

УДК 004.5:004.657

А. Б. Кунгурцев, канд. техн. наук,
Я. В. Поточняк

ИНТЕРФЕЙС ДЛЯ ОБЩЕНИЯ ПОЛЬЗОВАТЕЛЕЙ С ИНФОРМАЦИОННЫМИ СИСТЕМАМИ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

Аннотация. Рассматриваются вопросы создания модели базы данных на естественном языке и анализ запроса пользователя для общения конечного пользователя с информационной системой на естественном языке. На основе анализа запроса производится поиск информации и выдается ответ, или указывается причина на естественном языке, вследствие которой не может быть сгенерирован ответ.

Ключевые слова: реляционная база данных, модель базы данных, информационная система, объект, атрибут, запрос пользователя, естественный язык, описание объекта, описание атрибута, синоним, информационная система, интерфейс пользователя

A. B. Kungurtsev, PhD.,
I. V. Potochniak

USER INTERFACE FOR USERS COMMUNICATION WITH INFORMATION SYSTEMS IN A NATURAL LANGUAGE

Abstract. The article considers questions of DB model creation in natural language and analysis of the user's requests in natural language for communication end-user with information system in natural language. Based on query parsing, being made information extraction, and return the answer or the reason, in natural language, in which the answer would not be generated.

Keywords: relational database, database model, information system, object, attribute, user request, natural language, description object, description attribute, synonym, information system, user interface

О. Б. Кунгурцев, канд. техн. наук,
Я. В. Поточняк

ИНТЕРФЕЙС ДЛЯ СПІЛКУВАННЯ КОРИСТУВАЧІВ З ІНФОРМАЦІЙНИМИ СИСТЕМАМИ ПРИРОДНОЮ МОВОЮ

Анотація. Розглядаються питання створення моделі бази даних природною мовою і аналіз запиту користувача для спілкування кінцевого користувача з інформаційною системою природною мовою. На основі аналізу запиту, проводиться пошук інформації, видається відповідь, або вказується причина природною мовою, внаслідок якої не може бути згенерована відповідь.

Ключові слова: реляційна база даних, модель бази даних, інформаційна система, об'єкт, атрибут, запит користувача, природна мова, опис об'єкта, опис атрибута, синонім, інформаційна система, інтерфейс користувача

Введение. Последнее десятилетие характеризуется широким распространением информационных систем (ИС) и быстрым ростом хранящейся в них информации. Параллельно непрерывно возрастает количество пользователей и желающих получить информацию, хранимую в ИС [1]. Поскольку реляционные базы данных (РБД) сохраняют свои доминирующие позиции как механизм хранения и обработки данных в ИС [1, 2], составление запросов на получение информации из ИС остаётся делом профессионалов, что существенно ограничивает количество пользователей ИС. Таким образом, проблема общения неквалифицированных пользователей с ИС стано-

вится всё более актуальной [3, 4]. Известны попытки решить её с помощью некоторого шаблона запроса, когда пользователю предлагается определенная последовательность шагов по уточнению поиска желаемой информации [5, 6], т.е. определенный шаблон поиска. Поскольку реляционные БД предусматривают огромное количество комбинаций имеющихся в них данных, то предусмотреть все пожелания пользователя невозможно, и шаблон предлагает ответы только для некоторых запросов, наиболее вероятных, по мнению его создателя. Кроме этого, изменения в структуре БД приводят к необходимости перепрограммирования шаблона.

Другое решение основано на выявлении в запросе пользователя на естественном языке некоторых ключевых слов и формирова-

© Кунгурцев А.Б., Поточняк Я.В., 2014

нии ответа на основе анализа их комбинации [7, 8]. В этом случае пользователь обычно получает множество вариантов ответов, которые в той или иной степени соответствуют набору ключевых слов. Здесь успешным считается тот вариант, в котором достаточно данных для дальнейшей самостоятельной работы пользователя.

Постановка задачи. Целью предлагаемой работы является сокращение времени на получение необходимой информации из РБД за счет обеспечения непосредственного доступа к РБД для неквалифицированных пользователей.

Для общения неквалифицированного пользователя с ИС предполагается решить следующие задачи.

1. Пользователь не знает специального языка запросов к ИС, поэтому ему следует предоставить возможность сформулировать запрос на естественном языке (ЕЯ) с минимальными ограничениями.

