УДК 004.318

ИТЕРАТИВНЫЙ ПОДХОД К АНАЛИЗУ ЕСТЕСТВЕННО-ЯЗЫКОВЫХ ТЕКСТОВ: ЛОГИЧЕСКИЙ АСПЕКТ

С.Л. Крывый, Д.С. Бибиков

Киевский национальный университет имени Тараса Шевченко, тел.: (044) 522 3433, email:krivoi@i.com.ua
Институт кибернетики им. В.М. Глушкова НАН Украины, email: bb_coff@mail.ru

Описывается логический подход к анализу естественно языкового текста с целью извлечения знаний. В частности рассматривается использование линейной темпоральной логики для анализа и представления модальностей в таком тексте.

A logical approach to analysis of natural language text for extraction knowledges from this text is described. In special case consider using of linear temporal logic for representations of modalities in such text.

Введение

Проблемы, связанные с анализом естественно-языковых объектов (ЕЯО) традиционно относят к области искусственного интеллекта. Однако, объективные трудности, возникающие на пути анализа ЕЯО, не позволяют удовлетворительно решать проблему автоматизации такого анализа. Эти трудности связаны с тем, что проблема анализа ЕЯО относится к областям, которые плохо поддаются формализации.

Отметим некоторые из трудностей, связанных с таким анализом. Как формализовать и смоделировать

- эмоциональную окраску естественно-языкового текста (ЕЯТ);
- знания, имеющиеся в предложениях иронического характера;
- знания, имеющиеся в предложениях иносказательного характера;
- правдоподобные знания и достоверные знания;
- понятие здравого смысла и здравого рассуждения и т. д. и т. п.

Одни исследователи этих проблем считают, что моделирование знаний, извлеченных из ЕЯТ, не может ограничиваться формализацией только лишь непогрешимого интеллекта. Естественным основанием для такого мнения является то, что важной чертой естественного интеллекта есть способность вырабатывать здравые суждения, которые могут оказаться и недостоверными. В случае неполной, неточной и изменчивой информации наши суждения часто становятся только предположительными, а в следствии этого лишь правдоподобными и поэтому могут подлежать пересмотру и уточнению (модификации). Примером такого типа рассуждений является диагностика. Здесь, необходима формальная теория модифицируемых рассуждений.

Другие исследователи считают, что проблема анализа ЕЯТ решена, если извлеченные знания представлены в базе знаний и все проблемы по их анализу решаются средствами баз знаний. Это мнение, с нашей точки зрения, не совсем соответствует реальному состоянию дел. Главная проблема при обработке знаний в базах знаний – это та же модифицируемость знаний. Модификация знаний необходима по многим причинам. Различают два фундаментальных типа модифицируемых знаний: предположительные и индексируемые.

Предположительные знания являются всего лишь правдоподобными. Это связано с тем, что они неточные, поскольку базируются на неполной, неточной и изменчивой информации, а также по причине их естественной неточности и модифицируемости. Примерами такого типа знаний являются рассуждения по умолчанию, рассуждения с прототипами и знания статистического характера.

Индексируемые знания – это рассуждения, которые основаны на знаниях, предполагаемых полными, но которые таковыми не являются или перестают быть таковыми. В действительности часто встречается ситуация, когда выдвигаются модифицируемые (а иногда и неявные) соглашения, для наращивания наших знаний в условиях неполной или неизвестной информации. Основываясь на таких знаниях наши выводы могут быть логически корректными по отношению к этим добавленным знаниям. Однако, эти рассуждения оказываются модифицируемыми, так как они основываются на изменчивом состоянии знаний. Следует отметить, что некоторые корректные формы вывода, которые формализует классическая математическая логика, могут оказаться модифицируемыми. Это объясняется тем, что они применяются к базе знаний, которая зачастую пополняется всего лишь правдоподобными знаниями. Например, знания, занесенные одним исследователем в базу знаний, могут быть неправильно или неточно поняты и поэтому будут подвергаться модификации другим исследователем.

По проблеме анализа ЕЯО существует огромная литература, в которой описываются различные методы и подходы к решению частных случаев упомянутых проблем [1 – 6]. В частности, в работах [5 – 7] описывались способы извлечения первичных знаний из ЕЯТ, ограничиваясь окрестностью одного предложения (именно в этом смысле и употреблялось слово «первичных»).

В данной работе рассматривается задача семантического анализа предложений естественного языка с целью извлечения знаний, не ограничиваясь рамками одного предложения и с учетом модальностей. Первичной информацией при таком подходе выступает информация, которую дает система частотного анализа слов в ЕЯТ

[6]. На первом шаге итерации все глаголы и предикатные слова объявляются отношениями, на втором шаге итерации уточняется арность этих отношений и выполняется их семантический анализ в пределах одного предложения, на третьем шаге выполняется уточнение семантического смысла полученной информации исходя из некоторой локальной части текста или из всего текста в целом. Данный подход можно выполнять как в автоматизированном режиме, так и в диалоговом. Извлеченные из текста отношения трактуются как знания, в виде формул логики предикатов первого порядка, а модальности – в виде формул линейной темпоральной логики. Потом полученные формулы линейной темпоральной логики транслируются в формулы логики первого порядка. Отношения, соответствующие извлеченным предикатам, или сами предикаты заносятся в базу знаний, с помощью которой выполняется дальнейший логический и информационный анализ (схема такого анализа показана на рис. 1).

