BULLETIN OF NATIONAL ACADEMY OF SCIENCES
OF THE REPUBLIC OF KAZAKHSTAN
ISSN 1991-3494
Volume 5, Number 5(2014), 5 – 10

UDC 519.68; 681.513.7;
316.472.45; 007.51/.52

COMPUTER-ORIENTED METHODS OF DEFINITION OF DEGREE OF SIMILARITY OF SENTENCES IN A NATURAL LANGUAGE

T.V. Batura[1], F.A. Murzin[1], A.A. Perfiliev[1],
B.S. Baizhanov[2], M.V. Nemchenko[2]
tatiana.v.batura@gmail.com, murzin@iis.nsk.su, a_perfilev@mail.ru,
baizhanov@hotmail.com, nemchenko.imim@mail.ru

[1]A.P. Ershov Institute of Informatics Systems, Russian Academy of Sciences, Siberian Branch
[2]Institute of Mathematics, Informatics and Mechanics,
Committee of Science of the Ministry of Education and Science of the Republic of Kazakhstan

Key words: Information Retrieval System, Link Grammar Parser, syntactic analysis, semantics, relevance

Abstract. The basic considered problem consists in constructing algorithms, which getting into a text structure can deduce an adequate estimation of relevance of the text to the search inquiry. It is important, that the given estimation would be based on a context of search inquiry and would not be limited only by keywords, their similarity or frequency. Authors offered to use semantic-syntactical relations between words obtained on output of the Link Grammar Parser program system. In article, two algorithms of calculation of degree of similarity of sentences in a natural language are described. The second of them uses the approach based on the mathematical logic. Methods are partially implemented in the iNetSearch information retrieval system.

УДК 519.68; 681.513.7;
316.472.45; 007.51/.52

МАШИННО-ОРИЕНТИРОВАННЫЕ МЕТОДЫ ОПРЕДЕЛЕНИЯ СТЕПЕНИ БЛИЗОСТИ ПРЕДЛОЖЕНИЙ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

Т.В. Батура[1], Ф.А. Мурзин[1], А.А. Перфильев[1],
Б.С. Байжанов[2], М.В. Немченко[2]
tatiana.v.batura@gmail.com, murzin@iis.nsk.su, a_perfilev@mail.ru,
baizhanov@hotmail.com, nemchenko.imim@mail.ru
[1]Институт систем информатики им. А.П. Ершова СО РАН
[2]Институт математики, информатики и механики КН МОН Респ. Казахстан

Ключевые слова: информационно-поисковая система, Link Grammar Parser, синтаксический анализ, семантика, релевантность

Аннотация. Основная рассматриваемая задача состоит в построении алгоритмов, которые, проникая в структуру текста, могут вывести адекватную оценку релевантности текста поисковому запросу. Важно, чтобы данная оценка была основана на контексте поискового запроса и не ограничивалась только ключевыми словами, их близостью или частотой. Авторами было предложено использовать семантико-синтаксические отношения между словами предложения, получаемые на выходе программной системы Link

Grammar Parser. В статье описаны два алгоритма вычисления степени близости предложений на естественном языке. Второй из них использует подход, основанный на математической логике. Методы частично реализованы в информационно-поисковой системе iNetSearch.

Работа выполнена при поддержке гранта 2581/ГФ3 МОН РК

1. Введение

В условиях стремительного роста объемов информационных ресурсов возникает необходимость повышения качества поиска информации. Многие исследователи, например [1, 2], склоняются к необходимости проведения глубокого семантического анализа текстов для создания их семантических образов, на основе которых можно проводить тонкое ранжирование документов. Этот подход, несомненно, наиболее разумный, однако требует тщательной и долгой работы над созданием соответствующих теорий и подходящих инструментов для автоматической обработки текстов [3]. В частности, может потребоваться детальное описание различных областей знаний. Поэтому имеет смысл также поиск частичных решений.

Основная задача состоит в построении алгоритмов, которые, проникая в структуру текста, могут вывести адекватную оценку релевантности текста поисковому запросу. Важно, чтобы данная оценка была основана на контексте поискового запроса и не ограничивалась только ключевыми словами, их близостью или частотой.

В процессе решения поставленных задач авторами было предложено использовать семантико-синтаксические отношения между словами предложения, получаемые на выходе программной системы Link Grammar Parser [4,5]. Предложен способ (базовый алгоритм) вычисления степени совпадения естественно-языковых конструкций. Отметим, что в данный момент исследования полностью ориентированы на англоязычные источники. На основе вышеупомянутых идей была реализована информационно-поисковая система (ИПС) iNetSearch [6,7]. Проведенное тестирование системы iNetSearch показало эффективность предложенного алгоритма в решении задач поиска информации.