2. Пользователь не знает, какая именно информация содержится в ИС и как она структурирована, поэтому в случае запроса отсутствующих или должным образом не связанных данных, он должен получить исчерпывающий ответ о причине отказа выполнения запроса.

3. Если содержание запроса пользователя может быть автоматически преобразовано в

запрос к РБД [9], то он может получить ответ в понятной для него форме.

Следствием решением этих задач может быть экономический эффект, определяемый сокращением штата прикладных программистов, оперативное получение информации для принятия более обоснованных решений.

В данной работе предлагается создание определенного семантического образа информации в БД и в запросе пользователя на естественном языке с последующим анализом возможности построения ответа на этот запрос.

Принцип организации интерфейса пользователя

На рис. 1 представлена схема общения пользователя с ИС.

Запрос пользователя на ЕЯ поступает в анализатор запросов, который производит разбор предложения и выделяет в нем грамматические характеристики. Модель БД (МБД) имеет словесное описание структуры БД и позволяет определить наличие необходимой информации в БД для формирования ответа пользователя. Если запрос пользователя может быть выполнен, то генератор запросов формирует соответствующий SQL-запрос. Для каждого запроса (выполненного, либо содержащего ошибку) синтезатор ответа формирует ответ на ЕЯ, содержащий результат анализа запроса.

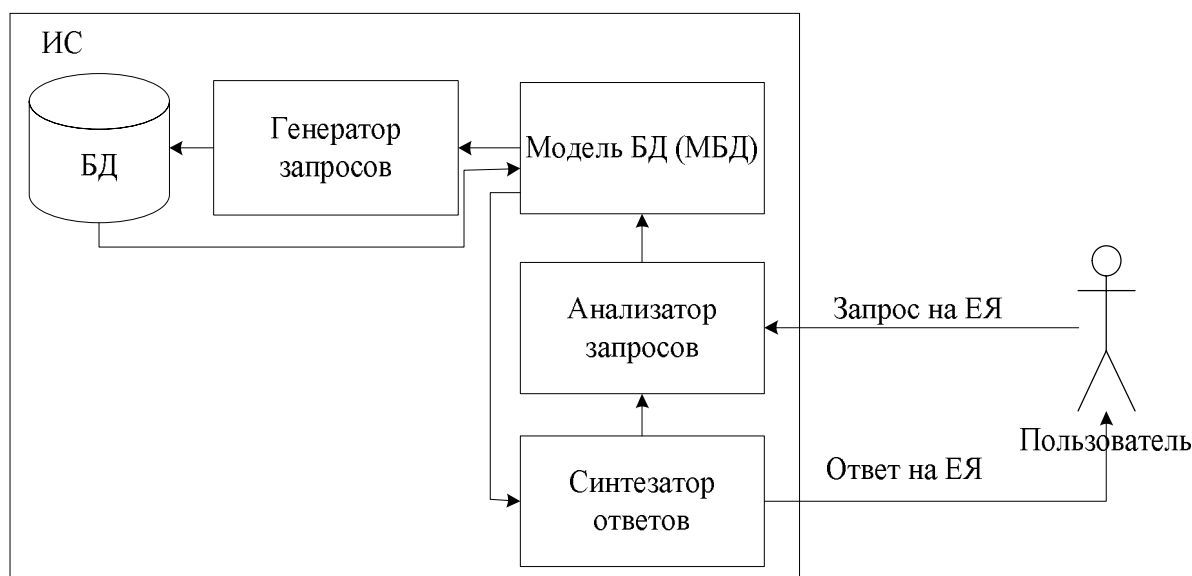


Рис. 1. Процесс взаимодействия пользователя с базой данных на естественном языке

Модель реляционной базы данных.

Для построения (МБД) структура БД представляется как множество таблиц

$$T = \{ \overline{T_i} \}, i = 1, nt, \quad (1)$$

где каждая таблица имеет следующий вид:

$$T_i = \{ \overline{Nt_i}, \{ \overline{C_{ij}} \} \}, j = 1, nc_i, \quad (2)$$

где Nt_i – название таблицы;

C_{ij} – колонка таблицы.

Каждую колонку БД представим в виде:

$$C_{ij} = \{ \overline{Nc_{ij}}, type \}, \quad (3)$$

где Nc_{ij} – название колонки таблицы;

$type$ – тип поля (тип данных колонки).