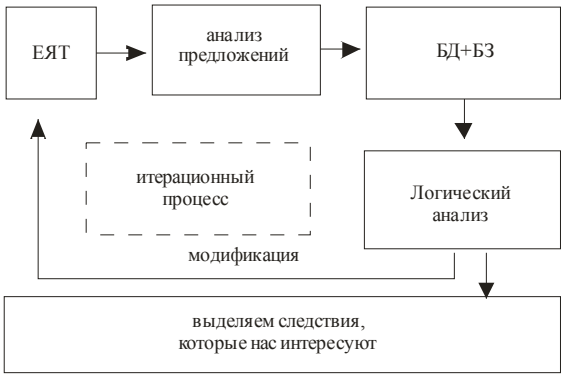


Рис. 1. Схема анализа ЕЯТ

Анализ такого типа существенно облегчается, если ЕЯТ является структурированным. Ведущей парадигмой структурирования информации на сегодняшний день являются онтологии или иерархические концептуальные структуры, представляющие собой модель предметной области, состоящей из иерархии понятий (концептов) предметной области, связей между ними и законов, действующих в рамках этой модели. Поэтому данный анализ ЕЯТ может служить одной из ступенек построения онтологий.

Формальная постановка задачи

Процесс автоматизации какой-либо деятельности, как правило, требует формализованной постановки задачи, которая дает возможность выполнения анализа данной задачи с целью выработки метода ее решения. Когда речь идет об автоматизации процесса извлечения знаний из ЕЯТ и построения соответствующей онтологии, то необходимо определить понятия «знание» и «извлечение знания». С целью формализации вышеуказанных понятий, введем следующие определения, пользуясь нотацией констрейнтного программирования [3].

Пусть дано некоторое множество D, на котором определена конечная совокупность $R = \{R_1, \ldots, R_k\}$ отношений $R_i \subseteq D^n$, $i = 1, 2, \ldots, k$, конечной арности. Языком ограничений L на D называется непустое множество $L \subseteq R$. Проблема выполнимости ограничений из L формулируется следующим образом.

Для произвольного множества D и языка ограничений L на D проблемой выполнимости ограничений $CSP(L)$ является решение такой комбинаторной задачи:

дана тройка $P = (V, D, C)$, где

- $V = \{v_1, \ldots, v_m\}$ – конечное множество переменных;

- $C = \{c_1, \ldots, c_q\}$ – конечное множество ограничений, где ограничение c_i из C – пара (s_i, R_i), где $s_i = (v_{i1}, ..., v_{ij})$ – кортеж, состоящий из переменных, $R_i \in L - n_j$ -арное отношение на D;

найти функцию $\varphi : V \to D$ такую, что $\forall (s_i, R_i) \in C$ кортеж $(\varphi(v_{i1}), \ldots, \varphi(v_{ij})) \in R_i$ либо убедится в том, что её не существует, $i = 1, 2, \ldots, k$. Множество D в этом случае называется областью проблемы, а функция φ называется интерпретацией $CSP(L)$.

В случае анализа ЕЯТ с целью извлечения знаний множество D интерпретируется как множество объектов, извлечённых из входного текста T, которое факторизовано по некоторому отношению эквивалентности R (это отношение будем называть отношением синонимии) и в котором «закодированы» отношения R_i, $i = 1, 2, \ldots, k$. Переменные из множества $V = \{v_1, v_2, \ldots, v_m\}$ принимают свои значения в этом факторизованном множестве объектов, фигурирующих в тексте T (это могут быть лексико-грамматические разряды, конкретные объекты (люди, даты, предметы и т. п.)).

Проблемой извлечения знаний из ЕЯТ называется проблема поиска интерпретации $\varphi : V \to D$ с явным построением отношений R_i совокупности $L \subseteq R$. При этом, отношения $R_i \in L,\ i = 1,2,...,k,$ извлеченные из текста *T,* будем называть *знаниями.*

Приведенное определение достаточно общее и его необходимо уточнять для настройки на конкретную область применения. Конкретизация интерпретации отношений в таком случае определяется целями, которые преследуются при анализе данного текста *T*. Хорошо известными примерами такого уточнения являются следующие примеры.

1. Лексико-грамматический анализ приводит к конкретизации интерпретации $\varphi : V \to T$ и отношений $R_i \in L$. Интерпретация φ в данном случае представляется в виде суперпозиции двух функций φ_1 и φ_2, т. е. $\varphi(V) = \varphi_2(\varphi_1(V)) = \varphi_1 * \varphi_2(V)$, где $*$ означает суперпозицию функций. Функции φ_1 и φ_2 реализуют процесс синтаксического и семантического анализа предложений текста *T*, а отношения R_1 и R_2 – это синтаксические ограничения (синтаксические правила языка, в котором представлен текст *T*) и семантические ограничения. Функцию φ_1 тоже можно рассматривать как суперпозицию отображений φ_{11} и φ_{12} , которые реализуют соответственно морфологический и синтаксический анализ предложений ЕЯТ T и которые вместе с отображением φ_2 составляют классическую систему лексико-грамматического анализа [4].

2. Силлогистика Аристотеля – другой пример уточнения интерпретации φ и отношений $R_i \in L$. В этом случае интерпретация φ носит теоретико-множественный характер, а отношения $R_i \in L$ – это отношение включения для множеств и его свойства. Более полное описание этого уточнения можно найти в [5, 7, 8].

3. Текст библиографического характера – пример хорошо структурированного текста. Это значит, что проблему извлечения знаний из такого текста можно решить в автоматизированном режиме.