Далее были предложены методы, которые обобщают подход, используемый в базовом алгоритме. Более точно, базовый алгоритм учитывает только так называемые инвариантные коннекторы, не принимая во внимание более сложную логику. Во втором случае применяются более тонкие методы. При сопоставлении двух предложений, точнее, при анализе их на близость осуществляется проверка ряда логических свойств. Примеры такого рода свойств: инвариантность коннектора, замена коннектора на дизъюнкцию других, расщепление коннектора на два коннектора, расщепление коннектора на два коннектора с инверсией и др. В настоящее время выделено 19 различных схем. Некоторые из них имеют несколько вариантов.

Однако можно высказать предположение, что дальнейшее развитие предложенного метода весьма затруднительно и не приведет к существенным улучшениям имеющихся результатов. Одной из причин является то, что на данном этапе возможности анализатора Link Grammar Parser почти полностью исчерпаны. Несмотря на то, что Link Grammar Parser обладает рядом преимуществ (высокая скорость работы, частичный охват семантики), он вынуждает оставаться на уровне синтаксиса с небольшим охватом семантики. Поэтому, чтобы получить существенное продвижение, необходимо перейти на более высокий уровень, к инженерии знаний.

2. Метапоисковая система iNetSearch

Система iNetSearch находится на стороне пользователя и требует подключения к сети Интернет. iNetSearch использует результаты запросов к существующим поисковым системам. Например, для тестирования использовался поисковый сервис Нигма.РФ (URL: http://www.nigma.ru), т.к. он переправляет запрос другим поисковым системам, тем самым, увеличивая возможный круг поиска. Реализованная система iNetSearch фильтрует результаты запросов.

Предложения на естественном языке, получаемые из результатов запросов (например, краткие сниппеты, которые выдал сервис Нигма.РФ), транслируются в синтаксические диаграммы системы Link Grammar Parser. Транслятор дополнительно проводит лемматизацию слов, приписывание метаинформации словам. Добавление синтаксических связей между словами, типизацию этих

связей. Link Grammar Parser также осуществляет приписывание зависимостей между придаточными предложениями. Это дает достаточно большой объем информации о предложении. Самое главное, что анализатор генерирует диаграммы синтаксического разбора, которые отображают синтаксические взаимосвязи между словами.

Основная задача состоит в том, чтобы оценить соответствие текста поисковому запросу. Делается это следующим образом. Имеются диаграмма синтаксического разбора для запроса и для конкретного предложения из текста. В базовом алгоритме предполагается, что если эти диаграммы похожи по лексике и по структуре связей, то мы признаем, что предложения (и в целом текст) релевантны запросу. В случае, когда учитываются перефразирования, обобщенный алгоритм на основе логического подхода принимает более изощренный вид, но в принципе идея та же.

3. Программная система Link Grammar Parser

Link Grammar Parser – это синтаксический анализатор английского языка, разработанный в 1990-е гг. в университете Корнеги-Мелона, базирующийся на некоторой теории. Отметим, что данная теория, вообще говоря, отличается от классической теории синтаксиса. Получив предложение, система приписывает к нему синтаксическую структуру, которая состоит из множества помеченных связей (коннекторов), соединяющих пары слов. Подробное описание системы можно найти в [4,5]

Link Grammar Parser имеет словари, включающие около 60000 словарных форм. Он позволяет анализировать большое число синтаксических конструкций, включая многочисленные редкие выражения и идиомы. Анализатор довольно устойчив; может пропустить часть предложения, непонятную ему, и определить структуру оставшейся части предложения. Он способен делать разумные предположения о синтаксической категории неизвестных ему слов (т. е. слов, которые отсутствуют в словарях) из контекста и написания. У анализатора есть данные об именах собственных, о числовых выражениях и разнообразных знаках препинания.

Правила соединения слов описаны в наборе словарей. Для каждого слова в словаре записывается, какими коннекторами оно может быть связано с другими словами предложения. Коннектор состоит из имени типа связи, в которую может вступать рассматриваемая единица анализа. Например, пометка S соответствует связи между субъектом и предикатом, O – между объектом и предикатом. Только основных, наиболее важных связей, имеется более ста. Для обозначения направления связи справа к коннектору присоединяется знак "+", слева – знак "–". Левонаправленный и правонаправленный коннекторы одного типа (см. Рис.1) образуют связь (link).