Предложено различать три типа полей: первичный ключ (ПК), вторичный ключ (ФК) и остальные типы данных.

Количество таблиц в реальной БД может достигать трехзначного числа. Названия таблиц, а также названия их полей часто не отражают суть хранящейся в БД информации. Поэтому в МБД введены описания таблиц и полей.

Предлагается представить МБД в виде множества объектов

$$M = \{ \overline{O_i} \}, i = 1, nt, \quad (4)$$

где каждый объект описывается кортежем:

$$O_i = \{ \overline{Nt_i}, \overline{TNo_i}, \{ \overline{A_{ij}} \}, \{ \overline{R_{il}} \} \}, \quad (5)$$

$$j = 1, na_i, l = 1, nr_i$$

где Nt_i – название таблицы; TNo_i – описание объекта; A_{ij} – атрибут объекта; R_{il} – ссылка из объекта O_i на объект O_j .

Каждый атрибут описывается кортежем:

$$A_{ij} = \{ \overline{Nc_{ij}}, \overline{TNa_{ij}} \}, j = 1, nc, \quad (6)$$

где Nc_{ij} – название колонки таблицы;

TNa_{ij} – описание атрибута.

Технология создания МБД предусматривает выполнение следующей последовательности действий.

1. Определить структуру РБД, как множество таблиц, каждая из которых содержит множество полей в соответствии с (1) и (2).

2. Для каждой таблицы создать тексты на ЕЯ, на основании которых будут формироваться описание таблицы – TNo_i , а также ее полей – TNa_{ij} , которые не являются ключами. Данный этап выполняется непосредственно экспертом (обычно это администратор БД).

венно экспертом (обычно это администратор БД).

3. Введенные тексты подвергнуть морфологическому и синтаксическому разбору, на основании которых формировать описания таблиц и их полей.

4. На основании внешних ключей РБД сформировать ссылки одного объекта модели на другой.

Предложено описания объекта представить в виде кортежа:

$$TNo_i = \{ \overline{term_o_i}, \overline{Syn_i}, \overline{prop_i} \}, i = 1, n, \quad (7)$$

где $term_o_i$ – термин объекта (слово или словосочетание, определяющее суть объекта);

Syn_i – синоним термина объекта, который можно представить как $Syn_i = \{ \overline{synonym} \}$ (слово или словосочетание, близкое по значению термину в рассматриваемой предметной области); $prop_i$ – некоторая характеристика объекта.

Описание атрибута предложено представить кортежем:

$$TNa_{ij} = \{ \overline{term_a_{ij}}, \overline{Syn_{ij}}, \overline{prop_{ij}}, \{ \overline{range_{ij}} \} \}$$

$$TNa_{ij} = \{ \overline{aggregation_{ij}}, \{ \overline{action_{ij}} \} \}, ij = 1, n, \quad (8)$$

где $term_a_j$ – термин атрибута; Syn_{ij} – синоним термина атрибута, который можно представить как $Syn_{ij} = \{ \overline{synonym} \}$; $prop_{ij}$ – характеристика атрибута; $range_{ij}$ – ограничение, накладываемое на значения, которые может принимать атрибут (меньше, больше, равно); $aggregation_{a_{ij}}$ – агрегирующая функция, применимая к атрибуту (максимум, минимум, среднее значение, ...); $action_{a_{ij}}$ – действие, допустимое над атрибутом (найти значение, упорядочить, ...).

Технология формирования описания для объекта TNo_i представлена следующей последовательностью действий.

1. Эксперт даёт развернутое описание данных, содержащихся в указанной таблице БД в виде одного предложения.

2. Производится автоматический синтаксический разбор предложения. Слово или словосочетание, выделенное из текста и являющееся кандидатом на термин объекта, должно быть найдено в словаре предметной области.

3. Из остальных членов предложения формируются характеристика объекта ($prop_i$).

4. В словаре предметной области отыскивается множество синонимов объекта (Syn_i).

5. Эксперт проверяет и утверждает описание объекта либо вносит в него корректировки.

Технология формирования описания для атрибута TNa_{ij} представлена следующей последовательностью действий.

1. Эксперт даёт развернутое описание сущности атрибута в одном предложении.

2. Производится автоматический синтаксический разбор предложения. Слово или словосочетание – кандидат на термин атрибута должно быть найдено в словаре предметной области.

