М. А. Шумская

АВТОМАТИЧЕСКОЕ ФОРМИРОВАНИЕ ОТВЕТА НА ВОПРОС ПОЛЬЗОВАТЕЛЯ КАК БАЗОВАЯ ФУНКЦИЯ ВОПРОСНО-ОТВЕТНОЙ ПОИСКОВОЙ СИСТЕМЫ

В последние годы в рамках технологий *Text Mining* развивается новое направление, связанное с разработкой интеллектуальных вопросно-ответных поисковых систем. Основные отличия такого рода системы от классической поисковой системы заключаются в ее умении обрабатывать запросы пользователя, сформулированные в виде вопросительных предложений на естественном языке, а также представлять результаты поиска не в виде ранжированных адресов сайтов, релевантных запросу, а в виде конкретного текстового фрагмента, содержащего нужную пользователю информацию.

Автором данной статьи была создана формальная модель англоязычной вопросно-ответной поисковой системы. Материалом исследования послужили тексты 19-ти научно-популярных статей на английском языке, а также 45 специальных вопросов по их содержанию, сформулированные жителями разных стран, для которых английский язык является родным, вторым или иностранным. Первая часть формальной модели (анализатор запросов) отвечает за анализ запроса пользователя, представленного в виде специального вопроса, начинающегося с вопросительного слова *When* или *Where*. Вторая часть формальной модели (синтезатор ответов), опираясь на результаты поиска, синтезирует конкретный ответ на поставленный вопрос. Рассмотрим подробнее процедуру автоматического формирования ответа на вопрос пользователя.

Работа синтезатора ответов состоит из двух основных этапов. На первом этапе во всем массиве найденных документов система должна корректно определить нужный текст (или несколько текстов), который будет соответствовать тематике и смыслу вопроса пользователя. Затем, из набора предложений текста (текстов) ей нужно выбрать предложения-кандидаты, содержащие необходимую пользователю информацию. Второй этап процесса синтеза ответа включает в себя проведение синтаксического анализа предложений-кандидатов, извлечение из них необходимой информации (исключительно на

основе синтаксического анализа), и непосредственно синтез ответа на основе определенной синтаксической структуры и данных, полученных в первой части формальной модели на этапе синтаксического анализа вопроса.

В целом лингвистическая база данных синтезатора ответов, как и база данных анализатора запросов, включает несколько компонентов.

1. Список синтаксических структур вопросов и соответствующих им структур ответов, представленный в табл. 1.

Таблица 1

Список структур вопросов и соответствующих им структур ответов

Структура предложения-запроса	Структура предложения-ответа
Aux.v | Pred = be	
WRB/PL aux.v Subj MV	Subj aux.v MV #MPL
WRB/PL Pred Subj	Subj Pred #MPL
Aux.v = do	
WRB/PL aux.v Subj Pred	Subj Pred #MPL
WRB/PL aux.v Subj Adv. Pred	Subj Adv. Pred #MPL
Aux.v = will|can	
WRB/TM aux.v Subj Pred	Subj aux.v Pred #MTM
WRB/TM aux.v Subj Pred D.Obj	Subj aux.v Pred D.Obj #MTM

2. Алфавитный тегированный словарь (состоящий из слов предложений-кандидатов в ответы), составленный с опорой на демо-версию POS-тегера Иллинойского Университета, США. Фрагмент словаря представлен в табл. 2.

Таблица 2

Фрагмент алфавитного тегированного словаря синтезатора ответа

‘s/POS	church/NN	is/VBZ	Serbian/JJ
1966/CD	citizen/NN	island/NN	sides/NNS
3600/CD	clinic/NN	its/PRP$	somewhere/RB
7/CD	construction/NN	Jimmy/NNP	stage/NN
a/DT	Croatia/NNP	man/NN	temples/NNS
Alabama/NNP	dozen/NN	Maryland/NNP	Tesla/NNP
among/IN	emigrated/VBD	Nash/NNP	the/DT
……	……	……	……

3. Список слов, формирующих модуль времени и модуль места. Фрагмент списка представлен в табл. 3.

Таблица 3

Фрагмент списка слов, входящих в состав модулей времени и места

Модуль времени	Модуль места
about 5,000 light-years	*at* St. Peter’s Basilica
around 3600 B.C.	*in* Barcelona
by spring 2016	*in* Bethesda, Maryland
……	……

4. Список контекстуальных синонимов, который формируется для каждого текста. Фрагмент списка представлен в табл. 4.