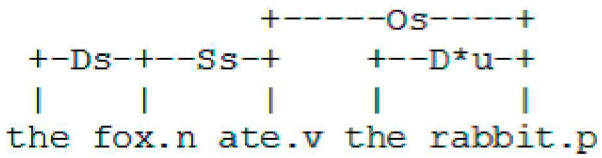

Рис. 1. Пример синтаксического разбора предложения

Получаемые диаграммы, по сути, являются аналогом так называемых деревьев подчинения предложений. В деревьях подчинения от главного слова в предложении можно задать вопрос к второстепенному. Таким образом, слова выстраиваются в древовидную структуру. Синтаксический анализатор может выдать две или более схемы разбора одного и того же предложения. Это явление называется синтаксической синонимией. Главной причиной, по которой анализатор называют семантической системой, можно считать уникальный по полноте набор связей (около 100 основных, причем некоторые из них имеют 3-4 варианта). В некоторых случаях тщательная работа над разными контекстами привела авторов системы к переходу к почти семантическим классификациям, построенным исключительно на синтаксических принципах. Так, выделяются следующие классы английских наречий: ситуационные наречия, которые относятся ко всему предложению в целом (clausal adverb); наречия времени (time adverbs); вводные наречия, которые стоят в начале предложения и отделены запятой (openers); наречия, модифицирующие прилагательные и т.д.

Из достоинств системы нужно отметить, что организация самой процедуры нахождения вариантов синтаксического представления очень эффективна. Построение идет не сверху вниз (top-

down) и не снизу вверх (bottom-up), а все гипотезы отношений рассматриваются параллельно: сначала строятся все возможные связи по словарным формулам, а потом выделяются возможные подмножества этих связей.

Это, конечно, приводит к алгоритмической непрозрачности системы, поскольку очень трудно проследить за всеми отношениями сразу. Во-вторых, не к линейной зависимости скорости алгоритма от количества слов, а к экспоненциальной, поскольку множество всех вариантов синтаксических структур на предложении из N слов в худшем случае равномощно множеству всех остовных деревьев полного графа с N вершинами.

Последняя особенность алгоритма заставляет разработчиков использовать таймер, для того чтобы вовремя останавливать процедуру, которая работает слишком долго. Однако все эти недостатки с лихвой компенсируются лингвистической прозрачностью системы, в которой с одинаковой легкостью прописываются валентности слова, причем порядок сбора валентностей внутри алгоритма принципиально не задается – связи строятся как бы параллельно, что полностью соответствует нашей языковой интуиции.

Отметим также отрицательные моменты.

1. Практическое тестирование системы показывает, что при анализе сложных предложений, длина которых превышает 25-30 слов, возможен комбинаторный взрыв, и результатом работы анализатора становится «панический» граф, как правило, случайный вариант синтаксической структуры, с лингвистической точки зрения неадекватной.

2. Применение описанных выше идей затруднено для флективных языков типа русского, ввиду значительно возрастающего объема словарей, которые возникают в силу морфологической развитости флективных языков. Каждая морфологическая форма должна описываться отдельной формулой, где нижний индекс входящего в нее коннектора должен будет обеспечивать процедуру согласования. Это приводит к усложнению набора коннекторов и к увеличению их количества. Для агглюнативных языков (например, тюркских) система станет еще более сложной.

4. Базовый алгоритм отождествления

4.1. Краткое описание алгоритма

Предполагаем, что мы работаем с деревьями, полученными в результате синтаксического анализа, проведенного системой Link Grammar Parser. Дальше производится «обобщение» таких деревьев. На этом этапе происходит нормализация словоформ. Могут быть произведены некоторые дополнительные преобразования предложений. Например, обратный порядок слов заменяется на прямой. Сложные формы глаголов «обрезаются» до простых форм. Глаголы могут переводиться в одну нормализованную форму в настоящем времени в простом виде. Сложные комбинации предлогов объединяются или даже удаляются. В результате получается «остов дерева», в котором удалены различные речеобразовательные конструкции. Такие деревья проходят процесс сравнения между собой (Рис.2). А именно, при определении релевантности текста запросу пользователя запрос сравнивается с предложениями в тексте.

Сначала производится сравнение лексики. Перед сличением слов, слова проходят простые фильтры на словоформу. В действительности, было бы нецелесообразно считать глагол и существительное одинаковым словом, но мы этим пренебрегаем. Само сличение слов производится достаточно просто. Проверяются гипотезы на соответствие двух слов по набору правил, если все правила проверены, и соответствие не выявлено, то слова считаются далекими по смыслу. Набор правил представляет собой условия, при которых всё-таки можно считать слова близкими. Это такие правила как непосредственное равенство слов, совпадение с точностью до окончания, синонимическая близость слов, наличие отношения гипоним-гипероним, слова с трансмутациями и прочие возможные не очень сложные варианты близости между словами.