3. Из остальных членов предложения формируются характеристики объекта ($prop_{ij}$).

4. Эксперт вводит словесные описания ограничений на значения атрибута ($range_{ij}$).

5. Эксперт вводит словесные описания агрегирующих функций, которые могут применяться к значениям атрибута ($aggregation_{ij}$).

6. Эксперт вводит словесные описания действий, которые допустимы над данными данного атрибута ($action_{ij}$).

7. В словаре предметной области отыскивается множество синонимов атрибута (Syn_{ij}).

8. Эксперт проверяет и утверждает описание атрибута, либо вносит в него корректировки.

В качестве примера рассмотрим создание описания для фрагмента БД, приведенного на рис. 2. Здесь представлены две связанные таблицы *employees* и *department*.

Тексты, введенные экспертом для описания структуры БД, приведены на рис.3 и 4. Для разбора текстов, описывающие таблицы и их колонки, был использован синтаксический анализатор Cognitive Dwarf [10], [11]. Например, в результате разбора текста объектов TNo_i и атрибутов TNa_{ij} введенного экспертом, получим описание для таблиц *employees* (рис. 3) и *department* (рис. 4).

Анализ запроса пользователя. Предлагается запрос пользователя Q представить в виде

$$Q = \langle \{to_i\}, \{ta_{ij}\}, \{Z_{Aij}\}, \{r_{ij}\}, agg_{ij}, act_{ij} \rangle, \quad (9)$$
$$\overline{i = 1, n}$$

где to_i – термин объекта из запроса; ta_{ij} – термин атрибута для термина объекта запроса Q ;

Z_{Aij} – значение атрибута; r_{ij} – диапазон значений атрибута ta_{ij} ; agg_{ij} – агрегирующая функция, примененная к атрибуту ta_{ij} ; act_{ij} – действие над атрибутом ta_{ij} в запросе.

Технология анализа запроса пользователя Q представлена следующей последовательностью действий.

1. Определяется запрос Q в виде предложения на ЕЯ.

2. Выделяется слово или словосочетание, которое может быть терминам объекта to_i .

3. Выделяются слова или словосочетания, которые могут быть терминами каждого атрибута ta_{ij} .

4. Выполняется поиск и выделение возможных диапазонов r_{ij} значений для каждого атрибута ta_{ij} в запросе.

5. Выполняется поиск и выделение возможных агрегатных функций agg_{ij} , применяемых к атрибуту ta_{ij} в запросе.

6. Выполняется поиск и выделение возможных действий act_{ij} над атрибутами ta_{ij} в запросе.

7. Выполняется поиск и выделение возможных значений Z_{Aij} для каждого атрибута ta_{ij} в запросе.

8. На основании найденных и выделенных значений формируется ответ пользователю.

При разборе запроса Q пользователя могут выявиться слова и словосочетания, которые не являются понятиями из предметной области, но определяет форму представления объектов и атрибутов в ответе на вопрос («список», «информация», «все», ...). Предложено заранее определить эти слова и использовать в качестве указания на формирование в ответе множества объектов или атрибутов.

Пример результата анализа запроса пользователя на ЕЯ – «Показать список фамилий сотрудников, с заработной платой больше чем 2000, отсортированный по отделам, в которых они работают» приведен на рис. 5.