Пример автоматизированной обработки ЕЯТ

Рассмотрим пример автоматизированной обработки некоторого фрагмента ЕЯТ *T*. Подсистема частотного анализа использует толковый словарь S естественного языка $L(X)$ (это может быть словарь русского, украинского, английского или какого-либо другого естественного языка).

Текст T состоит из предложений языка $L(X)$, не содержащий никаких символов, кроме символов алфавита X (т. е. T не содержит формул, графиков, рисунков и т. п.). Эта подсистема реализует отношение γ , которое является суперпозицией двух отношений $\gamma_1 * \gamma_2$, выполняемых последовательно. Содержательно отношение γ_1 означает распознавание принадлежности слова к данному языку и проверку правильности написания слова $t_{i_j} \in t_i$, где $t_i \in T$, в соответствии с написанием его в толковом словаре, т. е.

$$\gamma_1(t_{i_j}) = \begin{cases} 1, & \text{если } t_{i_j} \in S; \\ 0, & \text{если } t_{i_j} \notin S. \end{cases}$$

Если слово $t_{i_j} \in t_i$ распознано в словаре *S*, то оно заносится в словарь T' правильных слов, а если это не так, то предусматривается сигнализация о том, что данное слово отсутствует в словаре *S* и принимается решение о добавлении данного слова в словарь или его исправлении (слово может быть искажено, например, в следствии сканирования текста *T*).

Словари S и T' – входные данные для отношения γ_2 . Содержательный смысл отношения γ_2 сводится к тому, что если $\gamma_1(t_{i_j}) = 1$, то $\gamma_2(t_{i_j})$ определяет его грамматическую единицу языка (имя собственное, сказуемое, существительное, числительное и т. п.), а также возможные флексии слова $t_{i_j} \in t_i$. Областью интерпретации текста T является модель $A = (D, \Pi)$, где T – это исходный текст, возможно расширенный некоторой дополнительной информацией, а сигнатура предикатов (отношений) Π определяется из текста T в результате использования информации о различных вхождениях слова t_{i_j} в предложения $t_i \in T$. При этом вычисление отношения φ ограничивается отдельно взятым предложением $t_i \in T$, определяемым каждым вхождением слова t_{i_j} в текст *T*. В случае трудности определения предиката $\pi_i \in \Pi$, предусматривается диалоговый режим вычисления $\varphi(\pi_i)$ и $\gamma(\varphi(\pi_i))$. Предлагаемая система анализа на рис. 2 выглядит следующим образом:

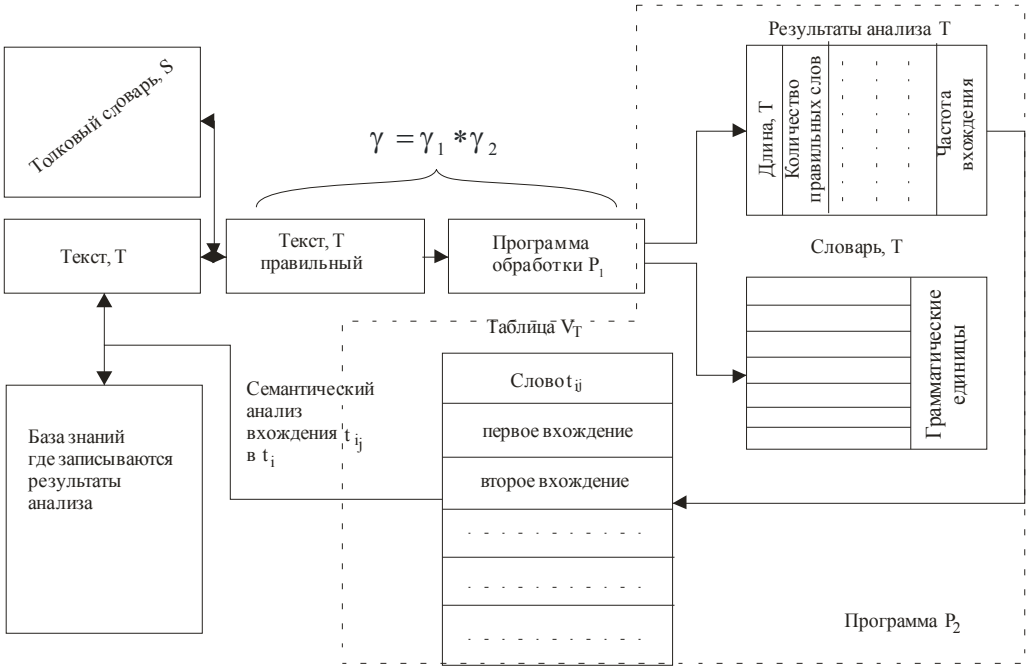
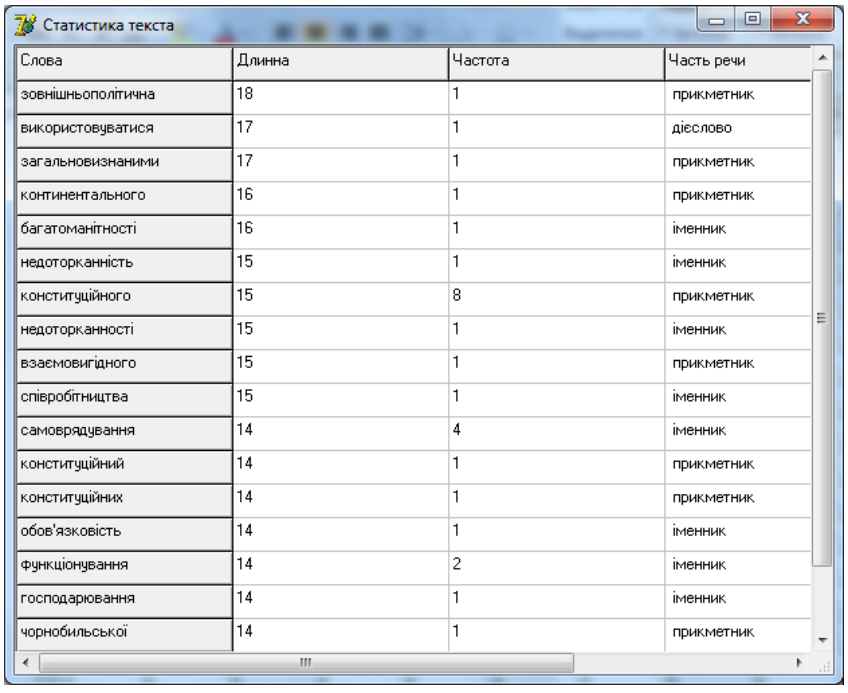