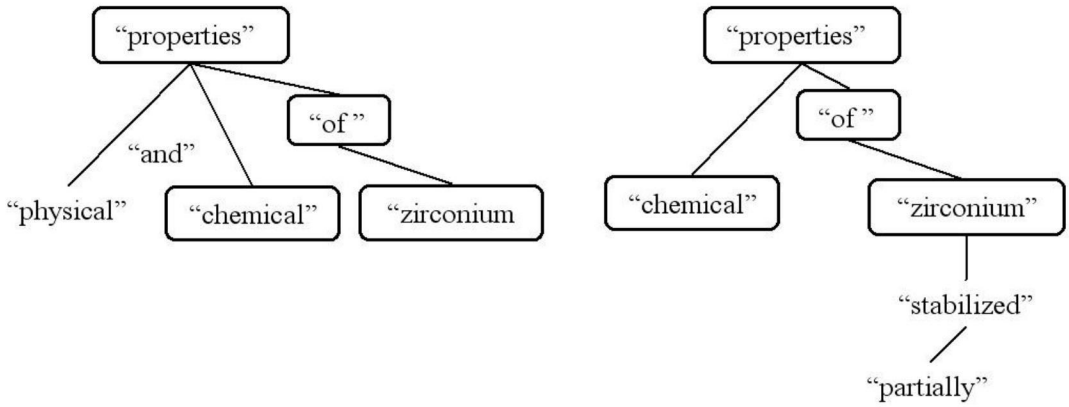

Рис.2: Пример сопоставления двух деревьев

4.2. Дополнительные возможности системы iNetSearch

Режим нечеткого поиска позволяет системе находить документы, которые содержат слова, похожие по написанию на слова запроса. Например, слова с опечатками: вкрапления отдельных букв, пропуски букв, перестановки рядом стоящих букв, замена символа на неправильный, перепутанная раскладка клавиатуры, некоторые просторечные выражения, сокращения, транслитерации и пр. Режим нечеткого поиска, настраиваемый в системе, также позволяет анализировать слова, написанные похожими символами из других языков и специальными символами, что обычно используется хакерами и спамерами для маскировки слов.

4.3. Сравнение связей

Далее предположим, что даны два предложения $\bar{x} =< x_1,\ldots,x_n >$, $\bar{y} =< y_1,\ldots,y_m >$, т.е. предложения рассматриваем, как вектор с компонентами из слов. Считаем, что произведен их разбор с помощью системы Link Grammar Parser. Рассмотрим множество всех таких пар $< i_1, i_2 >$, $< j_1, j_2 >$, что слова x_{i_1}, x_{i_2} и слова y_{j_1}, y_{j_2} соединены коннекторами одного и того же типа. При этом слова x_{i_1}, y_{j_1} и слова x_{i_2}, y_{j_2} близки в соответствии с тем или иным критерием. Например, их нормализованные формы одинаковые, они являются синонимами, слова похожие по написанию и т.д. Здесь возможна некоторая вариабельность алгоритма. Можно также игнорировать служебные слова: артикли, союзы, предлоги, междометия и др. Допустим теперь, что I – множество пар, упомянутых выше и принимаемых во внимание, и пусть его мощность $|I| = n$.

Далее пусть n_1, n_2 – количество коннекторов, получающихся в результате анализа предложений $\bar{x}, \bar{y}$ соответственно. В качестве меры похожести двух предложений можно ввести $\mu_0(\bar{x},\bar{y}) = n/\max(n_1,n_2)$ или $\mu_1(\bar{x},\bar{y}) = 2n/(n_1+n_2)$. В следующем разделе предложенный подход будет существенно обобщен. Окажется, что базовый алгоритм учитывает только так называемые инвариантные коннекторы, не принимая во внимание более сложную логику.

Таким образом, описанный выше метод позволяет ввести определенные меры близости между предложениями. Эти меры учитывают, как лексику, так и синтаксические отношения между словами. Минимальный вариант, дававший достаточно хорошие результаты, когда учитывались всего 8 связей: C, CC, S, SI, SF, SFI, SX, SXI.

Таблица 1 – Перечень наиболее важных связей системы Link Grammar Parser

Связь	Описание
C	соединяет подчинительный союз, глагол или прилагательное с подлежащим подчиненного предложения
CC	используется для соединения сочинительных союзов
S	соединяет подлежащее, выраженное существительным с глаголом
SI	соединяет подлежащее с глаголом в предложениях с инверсией главных членов предложения
SF	соединяет подлежащее, выраженное "it" или "there", с глаголом
SFI	соединяет подлежащее, выраженное "it" или "there", с глаголом вопросительном предложении с инверсией главных членов предложения
SX	используется для соединения местоимения "I" с глаголами "was" и "am"
SXI	используется для соединения местоимения "I" с глаголами "was" и "am" в случаях перестановки подлежащего и сказуемого

Были выделены 6 связей, учет которых мог существенно испортить ситуацию. Поэтому их целесообразно опускать. Всего в большей или меньшей мере анализу подверглись 45 связей.

5. Логические методы отождествления

Как и раньше считаем, что L – множество слов некоторого естественного языка. Для любого слова $x \in L$ обозначим $Norm(x)$ его нормализованную форму. Запись $Syn(x, y)$ обозначает, что x, y – синонимы.