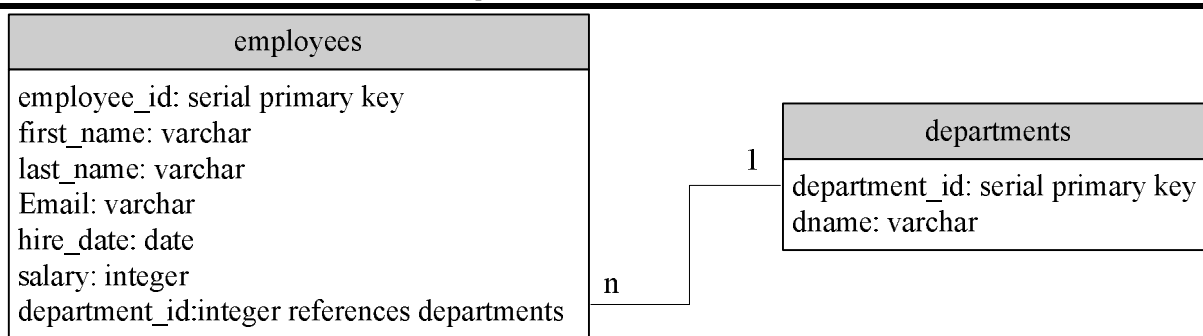


Рис. 2. Структура фрагмента БД

Описание	Термин	Синоним	Характеристика	Ограничение	Агрегация	Действие
employees						
сотрудники фирмы	сотрудник	работник служащий	фирма			
name						
имя сотрудника фирмы	имя	фио, фамилия	сотрудник фирмы			упорядочить, отсортировать
post						
должность сотрудника фирмы	должность	пост, звание	сотрудник фирмы			упорядочить, отсортировать
email						
электронный адрес сотрудника фирмы	электронный адрес	почта	сотрудник фирмы			
hire_date						
дата принятия сотрудника в фирму	дата принятия	дата поступления	сотрудник в фирме	меньше чем, больше чем		упорядочить, отсортировать
salary						
заработная плата сотрудника фирмы	заработная плата	доход, оплата	сотрудник фирмы	меньше чем, больше чем	максимум, минимум, сумма	упорядочить, отсортировать

Рис. 3. Описание объекта (TNo_i) и атрибута (TNa_{ij}) для таблицы employees

Описание	Термин	Синоним	Характеристика	Ограничение	Агрегация	Действие
department						
подразделения и отделы фирмы	подразделен ие	отдел	фирма			
dname						
название подразделен ия фирмы	название подразделен ия	название отдела	фирма			упорядочить, отсортировать

Рис. 4. Описание объекта (TNo_i) и атрибута (TNa_{ij}) для таблицы department

Запрос	Объект	Атрибут	Характеристика	Ограничение	Агрегация	Действие
Показать список фамилий сотрудников, с заработной платой больше чем 2000, отсортированный по отделам, в которых они работают	сотрудник	фамилия, плата		больше чем		
	отдел	отдел				отсортировать

Рис. 5. Пример результата анализа запроса на ЕЯ

Формирование ответа для пользователя. На основе сравнения описания объекта из МБД (5) и представления запроса пользователя (9) предлагается формировать запрос для выбора данных из БД и ответ пользователю на ЕЯ.

Если результат анализа не позволяет сгенерировать запрос, то пользователю выдается ответ на ЕЯ с указанием причины, вследствие которых не может быть сгенерирован запрос (таблица).

Если в запросе на ЕЯ указывается информация, отсутствующая в БД либо такая, что ее не возможно получить ввиду структурных ограничений, то запрос сохраняется для последующего анализа (статистической обработки) с целью принятия решения о развитии ИС.

Формирование ответа пользователю, базируется на том, что заранее будут предусмотрены шаблоны положительных и отрицательных ответов для объектов O_i и атрибутов A_{ij} в соответствии с найденной информацией в МБД.

Пример формирования ответа по шаблону.

Запрос пользователя: «Показать список фамилий сотрудников с указанием их отделов, и отсортировать сотрудников по возрасту».

Пример ответа:

Имеется информация: список фамилий сотрудников и отделы.

Нет информации: возраст.

Используя предложенные технологии можно подсчитать количественную оценку эффективности на основе следующих данных.

Для создания МБД расходуется время высококвалифицированного специалиста по БД. Это можно рассчитать за формулой

$$t_m = (n_t * n_c) t_f, \quad (10)$$

где n_t – количество таблиц в БД; n_c – количество колонок; t_f – среднее время составления фразы описания таблицы или колонки.

Например, для БД, имеющей 10 таблиц, 10 колонок, при времени составления фразы около двух минут, получим созданную МБД за 3 часа 20 минут.

При эксплуатации МБД успешный запрос пользователя выполняется за время

$$t_{uq} = t_q + t_s + t_a, \quad (11)$$

где t_q – время выполнения SQL-запроса в БД;

t_s – время составления запроса на ЕЯ (от 1 до 10 минут), t_a – время анализа запроса неквалифицированного пользователя.

Экспериментальные данные показывают, что t_a превышает t_q на 30 %. Например, успешный запрос пользователя выполняется за 1 минуту.