Рис. 2. Схема системы анализа предложений естественного языка

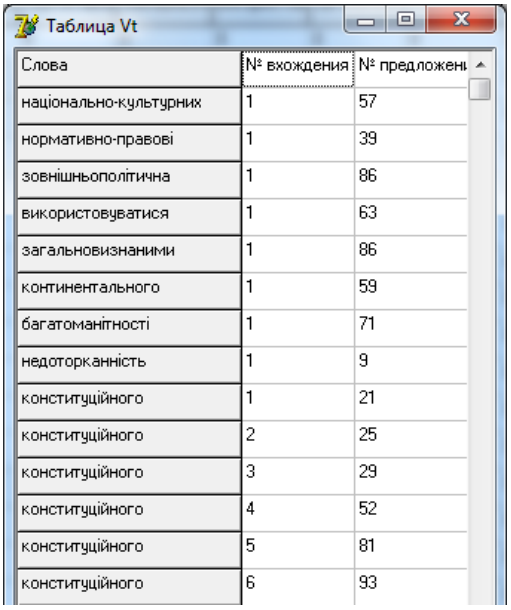
В этой схеме отношение $\gamma = \gamma_1 * \gamma_2$ вычисляет программа P_1. Результатом ее работы является два файла F_1 и F_2, заполненные соответственно числовыми характеристиками слов входного текста T и словами t_{i_j} предложений этого текста. Структура файла F_2 показана на рис. 3.


Статистика текста

Слова	Длинна	Частота	Часть речи
зовнішньополітична	18	1	прикметник
використовуватися	17	1	дієслово
загальновизнаними	17	1	прикметник
континентального	16	1	прикметник
багатоманітності	16	1	іменник
недоторканність	15	1	іменник
конституційного	15	8	прикметник
недоторканності	15	1	іменник
взаємовигідного	15	1	прикметник
співробітництва	15	1	іменник
самоврядування	14	4	іменник
конституційний	14	1	прикметник
конституційних	14	1	прикметник
обов'язковість	14	1	іменник
функціонування	14	2	іменник
господарювання	14	1	іменник
чорнобильської	14	1	прикметник

Рис. 3. Структура файла F_2

Файлы F_1 и F_2, сформированные программой P_1, служат входными данными для работы программы P_2, которая вычисляет отношение φ. При этом, работа программы P_2 сводится к построению таблицы V_T для слов $t_{i_j} \in t_i$, $t_i \in T$. Затем, по этой таблице и предложениям текста T определяется семантический смысл рассматриваемого предложения. Предложение $t_i \in T$ определяется на основе номера вхождения слова t_{i_j} в текст T с помощью таблицы V_T, вид которой показан на рис. 4.



Слова	№ вхождения	№ предложени
національно-культурних	1	57
нормативно-правові	1	39
зовнішньополітична	1	86
використовуватися	1	63
загальновизнаними	1	86
континентального	1	59
багатоманітності	1	71
недоторканність	1	9
конституційного	1	21
конституційного	2	25
конституційного	3	29
конституційного	4	52
конституційного	5	81
конституційного	6	93

Рис. 4. Структура таблицы V_T

Итеративный подход к анализу ЕЯТ: логический аспект

В общем случае неструктурированного текста семантический анализ затруднителен и возникает вопрос: каким ограничениям должен удовлетворять исходный текст, чтобы можно было каким либо образом анализировать ЕЯТ. Ответ на этот вопрос усложняется ещё и тем, что в ЕЯТ возможны временные зависимости и модальности. Примером такого типа предложений могут служить предложения «Пока брат не извинится, я с ним не буду разговаривать» и «Я всегда любил играть в крокет и до вчерашнего дня я любил пить после игры кофе, а с сегодняшнего дня я пью только чай». Учитывая эту ситуацию предлагается вести анализ ЕЯТ итеративным методом, используя толковый словарь, словарь, извлечённый из текста, места вхождения слов в ЕЯТ и частотный анализ слов. Поясним итеративный подход к анализу ЕЯТ с помощью примера. С этой целью рассмотрим фрагмент некоторого текста из книги Эрленд Лу «НАИВНО. СУПЕР».

«Я всегда любил играть в крокет со своим братом. Но после ссоры с братом, я заявил, что пока брат не извинится, я с ним не буду разговаривать… Вчера брат пришёл с извинениями и я его простил. На следующий день мы с братом стали играть в крокет. Это бывает нечасто. Принадлежности для крокета, брошенные под сараем, совсем сгнили. Мы объездили несколько заправок в поисках нового комплекта. Брат расплатился по одной из своих кредитных карточек. Потом мы шагами отмерили в родительском саду расстояния, вбили в землю воротца и колышки. Я выбрал красный цвет, мой брат – желтый.