Возникают два вида эквивалентностей:

1) $x_1 \approx x_2 \leftrightarrow x_1 = x_2 \vee Syn(x_1, x_2)$,

2) $x_1 \equiv x_2 \leftrightarrow Norm(x_1) = Norm(x_2)$.

Предложение рассматриваем, как вектор с компонентами из слов $\bar{x} =< x_1, \ldots, x_n >$. Функция $Norm$ может быть естественно распространена на предложения $Norm(\bar{x}) =< Norm(x_1), \ldots, Norm(x_n) >$. Текст $T =< \bar{x}_1, \ldots, \bar{x}_n >$ есть последовательность предложений.

Пусть запись $\bar{x} \models P(x_i, x_j)$ обозначает, что в схеме разбора предложения $\bar{x} =< x_1, \ldots, x_n >$ посредством анализатора Link Grammar Parser имеется коннектор типа P, идущий от слова x_i к слову x_j. Знак $\models$ означает, что фактически мы имеем дело с моделью. Основным множеством модели является множество пар $\{< 1, x_1 >, \ldots, < n, x_n >\}$. Так как одно и то же слово может входить в предложение два и более раз, то это приводит к необходимости рассмотрения именно пар, а не отдельных слов. Ввиду сказанного выше, корректным является даже обозначение $\bar{x} \models \varphi$, где φ – формула, например, исчисления предикатов первого порядка. Фактически $\bar{x}$ одновременно является обозначением и для вектора, и для модели.

Предположим, что даны два предложения $\bar{x} =< x_1, \ldots, x_n >$, $\bar{y} =< y_1, \ldots, y_m >$. Интерес представляют функции f такие, что $dom(f) \subseteq \{1, \ldots, n\}$, $range(f) \subseteq \{1, \ldots, m\}$ с дополнительными свойствами типа: $f(i) = j \rightarrow x_i \approx y_j$, $f(i) = j \rightarrow x_i \equiv y_j$ и другие подобные.

При сопоставлении двух предложений, точнее, при анализе их на близость осуществляется проверка ряда логических свойств. Например, пусть $f(i_1) = j_1, f(i_2) = j_2$. Теперь приведены примеры такого рода свойств.

Инвариантность коннектора

$\bar{x} \models P(x_{i_1}, x_{i_2}) \to \bar{y} \models P(y_{j_1}, y_{j_2})$.

Замена коннектора на дизъюнкцию других

$\bar{x} \models P(x_{i_1}, x_{i_2}) \to \bar{y} \models \bigvee_t Q_t(y_{j_1}, y_{j_2})$.

Расщепление коннектора на два коннектора

$\bar{x} \models P(x_{i_1}, x_{i_2}) \to \exists k\, (\bar{y} \models Q(y_{j_1}, y_k) \wedge R(y_k, y_{j_2}))$.

Расщепление коннектора на два коннектора с инверсией

$\bar{x} \models P(x_{i_1}, x_{i_2}) \to \exists k\, (\bar{y} \models Q(y_{j_2}, y_k) \wedge R(y_k, y_{j_1}))$.

Принимая во внимание, что $\bar{y}$ является обозначением для соответствующей модели, формула из третьего пункта может быть переписана в виде $\bar{x} \models P(x_{i_1}, x_{i_2}) \to \bar{y} \models \exists y Q(y_{j_1}, y) \wedge R(y, y_{j_2})$. В аналогичном виде может быть записана формула из четвертого пункта.

Ниже показан пример анализа двух предложений, одно из которых является перефразированным вариантом другого (см. Рис. 3).