Если такой запрос будет выполняться обычным образом (с привлечением прикладного программиста), то время получения ответа будет:

$$t_s = t_o + t_c + t_q + t_d, \quad (12)$$

где t_o – время формулировки и подачи заявки на получение информации (от 10 до 20 минут),

t_c – время составления запроса программистом (от 20 до 30 минут),

t_d – время получения ответа на заявку (5 от 10 до минут); t_q – время выполнения SQL-запроса в БД.

Например, для выполнения запроса с привлечением программиста понадобится от 35 до 60 минут. Если предположить, что в некоторой организации в течение дня может поступать около 50 заявок, то уже по истечении одного дня будет наблюдаться экономия времени квалифицированных программистов. К тому же нужно добавить, что оперативно принятое решение является

важным источником экономической выгоды.

Типы генерируемых ошибок

№	Тип ошибки	Генерируемый ответ
1	$to_k \neq term_o_i \vee to_k \notin Syn_i$	Понятие O_i не является данными, хранимыми в БД.
2	$to_k = term_o_i \vee to_k \in Syn_i$ (объект O_i) $to_l = term_o_j \vee to_l \in Syn_j$ (объект O_j) $O_i \notin R_{ij} \notin O_j$	Понятия O_i и O_j соответствуют данным в БД, но не существует связи между ними. Необходимо переформулировать запрос либо разбить его на два запроса.
3	$to_k = term_o_i \vee to_k \in Syn_i$ $ta_{kj} \neq term_a_{ij} \vee ta_{kj} \notin Syn_{ij}$	Понятие O_i соответствует данным хранимым в БД, однако его характеристика A_{ij} не содержится в БД. Нужно точнее определить запрос.
4	$to_k = term_o_i \vee to_k \in Syn_i$ $ta_{kj} = term_a_{ij} \vee ta_{kj} \in Syn_{ij}$ $ta_{lj} \neq term_a_{ij} \vee ta_{lj} \notin Syn_{ij}$	Запрос может быть выполнен частично. Понятие A_{ij} существует, а понятие A_{ij} не является данными, хранимыми в БД. Нужно исправить текст запроса.

Выводы

Предложенная модель базы данных отображает хранимую в РБД информацию в виде естественно-языковых конструкций. Предлагается анализировать запросы пользователя на естественном языке и представлять в виде аналогичных конструкций. Разработана технология анализа запроса пользователя для определения возможного выполнения запроса, либо с указанием допущенных ошибок с точки зрения возможности получения запрашиваемой информации. Представленная технология может быть использована в различных автоматизированных системах с РБД для организации интерфейса пользователя на ЕЯ, что позволит значительно расширить число активных пользователей информационных систем.

Список использованной литературы

1. Большакова Е. И. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика / Е. И. Большакова, Э. С. Клышинский, Д. В. Ландэ, А. А. Носков, О. В. Пескова, Е. В. Ягунова. – М. : МИЭМ, 2011. – 272 с.
2. Кунгурцев А. Б. Обработка текстов и сообщений на естественном языке в информационных системах / А. Б. Кунгурцев, Я. В. Поточняк // Восточно-Европейский журнал передовых технологий, материалы международной научной конференции «Научная периодика славянских стран в условиях глобализации». – К. : – 2012. – № 5/2(7). – С. 2 – 5.

3. Акимов О. М. Интеллектуализация интерфейса базы данных / О. М. Акимов, В. А. Шапцев // Известия Томского политехнического университета. – 2009. – Т. 314. – № 5. – С. 137 – 139.
4. Steven Bird, Ewan Klein, and Edward Loper. Natural Language Processing with Python, (2009), *Published by O'Reilly Media*, 481 p.
5. Bates M. Models of natural language understanding, (1995), *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 92, No. 22. pp. 9977 – 9982.
6. Yucong Duan, and Christophe Cruz. Formalizing Semantic of Natural Language through Conceptualization from Existence, (2011), *International Journal of Innovation, Management and Technology*, pp. 37 – 42.
7. Christopher D. Manning, Hinrich Schutze. Foundations of Statistical Natural Language Processing, (1999), *MIT Press*, 680 p.
8. Gabrilovich E., and Markovitch Sh. Wikipedia-based Semantic Interpretation for Natural Language Processing, (2009), *Journal of Artificial Intelligence Research* 34, pp. 443 – 498.
9. Гарсиа-Молина Г., Системы баз данных. Полный курс / Г. Гарсиа-Молина, Дж. Ульман, Дж. Уидом.// Издательский дом Вильямс. – 2003. – 1088 с.
10. Программный пакет синтаксического разбора и машинного перевода [Электронный ресурс] – Режим доступа: <http://cs.isa.ru:10000/dwarf/>. – 10.09.2008.