Мы начали играть, и некоторое время все шло прекрасно. Я быстро прошел первые и вторые воротца. Заработал дополнительный ход и продолжил игру …»

На первом шаге итерации, как было сказано ранее, все глаголы и предикатные слова объявляются отношениями. Например, в предложении «Вечером мы с братом стали играть в крокет», как глагол, первым выделяется словосочетание «стали играть», которое в диалоговом режиме уточняется со специалистом. В результате чего мы получаем в черновом варианте предикат СТАЛИ_ИГРАТЬ (я, брат, крокет). Далее, из предложения «Мы объездили несколько заправок в поисках нового комплекта» получаем ИСКАТЬ (я, брат, комплект, заправок), а так же ОТМЕРИЛИ (я, брат, сад, расстояния), ВБИЛИ (я, брат, воротца) и ВБИЛИ (я, брат, колышки) и т.д.

При анализе сложных (сложно подчинённых) предложений они автоматически разбиваются на простые, причем существительное которое присутствует в одной из составляющих частей, дублируется и на вторую часть предложения. Таким образом, вместо одного предложения «Потом мы шагами отмерили в родительском саду расстояния, вбили в землю воротца и колышки.», мы получаем два простых: «Шагами отмерили в родительском саду расстояния», «мы вбили в землю воротца» и «мы вбили в землю колышки» которыми оперирует программа.

Разбиение сложного предложения на простые никак не влияет на логическую связанность текста, потому как первый этап препроцессинга подразумевает построения статистической таблицы V_T аналогично той, что показана на рис. 4, в которой четко фиксируется какому предложению принадлежат слова.

На втором шаге выполняется определение арности и окончательное формирование предикатов. Как результат получаем следующие предикаты: ИСКАТЬ (я, брат, комплект) арность 3, а так же МЕРЯТЬ (я, брат, расстояния) арность 3, ВБИВАТЬ (я, брат, воротца), ВБИВАТЬ (я, брат, колышки) – арности обоих отношений 3. Если арность больше 2, тогда имеет место замена предиката на конъюнкцию двух новых: ВБИВАТЬ (я, воротца) $\wedge$ ВБИВАТЬ (я, колышки) $\wedge$ ВБИВАТЬ (брат, воротца) $\wedge$ ВБИВАТЬ (брат, колышки). В

приведенном случае семантическая обработка дает нам эквивалентную замену «мы» на смысловую конъюнкцию «я» $\wedge$ «брат», и тогда вместо полученных ранее трех предикатов, получаем две конъюнкции: ИСКАТЬ (я, комплект) $\wedge$ ИСКАТЬ (брат, комплект), МЕРЯТЬ (я, расстояния) $\wedge$ МЕРЯТЬ (брат, расстояния) и т. д.

Таким образом, получаются такие предикаты:

- ЛЮБИТЬ_ИГРАТЬ_КРОКЕТ (я, брат);
- ССОРА (я, брат);
- РАЗГОВАРИВАТЬ (я, брат);
- ИЗВИНЯТЬСЯ (брат, я);
- ПРОСТИТЬ (я, брат);
- СТАЛИ_ИГРАТЬ_КРОКЕТ (я, брат);
- и т. д.

Что касается модальностей и временных зависимостей, имеющихся в тексте, то на помощь приходит модальная временная логика. Начало вышеприведенного текста принимает такую форму в этой логике:

F^- (ЛЮБИТЬ_ИГРАТЬ_КРОКЕТ (я, брат)) – «Я всегда любил играть в крокет со своим братом».

O^- (ССОРА(я, брат)) – «после ссоры с братом»,

(ИЗВИНЯТЬСЯ (брат, я)) $\cup$ ($\neg$ РАЗГОВАРИВАТЬ (я, брат)) – «пока брат не извинится, я с ним не буду разговаривать…»

O^- (ИЗВИНЯТЬСЯ (брат, я)) $\wedge$ O^- (ПРОСТИТЬ (я, брат)) – «вчера брат пришёл с извинениями и я его простил».

O (СТАЛИ_ИГРАТЬ_КРОКЕТ (я, брат)) – «На следующий день мы с братом стали играть в крокет».

Тогда зависимости между предложениями выглядят таким образом:

(F^- (ЛЮБИТЬ_ИГРАТЬ_КРОКЕТ (я, брат)) $\wedge$ O^- (ССОРА (я, брат)) $\rightarrow$

((ИЗВИНЯТЬСЯ (брат, я)) $\cup$ ($\neg$ РАЗГОВАРИВАТЬ (я, брат)));

(O^- (ИЗВИНЯТЬСЯ (брат, я)) $\wedge$ O^- (ПРОСТИТЬ (я, брат)) $\rightarrow$

O (СТАЛИ_ИГРАТЬ_КРОКЕТ (я, брат)).

Если в процессе такого анализа возникают трудности с определением смысла того или иного предложения, то уточнение семантического смысла выполняется в диалоговом режиме при появлении семантической неоднозначности. Подобного рода ситуация может появиться, например, в таком предложении: «Проезжая по набережной Черного моря, он сравнил облака с танцующими сатирами. Но когда он присмотрелся, то понял, что на самом деле эти небесные сатиры не плясали а плыли». Из первого предложения получаем, что существительное «облака» логично эквивалентно понятию «сатиры», а из второго предложения находим, что сатиры на небе не танцуют, а плывут. Что же на самом деле имеет место в этом предложении – «облака пляшут или плывут»? В таком случае специалисту на выбор предлагаются варианты семантических значений слова. Полученные таким образом формулы и отношения транслируются в формулы логики предикатов и по результатам этой трансляции строится база знаний.