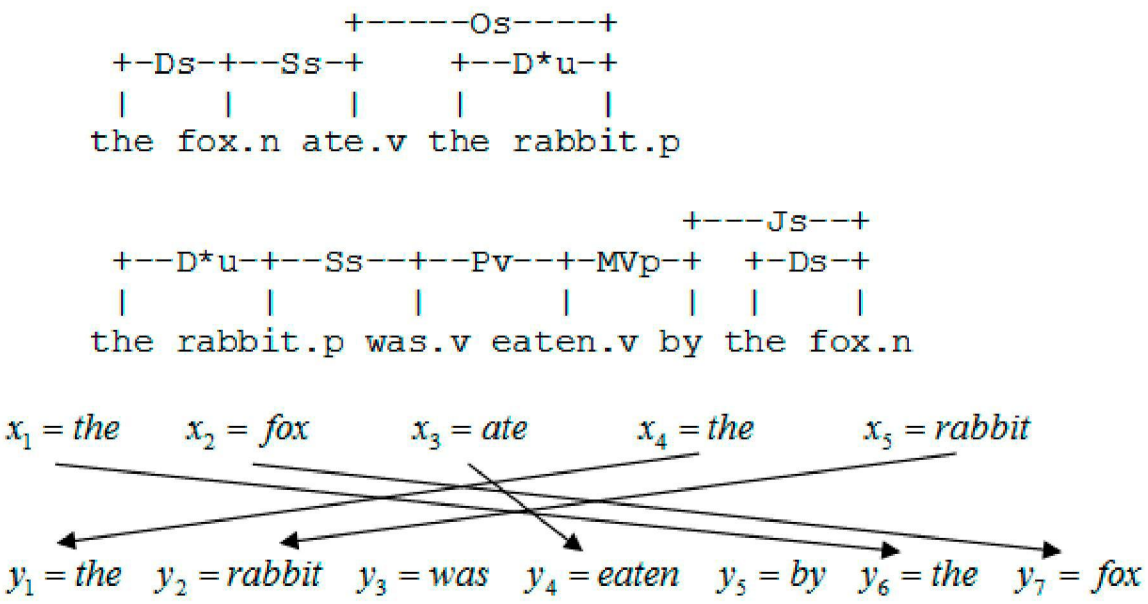

Рис. 3. Результаты работы анализатора Link Grammar Parser и действие функции f

Таким образом, имеем $f(1) = 6,\ f(2) = 7,\ f(3) = 4,\ f(4) = 1,\ f(5) = 2$.

При этом отображении получаем:

1) $Norm(ate) = Norm(eaten)$ или, что то же самое $ate \equiv eaten$;

2) коннекторы Ds и D*u сохраняются, т.е. они инвариантны;

3) $\bar{x} \models \mathrm{Ss}(fox, ate) \to \bar{y} \models \mathrm{MVp}(eaten, by) \wedge \mathrm{Js}(by, fox)$, т.е. имеет место расщепление коннектора Ss с инверсией;

4) $\bar{x} \models \mathrm{Os}(ate, rabbit) \to \bar{y} \models \mathrm{Ss}(rabbit, was) \wedge \mathrm{Pv}(was, fox)$, т.е. аналогично имеет место расщепление с инверсией, но другого коннектора Os.

Резюмируя можно сказать, что в нашем распоряжении имеются правила вида $R_i : \bar{x} \models \varphi_i(x_1, x_2) \to \bar{y} \models \psi_i(y_1, y_2)$.

Далее строится функция f, и проводится анализ, встречаются ли индексы $i_1, i_2,\ j_1 = f(i_1),\ j_2 = f(i_2)$ такие, что на конкретных словах из предложений $\bar{x}, \bar{y}$ выполнено

правило R_i, т.е. $\overline{x} \models \varphi_i(x_{i_1}, x_{i_2}) \to \overline{y} \models \psi_i(y_{j_1}, y_{j_2})$. Для простоты можно говорить, что правило выполняется на паре $< i_1, i_2 >$.

Рассмотрим множество всех таких пар $< i_1, i_2 >$, на которых выполнено одно из правил. Обозначим это множество I, и пусть его мощность $|I| = n$. Отметим, что анализатор Link Grammar Parser допускает между двумя словами наличие только одного коннектора. Поэтому будет выполняться не более, чем одно правило.

Далее пусть n_1, n_2 – количество коннекторов, получающихся в результате анализа предложений $\overline{x}, \overline{y}$ соответственно. В качестве меры похожести двух предложений можно ввести $\mu_0(\overline{x}, \overline{y}) = n / \max(n_1, n_2)$ или $\mu_1(\overline{x}, \overline{y}) = 2n/(n_1 + n_2)$. Предложенный подход обобщает подход, используемый в базовом алгоритме. Более точно, базовый алгоритм учитывает только инвариантные коннекторы, не принимая во внимание более сложную логику.

Рассмотрим пример сравнения двух предложений на похожесть:

```
        +-----Js----+
 +-Ss-+-MVp-+  +---Ds--+
 |  |   | |    |
he went.v to.r the institute.n
```

```
        +-----Js----+
 +-Ss-+-MVp-+  +---Ds--+---Mp--+---J---+
 |  |   | |    |     |     |
he went.v to.r the institute.n of Hydrodynamics
```

Легко видеть, что $n_1 = 4, n_2 = 6$. Далее видим, что все четыре коннектора Ss, MVp, Ds, Js из первого предложения сохраняются (инвариантны), поэтому $n = 4$. В итоге получаем $\mu_0(\overline{x}, \overline{y}) = 4/\max(4, 6) = 4/6 = 2/3$ и $\mu_1(\overline{x}, \overline{y}) = 2 \cdot 4/(4+6) = 8/10 = 4/5$. То есть мы видим, что эти меры близости различаются.

В заключение отметим, что на наши исследования, рассмотренные в данной главе, в значительной мере повлияли работы Лбова Г.С. [8] и Викентьева А.А. [9], в которых, в частности, рассматриваются различные меры близости между логическими формулами.