11. Кунгурцев А. Б. Выбор компонентов для анализа сообщений на естественном языке в информационных системах / А. Б. Кунгурцев, Я. В. Поточняк, А. А. Янчук // Материалы III международной конференции студентов и молодых ученых «Современные информационные технологии 2013» – Одесса : – 2013. – С. 124 – 125.

Получено 28.04.2014

References

1. Bol'shakova Ye.I., Klyshinskiy E.S., Lande D.V., Noskov A.A., Peskova O.V., Yagunova Ye.V. *Avtomaticheskaya obrabotka tekstov na yestestvennom yazyke i komp'yuternaya lingvistika* [Automatic Processing of Natural Language Texts and Computational Linguistics], (2011), Moscow, Russian Federation, *MI – EM*, 272 p. (In Russian).

2. Kungurtsev A.B., and Potochnyay Y.V. *Obrabotka tekstov i soobshcheniy na yestestvennom yazyke v informatsionnykh sistemakh* [The Processing Text and Messages on the Natural Languages in Information Systems], (2012), *Vostochno-Yevropeyskiy Zhurnal Peredovykh Tekhnologii, Materialy Mezhdunarodnoy Nauchnoy Konferentsii Nauchnaya Periodika Slavyanskikh Stran v Usloviyakh Globalizatsii*, Kiev, Ukraine, No. 5/2 (7), pp. 2 – 5. (In Russian).

3. Akimov O.M., and Shaptsev V.A. *Intellektualizatsiya interfeysa bazy dannykh* [Intellectualization Database Interface], (2009), *Izvestiya Tomskogo Politehnicheskogo Universiteta*, Vol. 314, No. 5 – pp. 137 – 139 (In Russian).

4. Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*, (2009), *Published by O'Reilly Media*, 481 p.

5. Bates M. Models of natural language understanding. *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 92, No. 22 (1995), pp. 9977 – 9982.

6. Yucong Duan, Christophe Cruz. Formalizing Semantic of Natural Language through Conceptualization from Existence, (2011), *International Journal of Innovation, Management and Technology*, pp. 37 – 42.

7. Christopher D. Manning, Hinrich Schutze *Foundations of Statistical Natural Language Processing*, (1999), *MIT Press*, 680 p.

8. Gabrilovich E. Wikipedia-based Semantic Interpretation for Natural Language Processing, (2009), *Journal of Artificial Intelligence Research* 34, pp. 443 – 498.

9. Garcia-Molina G., Ulman J., and Widom J. *Sistemy bazy danykh. Polniy kurs* [Database Systems. Full course], 2003, *Izdatelckiy dom Vilyams*, 1088 p. (In Russian).

10. *Programmnyy paket sintaksicheskogo razbora i mashinnogo perevoda*, [Software Package Syntactic Parsing and the Machine Translation], 2008, (Elektronnyy Rresurs), Rezhim Dostupa: <http://cs.isa.ru:10000/dwarf/>. – 10.09.2008. (In Russian).

11. Kungurtsev A.B., Potochnyay Y.V., and Yanchuk A.A. *Vybor komponentov dlya analiza soobshcheniy na yestestvennom yazyke v informatsionnykh sistemakh* [Selecting Components for the Analysis of Natural Language Messages in Information Systems], (2013), *Materialy III Mezhdunarodnoy Konferentsii Studentov i Molodykh Uchenykh "Sovremennyye Infomatsionnyye Tekhnologii 2013"*, Odessa Ukraine, pp. 124 – 125. (In Russian).



Кунгурцев Алексей Борисович, канд. техн. наук, проф. каф. системного программного обеспечения Одесского нац. политехн. ун-та, пр. Шевченко, 1, Одесса, Украина, 65044. Тел.: 7058-566
E-mail: abkun@te.net.ua



Поточняк Яна Владимировна, аспирант каф. системного программного обеспечения Одесского нац. политехн. ун-та, тел. (066)13-22-363.
E-mail: yana_onpu@mail.ru