Трансляция формул линейной темпоральной логики в формулы языка предикатов первого порядка

Для описания процесса трансляции формул линейной темпоральной логики (LTL) в формулы логики предикатов первого порядка, приведем определение синтаксиса и семантики формул LTL.

Синтаксис LTL *(linear temporal logic).* Пусть $P = \{p, q, r, ...\}$ – множество пропозициональных атомарных высказываний. Тогда синтаксис LTL строится по таким правилам:

$$LTL := P \mid 0 \mid LTL \rightarrow LTL \mid LTL \cup^+ LTL \mid LTL \cup^- LTL,$$

где 0 – логическая константа false; $\rightarrow$ – импликация; $\cup^+$ и $\cup^-$ – модальные until-операторы настоящего и будущего, а также для прошедшего соответственно. Остальные связки вводятся с помощью таких эквивалентностей:

$$\neg\varphi = \varphi \rightarrow 0,\ 1 = \neg 0 = 0 \rightarrow 0,\ \varphi \vee \psi = \neg\varphi \rightarrow \psi,$$
$$\varphi \wedge \psi = \neg(\neg\varphi \vee \neg\psi) = \neg(\varphi \rightarrow \neg\psi),$$
$$\varphi \Leftrightarrow \psi = (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi).$$

Модальные операторы G, F, F^-, G^- а также O (next) вводятся с помощью until-операторов $\cup^+$ и $\cup^-$ следующим образом:

$$O\varphi = (0 \cup^+ \varphi),\ F\varphi = (1 \cup^+ \varphi),\ G\varphi = \neg F\neg\varphi = \neg(1 \cup^+ \neg\varphi),$$
$$F^-\phi \Leftrightarrow (1 \cup^- \phi),\ \ G^-\phi \Leftrightarrow \neg F^-\neg\phi.$$

Семантика LTL-*формул.* Семантика формул LTL описывается с помощью модели Крипке, которая для LTL имеет вид $M = (T, R, I, t_0)$, где T – множество моментов времени, $t_0 \in T$ – начальный момент времени, а I – интерпретация, сопоставляющая отношению R бинарное отношение достижимости – отношение линейного

порядка $I(R) = \leq \subseteq T \times T$, а каждому $p \in P$ – подмножество тех моментов времени, в которые p -истинно. Семантика LTL – формул определяется на модели *M* таким способом:

$$M \models p \Leftrightarrow t_0 \in I(p), p \in P,$$

$$M \not\models 0,$$

$$M \models (\varphi \to \psi) \Leftrightarrow \text{из } M \models \varphi \; M \models \psi,$$

$$M \models (\varphi \cup^{+} \psi) \Leftrightarrow \begin{array}{l}(\exists t_1 \in T)(t_0 < t_1 \,\&\, t_1 \models \psi)\,\& \\ (\forall t_2 \in T)(t_0 < t_2 < t_1 \to t_2 \models \varphi)\end{array}$$

$$M \models (\varphi \cup^{-} \psi) \Leftrightarrow \begin{array}{l}(\exists t_1 \in T)(t_1 < t_0 \,\&\, t_1 \models \psi)\,\& \\ (\forall t_2 \in T)(t_1 < t_2 < t_0 \to t_2 \models \varphi)\end{array}$$

Известно, что формулы линейной темпоральной логики транслируются в язык *FOL* с одной свободной переменной [9]. Правила трансляции получаются из определения семантики LTL-формул, которая приведена выше. В результате получается некоторый фрагмент FOL, называемый языком монадической логики первого порядка и обозначаемый *mFOL*.

Исходя из определения семантики LTL – формул, определяются правила трансляции в язык *FOL*, а точнее в язык монадической логики первого порядка (*mFOL*):

$$FOL(p) = p(t_0), \text{ где } p \in P, t_0 \in T,$$

$$FOL(0) = (t_0 \neq t_0) = \neg(t_0 = t_0),$$

$$FOL(\varphi \to \psi) = FOL(\varphi) \to FOL(\psi)$$

$$FOL(\varphi \cup^{+} \psi) = (\exists t_1 \in T)(t_0 < t_1 \,\&\, FOL(\psi)\{t_0 := t_1\})\,\& \\ (\forall t_2 \in T)(t_0 < t_2 < t_1 \to FOL(\varphi)\{t_0 := t_2\}),$$

$$FOL(\varphi \cup^{-} \psi) = (\exists t_1 \in T)(t_1 < t_0 \,\&\, FOL(\psi)\{t_0 := t_1\})\,\& \\ (\forall t_2 \in T)(t_1 < t_2 < t_0 \to FOL(\varphi)\{t_0 := t_2\}),$$

где переменные t_1 и t_2 – произвольные переменные, которые не входят в формулы $FOL(\varphi)$ и $FOL(\psi)$, а *FOL* означает (программу) алгоритм трансляции.

Такую трансляцию иногда называют стандартной трансляцией. В процессе трансляции формул $\varphi \cup^{+} \psi$, $\varphi \cup^{-} \psi$ и $\circ\varphi$ символами t' и t'' обозначают произвольные переменные, которые не входят в $FOL(\varphi)$ или $FOL(\psi)$. Формула $FOL(\varphi)\{t_0 := t'\}$ означает формулу $FOL(\varphi)$, где каждое свободное вхождение переменной t_0 замещается переменой t' .