6. Заключение

Для демонстрации эффективности работы системы были произведены тестовые испытания на основе базового алгоритма. Были сформированы десять простых запросов из области неорганической химии. По каждому запросу были загружены списки адресов с их описанием, которые поисковики обычно выдают пользователю. По этим коротким описаниям (сниппетам; англ. snippet) производилась оценка ресурса. Для сравнения с поисковой системой (а именно с системой Нигма.РФ, т.к. она переадресует запросы другим системам) была составлена статистика.

Система оставляла релевантные ссылки, отбрасывая нерелевантные по ее мнению. В итоге выяснилось, что на проведенных тестах в среднем из 100 ссылок, полученных из поискового сервиса Нигма.РФ, система выделяла 5-15 качественных релевантных ссылок, около 5 ссылок система ошибочно принимала за релевантные и остальные отбрасывала, как нерелевантные, что соответствовало действительности. Это показывает, что данная система смогла произвести фильтрацию на хорошем уровне.

Далее было проведено сравнение двух методов сопоставления конструкций естественного языка – базового (используемого в первоначальной версии системы iNetSearch) и нового (с учетом перефразирования предложений), описанного в работах [9,10]. Запросы, перефразированные варианты которых необходимо было найти, составлялись по различным тематикам. Источниками запросов служили: коллекция научных статей более чем по 20-ти темам и коллекция текстов

общеобразовательного плана. Для оценки качества поиска использовались три различные числовые характеристики. В среднем поисковая система стала одобрять меньше нерелевантных документов и больше релевантных. С другой стороны отметим, что несмотря на предпринятые очень большие усилия, метод, учитывающий перефразирования, позволил улучшить работу системы iNetSearch, но незначительно. Логические методы, описанные в данной статье представляют собой дальнейшую серьезную проработку вопроса, но на практике детально они не тестировались.

Несколько слов о границах применимости методов. Очевидно, что предложенные методы применимы только к предложениям, которые достаточно корректно могут быть проанализированы системой Link Grammar Parser. То есть методы основаны на предположении, что на вход ему подается диаграмма связей, правильно отражающая связи между понятиями. Отметим, что Link Grammar Parser не всегда строит для предложения адекватную диаграмму связей. Более того, в большинстве случаев он строит на предложении несколько диаграмм связей, каждая из которых удовлетворяет требованиям к диаграммам и потому не может быть отброшена. Чаще всего это бывает вызвано тем, что в предложении имеет место частеречная омонимия, или же формально слова можно связать друг с другом по-разному, так что предложение получает разную интерпретацию.

Человек, пользуясь знаниями о предметной области, а также опытом, подсказывающим ему, какие слова в каких смысловых связях могут или не могут состоять, чаще всего может интерпретировать данную синтаксическую конструкцию однозначно и выбрать для данного предложения единственную диаграмму связей, предложенную Link Grammar Parser. Однако в самом анализаторе такие знания не заложены, вследствие чего он может выдать для предложения целый список диаграмм, и нельзя знать заранее, какой по счету будет идти «правильная» диаграмма (хотя чаще всего она выдается все-таки первой).

Предложенные методы не могут отождествлять перефразированные предложения в том случае, если в сравниваемых предложениях содержатся формально разные системы понятий, или же понятия связаны друг с другом разными семантико-синтаксическими отношениями, хотя предложения могут выражать одну и ту же мысль. В этих случаях необходимо привлечение дополнительных знаний о семантике слов, например, использование соответствующих баз знаний.

ЛИТЕРАТУРА

[1] Salton G. Automatic Information Organization and Retrieval, 1968, 514 p.

[2] Лезин Г.В., Тузов В.А. Семантический анализ текста на русском языке: семантико-синтаксическая модель предложения // Экономико-математические исследования: математические модели и информационные технологии. – СПб.: Наука, 2003. – Вып. 3. – С. 282–303.

[3] Батура Т.В., Мурзин Ф.А. Машинно-ориентированные логические методы отображения семантики текста на естественном языке: моногр. / Институт систем информатики им. А.П. Ершова СО РАН. Новосибирск: Изд. НГТУ, 2008. 248 с.

[4] Temperley D., Sleator D., Lafferty J. Link Grammar Documentation [Electronic resource]. – 1998. – Mode of access: http://www.link.cs.cmu.edu/link/dict/index.html (accessed 15 November 2012)

[5] Sleator D., Temperley D. Parsing English with a Link Grammar. Pittsburgh: School of Computer Science Carnegie Mellon University, 1991. – 93 p.

[6] Murzin F., Perfliev A., Shmanina T. Methods of syntactic analysis and comparison of constructions of a natural language oriented to use in search systems // Bull. Nov. Comp. Center, Comp. Science, 2010, Iss. 31, – P. 91-109.

