:potential.OrSuffix={}

Для выделения препозитивных и постпозитивных определений при существительных нам приходится использовать плагины для выделения произвольных шаблонов. Здесь для выделения аннотаций одного типа используется множество не связанных между собой конструкций. И список этих конструкций, т.е. соответствующих шаблонов, может изменяться и пополняться. У нас все конструкции в виде регулярных выражений хранятся в отдельном файле и при необходимости могут быть отредактированы экспертом.

Пример: постпозитивное определение типа <существительное> + <наречие>, в виде регулярного выражения - ^NN[SP(PS)]?$ ^RB$

Пример выражения: the room upstairs.

В заключение доклада стоит отметить, что использование лексико-синтаксических шаблонов позволяет упростить работу по разметке корпуса для экспертов, однако это не всегда обеспечивает стопроцентную точность и полноту результатов. Причина в том, что не всегда конструкцию можно описать так, чтобы компьютер понял различия, например, между препозитивным определением «существительное»+ «существительное» и двумя существительными, стоящими рядом в рамках другой языковой конструкции. Т.е. семантика не всегда может быть отражена в конструкции. Но тем не менее, получаемые с помощью лексико-синтаксических шаблонов результаты позволяют быстро приступить к исследованию встречаемости большинства маркеров стиля.

Библиографический список

1. *Большакова Е.И.* Лексико-синтаксические шаблоны в задачах автоматической обработки текстов / *Н.В. Баева, Е.А. Бордаченкова, Н.Э. Васильева, С.С. Морозов* // Компьютерная лингвистика и интеллектуальные технологии: Труды Международной конференции Диалог ‘2007 / Под ред. *Л.Л. Иомдина, Н.И. Лауфер, А.С. Нариньяни, В.П. Селегея* – М.: Изд-во РГГУ, 2007. − C. 70-75.
2. *Jacobs P.S., Krupka G.R., Rau L.F.* Lexico-semantic pattern matching as a companion to parsing in text understanding [Электронный ресурс] // URL: http://pdf.aminer.org/000/511/641/lexico\_semantic\_pattern\_matching\_as\_a\_companion\_to\_parsing\_in.pdf (дата обращения: 10.01.2015).
3. GATE JAPE Grammar Tutorial Version 1.0 [Электронный ресурс] // URL: https://gate.ac.uk/sale/thakker-jape-tutorial/GATE%20JAPE%20manual.pdf
4. *Большакова Е.И.* Язык лексико-синтаксических шаблонов LSPL: опыт использования и пути развития // Программные системы и инструменты: Тематический сборник, № 15 / Под ред. Королева Л.Н. – М.: Изд. отдел факультета ВМК МГУ; МАКС Пресс, 2014. − C. 15-26.
5. *Рабчевский Е.А.* Автоматическое построение онтологий на основе лексико-синтаксических шаблонов для информационного поиска. // Труды 11-й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» – RCDL’2009. – Петрозаводск, 2009. – С. 69-77.
6. Penn Treebank P.O.S. Tags [Электронный ресурс] // URL: https://www.ling.upenn.edu/courses/Fall\_2003/ling001/penn\_treebank\_pos.html