Например, предложение «До тех пор, пока нет запроса на доступ к общей памяти от некоторого процесса вычислений, до тех пор не будет разрешения на доступ к общей памяти». Это предложение переводится в LTL-формулу:

$$(\neg(ent-c-m) \cup^{-} (req-c-m)) \cup^{+} (ent-c-m),$$

где $ent-c-m$ и $req-c-m$ обозначают формулы «есть доступ к общей памяти» и «есть запрос к общей памяти», соответственно. Тогда трансляция этой формулы имеет вид:

$$FOL((\neg(ent-c-m) \cup^{-} (req-c-m)) \cup^{+} (ent-c-m)) =$$

$$= \exists t_1(t_0 < t_1 \wedge ent-c-m(t_1) \wedge \forall t_2(t_0 < t_2 < t_1 \to FOL((\neg ent-c-m \cup^{-} req-c-m))))\{t_0 := t_2\} =$$

$$= \exists t_1(t_0 < t_1 \wedge req-c-m(t1) \wedge \forall t_2(t_0 < t_2 < t_1) \to \exists t_3(t_3 < t_2\ req-c-m(t_3) \wedge$$

$$\wedge \forall t_4(t_3 < t_4 < t_2 \to \neg ent-c-m(t_4)))).$$

Как видно из работы алгоритма *FOL*, язык *mFOL* должен содержать два предиката: $\Pi_{<} =<$ и $\Pi_{\neq} = \neq$. Свойства этих предикатов следующие:

а) свойства предиката $\Pi_{<}$:

а1) $\Pi_{<}(t,t') \,\&\, \Pi_{<}(t',t'') \to \Pi_{<}(t,t'')$ (транзитивность)

а2) $\Pi_{<}(t,t) \to 0$ (иррефлексивность)

б) свойства предиката $\Pi_{\neq}$:

б1) $\Pi_{\neq}(t,t') \Leftrightarrow \Pi_{\neq}(t',t)$ (симметричность)

б2) $\Pi_{\neq}(t,t) \to 0$ (иррефлексивность)

Учитывая предикаты $\Pi_{<}$ и $\Pi_{\neq}$, транслятор FOL принимает вид:

$$FOL(P) = P(t_0)$$

$$FOL(0) = \Pi_{\neq}(t_0, t_0),$$

$$FOL(\varphi \to \psi) = FOL(\varphi) \to FOL(\psi)$$

$$FOL(\varphi \cup^{+} \psi) = (\exists t_1)(\Pi_{<}(t_0, t_1) \,\&\, FOL(\psi)\{t_0 := t_1\})\,\&$$

$$\&(\forall t_2)(\Pi_<(t_0,t_2)\,\&\,\Pi_<(t_2,t_1)\to FOL(\varphi)\{t_0 := t_2\}),$$

$$FOL(\varphi \cup^- \psi) = (\exists t_1)(\Pi_<(t_1,t_0)\,\&\,FOL(\psi)\{t_0 := t_1\})\,\&$$

$$\&(\forall t_2)(\Pi_<(t_1,t_2)\,\&\,\Pi_<(t_2,t_0)\to FOL(\varphi)\{t_0 := t_2\}),$$

где t_1 и t_2 не являются переменными формул $FOL(\varphi)$ и $FOL(\psi)$.

Применим теоретические выводы к конкретным примерам:

а) «Я всегда любил играть в крокет со своим братом». Это выражение имеет следующее представление:

$$F^-(ЛЮБИТЬ\_ИГРАТЬ\_КРОКЕТ\ (я, брат)) \Rightarrow$$

$$\Rightarrow \neg\left(1 \cup^- \neg\left(ЛЮБИТЬ\_ИГРАТЬ\_КРОКЕТ\ (я, брат)\right)\right) \Rightarrow$$

$$\Rightarrow \exists t_1(\Pi_<(t_0,t_1) \wedge ЛЮБИТЬ\_ИГРАТЬ\_КРОКЕТ\ (я, брат)) \wedge \forall t_2(\Pi_<(t_0,t_2) \wedge \Pi_<(t_2,t_1) \to 1) \Rightarrow$$

$$\Rightarrow \exists t_1((t_0 < t_1) \wedge ЛЮБИТЬ\_ИГРАТЬ\_КРОКЕТ\ (я, брат)) \wedge \forall t_2((t_0 < t_2) \wedge (t_2 < t_1) \to 1);$$

б) «после ссоры с братом»:

$$O^-(ССОРА\ (я, брат)) \Rightarrow$$

$$\Rightarrow (\exists t_1 \Pi_<(t_1,t_0) \wedge (ССОРА\ (я, брат)) \wedge (\forall t_2\ \Pi_<(t_1,t_2) \wedge \Pi_<(t_2,t_0) \to ССОРА\ (я, брат)) \Rightarrow$$

$$\Rightarrow (\exists t_1(t_1 < t_0) \wedge (ССОРА\ (я, брат)) \wedge (\forall t_2\ (t_1 < t_2) \wedge (t_2 < t_0) \to ССОРА\ (я, брат));$$