[7] Перфильев А.А., Мурзин Ф.А., Шманина Т.В. Методы синтаксического анализа и сопоставления конструкций естественного языка, ориентированные на применение в информационно-поисковых системах // Вестник НГУ, том 9, выпуск 4, 2011. – С 50-59.

[8] Лбов Г.С. Методы обработки разнотипных экспериментальных данных // моногр. / Институт математики СО АН. – Новосибирск: Изд. Наука, 1981. – 160 с.

[9] Викентьев А.А., Викентьев Р.А. О метриках для формул от разнотипных переменных и мерах опровержимости // Труды второй международной молодежной школы-конференции «Теория и численные методы решения обратных и некорректных задач». 2011. Часть 1. – С. 192-209. [Электрон. ресурс]. Режим доступа: http://semr.math.nsc.ru/v8/c182-410.pdf (дата обращения: 18 августа 2014)

REFERENCES

[1] Salton G. Automatic Information Organization and Retrieval, 1968, 514 p.

[2] Lezin G.V., Tuzov V.A. The semantic analysis of the text in Russian: semantico-syntactical model of the sentence// Economic-mathematical researches: mathematical models and information technologies.– СПб.: Наука, 2003. –Is. 3. – P. 282–303. (in Russian)

[3] Batura T.V., Murzin F.A. The machine-oriented logic methods of representation of semantics of the text in natural language // The monograph / A.P. Ershov Institute of Informatics Systems SB RAS. – Novosibirsk: Publishing Company of NGTU, ISBN 978-5-7782-1138-4, 2008. – 248p. (in Russian)

[4] Temperley D., Sleator D., Lafferty J. Link Grammar Documentation [Electronic resource]. – 1998. – Mode of access: http://www.link.cs.cmu.edu/link/dict/index.html (accessed 15 November 2012)

[5] Sleator D., Temperley D. Parsing English with a Link Grammar. Pittsburgh: School of Computer Science Carnegie Mellon University, 1991. – 93 p.

[6] Murzin F., Perfliev A., Shmanina T. Methods of syntactic analysis and comparison of constructions of a natural language oriented to use in search systems // Bull. Nov. Comp. Center, Comp. Science, 2010, Iss. 31, – P. 91-109.

[7] Murzin F., Perfliev A., Shmanina T. Methods of syntactic analysis and comparison of constructions of a natural language oriented to use in search systems // Vestnik of Novosibirsk State Univ. Ser.:Information Technologies. – Novosibirsk, 2012. –Vol. 9, Is. 4. – P. 13-28. (in Russian)

[8] Lbov G.S. Methods of processing of polytypic experimental data // The monograph / Sobolev Institute of Mathematics SB RAS . – Novosibirsk: Nauka, 1981. – 160 p. (in Russian)

[9] Vikentiev A.A., Vikentiev R.A. On the metrics for formulas containing polytypic variables and measures of denyty // Proc.of the Second International Youth School-Conference «Theory and numerical methods of the decision of inverse and incorrect problems». 2011. Part 1. – P. 192-209. [Electronic resource]. – 1998. – Mode of access: http://semr.math.nsc.ru/v8/c182-410.pdf (accessed 18 August 2014) (in Russian)

ТАБИҒИ ТІЛДЕГІ СӨЙЛЕМДЕРДІҢ ЖАҚЫНДЫҚ ДӘРЕЖЕСІН АНЫҚТАУ МАШИНАЛЫҚ-БАҒДАРЛАНҒАН ӘДІСІ

Т.В. Батура[1], Ф.А. Мурзин[1], А.А. Перфильев[1], Б.С. Байжанов[2], М.В. Немченко[2]

Тірек сөздер: ақпаратты-іздестіру жүйесі, Link Grammar Parser, синтаксистік талдау, семантика,релеванттық.

Аннотация: Негізгі қарастырылып отырған мәселе , мәтіннің құрылысына еніп, барлау сұранысындағы мәтіннің релевантына адекватты бағасын шығару алгоритмдерін құрастыру болып табылады. Берілген баға барлау сұранысына негізделген болуы және тек тірек сөздермен, олардың жақын алысына ғана шектелмеуі өте маңызды. Авторлар Link Grammar Parser программалық жүйенің шығу кезінде пайда болатын сөйлем сөздері арасындағы семантика-синтаксистік қатынастарды қолдануын ұсынды. Мақалада сөйлемдердің табиғи тілге жақындық дәрежесі екі есептеу алгоритмімен суреттелді. Олардың екіншісі математикалық логикаға сүйеніп жасалынды.Бұл әдістер ішінара iNetSearch ақпаратты-іздестіру жүйесінде қолданылды.

Поступила 09.09.2014 г.