в) «пока брат не извинится, я с ним не буду разговаривать…»:

$$(ИЗВИНЯТСЯ\ (брат, я)) \cup (\neg РАЗГОВАРИВАТЬ\ (я, брат)) \Rightarrow$$

$$\Rightarrow (\exists t_1\ \Pi_<(t_0,t_1)) \wedge (\neg РАЗГОВАРИВАТЬ\ (я, брат)(t_1)) \wedge (\forall t_2 \Pi_<(t_0,t_2) \wedge \Pi_<(t_2,t_1) \to$$

$$(ИЗВИНЯТЬСЯ\ (брат, я)(t_2)) \Rightarrow$$

$$\Rightarrow (\exists t_1(t_0 < t_1)) \wedge (\neg РАЗГОВАРИВАТЬ\ (я, брат)) \wedge (\forall t_2(t_0 < t_2) \wedge (t_2 < t_1) \to$$

$$(ИЗВИНЯТЬСЯ\ (брат, я));$$

г) «вчера брат пришел с извинениями и я его простил»:

$$O^-(ИЗВИНЯТЬСЯ\ (брат, я)) \wedge O^-(ПРОСТИТЬ\ (я, брат)) \Rightarrow$$

$$\Rightarrow 0 \cup^- (ИЗВИНЯТЬСЯ\ (брат, я)) \wedge 0 \cup^- (ПРОСТИТЬ\ (я, брат)) \Rightarrow$$

$$\Rightarrow (\exists t_1 \Pi_<(t_1,t_0) \wedge (ИЗВИНЯТЬСЯ\ (брат, я))) \wedge (\forall t_2\ \Pi_<(t_1,t_2) \wedge \Pi_<(t_2,t_0) \to ИЗВИНЯТЬСЯ\ (брат, я)) \wedge$$

$$\wedge (\exists t_1 \Pi_<(t_1,t_0) \wedge (ПРОСТИТЬ\ (я, брат))) \wedge (\forall t_2\ \Pi_<(t_1,t_2) \wedge \Pi_<(t_2,t_0) \to ПРОСТИТЬ\ (я, брат) \Rightarrow$$

$$\Rightarrow (\exists t_1(t_1 < t_0) \wedge (ИЗВИНЯТЬСЯ\ (брат, я))) \wedge (\forall t_2\ (t_1 < t_2) \wedge (t_2 < t_0) \to ИЗВИНЯТЬСЯ\ (брат, я)) \wedge$$

$$\wedge (\exists t_1(t_1 < t_0) \wedge (ПРОСТИТЬ\ (я, брат))) \wedge (\forall t_2\ (t_1 < t_2) \wedge (t_2 < t_0) \to ПРОСТИТЬ\ (я, брат))$$

д) «на следующий день мы с братом стали играть в крокет»

$$O(СТАЛИ\_ИГРАТЬ\_КРОКЕТ\ (я, брат)) \Rightarrow$$

$$\Rightarrow 0 \cup^+ СТАЛИ\_ИГРАТЬ\_КРОКЕТ\ (я, брат) \Rightarrow$$

$$\Rightarrow (\exists t_1\ \Pi_<(t_0,t_1) \wedge (СТАЛИ\_ИГРАТЬ\_КРОКЕТ\ (я, брат)) \wedge (\forall t_2\ \Pi_<(t_0,t_2) \wedge \Pi_<(t_2,t_1) \to 0) \Rightarrow$$

$$\Rightarrow (\exists t_1\ (t_0 < t_1) \wedge (СТАЛИ\_ИГРАТЬ\_КРОКЕТ\ (я, брат)) \wedge (\forall t_2\ (t_0 < t_2) \wedge (t_2 < t_1) \to 0).$$

Заключение

Описанные в данной работе способы автоматизации обработки ЕЯТ составляют основу как теоретического, так практического анализа извлечения знаний из ЕЯТ. Используя эту основу и прежде всего ее реализацию, предполагается наращивание ее мощности за счет построения новых мета отношений над построенными отношениями, являющимися отдельными частями знаний в исследуемом тексте.

1. *Палагин А.В., Крывый С.Л., Петренко Н.Г.*, Знание ориентированные информационные системы с обработкой естественно-языкових объектов: основы методологии и архитектурно-структурная организация. – УСиМ. – 2009. – № 3. – С. 42 – 55.
2. *Палагин А.В., Петренко Н.Г.* Системно-онтологический анализ предметной области. – УСиМ. – 2009. – № 4. – С. 3 – 14.
3. *Cohen D. Jeavons P.* The Complexity of Constraint Languages. In “Handbook of Constraint Programming. – Edited by F. Rossi, P. van Beek and T. Walsh. – 2006. – P. 245 – 280.
4. *Апресян Ю.Д.* Лингвистический процессор для сложных информационных систем. – М.: Наука, 1992. – 324 с.
5. *Палагін О.В., Кривий С.Л., Петренко М.Г., Бібіков Д.С.* Алгебро-логічний підхід до аналізу та обробки текстової інформації. // Проблеми программування. – 2010. – № 2–3. – С. 318 – 329.
6. *Палагин О.В., Крывый С.Л., Бибиков Д.С.* Обработка предложений естественного языка с использованием словарей и частоты появления слов. – Natural and Artificial Intelligence Intern Book Series. – Inteligent Processing. – ITHEA. – Sofia. – 2010. – N 9. – P. 44 – 52.
7. *Палагін О.В., Кривий С.Л., Петренко М.Г., Бібіков Д.С.* Формально-логічний підхід до побудови системи аналізу знань в різних предметних областях // Проблеми програмування. – 2010. – № 2 – 3. – С. 382 – 389.
8. *Кулик Б.А.* Логика естественных рассуждений. – С.-Петербург: Невский диалект, 2001. – 127 с.
9. *Clarke E.M., Schlingloff B.-H.* Model checking. In Handbook of Automated Reasoning. Eds. A. Robinson and A/ Voronkov. – Elsevier Science Publishers B.V. – 2001. – P. 1360 – 1522.