Таблица 4

Фрагмент списка контекстуальных синонимов

inhabit	live
reign	rule
Kepler-78	Planet, it
Tesla	Nikola Tesla, he
Wales	Jimmy Wales, he
……	……..

5. Список последовательностей тегов, которые соответствуют словам, входящим в именные группы (*noun phrase, NP*). Фрагмент списка представлен в таблице 5

Таблица 5

Фрагмент списка последовательностей тегов слов именных групп

NN	DT NNP NNP
NNS	DT NNP NNS
NNP	JJ JJ NNS
NNPS	NNP DT NNP
DT NNS	DT NN IN NNPS
JJ NNP	DT NN NN NN
NN NNP	DT NN POS NNS
NN NNS	DT JJ NN IN NN
……..	……

6. Список форм глаголов, фрагментарно представленный в табл. 6.

Таблица 6

Фрагмент списка форм глаголов

VB	**VBZ** (для ед. ч.)	**VBP** (для мн. ч.)	**VBD**	**VBN**	**VBG**
arrive, conclude, consecrate, die, grace, live, locate, reserve, rule	VB + s	VB	VB + d	VB + d	VB–e + ing
bomb, crash, finish, vanish	VB + es		VB + ed	VB + ed	VB + ing
crown, discover, exist, found, found, happen, inhabit, reign	VB+”s”				
be	is	are	was – ед. ч. were – мн. ч.	been	being
……..	……	……	……	……...	……

Правила приписывания синтаксических функций словам в зависимости от их принадлежности к различным частям речи и позиции в предложении, а также правила согласования членов предложений прописаны, соответственно в алгоритме анализатора запросов и синтезатора ответов.

Рассмотрим подробнее компоненты лингвистической базы данных синтезатора ответов. Тегированные словари составлялись на основе слов текстов статей и вопросительных запросов с применением демо-версии тегера, т.е. фактически каждый текст проходил POS-обработку, а затем на основе полученного словаря проводился анализ вопросительных предложений. Однако промышленной системе, способной обслуживать множество пользователей с опорой на огромный текстовый массив, был бы необходим единый тегированный словарь, содержащий многие тысячи слов и специальных символов, которые могут идентифицироваться как токены (токен – один символ или последовательность символов от пробела до пробела). В основном токенами являются слова естественного языка, однако в него может входить, например, и апостроф. Так, некоторые системы автоматического POS-тегирования текстов распознают слово *Mike's* как единый токен, другие (как и использованный нами тегер) делят слово на два токена: *Mike* и *'s*. Именно поэтому представленный в таблице 2 фрагмент словаря содержит не только слова и словоформы, но и специальные символы и цифры. Из данных таблицы 3 хорошо видна определенная закономерность в образовании словосочетаний, выполняющих синтаксическую функцию модуля времени и места, что объясняется аналитической природой английского языка. Следовательно, представляется возможным написание правил, по которым эти модули будет определяться автоматически. Сложнее обстоит дело с формированием списка контекстуальных синонимов, поскольку часто слова имеют схожее значение лишь в рамках определенного контекста. Тем не менее, при анализе больших текстовых массивов можно выделить некоторый набор самых часто встречающихся вариантов контекстуальных заменителей и составить из них базовый список, который будет дополнять личный список каждого отдельного текста. Подобные уникальные списки синонимов формируются при первом анализе текста системой и хранятся вместе с поисковым образом каждого документа.

Важным компонентом синтезатора ответа являются правила распознавания и приписывания словам синтаксических функций. В отличие от количества правил, содержащихся в анализаторе запросов (вопросительных предложений), правила синтезатора ответов не так многообразны, поскольку в данном случае нет необходимости проводить полный и глубокий разбор предложений-кандидатов на ответ. Например, из предложения-кандидата ответа на вопрос *When was Jimmy Wales born?* компьютеру необходимо извлечь только конкретную фактографическую информацию, в то время как субъект и предикат ситуации известны из самого вопроса. Таким образом, протегировав предложение-кандидат и включив в него дополнительные теги модулей времени и места, компьютер формирует следующую запись:

Jimmy/NNP Wales/NNP was/VBD born/VBN #MPLin/IN Huntsville/NNP , Alabama/NNP#MPL , #MTMon/IN August/NNP 7/CD , 1966/CD#MTM .

Из примера хорошо видно, что при большом, или практически полном соответствии слов запроса пользователя и слов предложения текста глубокий синтаксический анализ с полным определением синтаксических функций

является излишним. Еще один заслуживающий внимания вопрос связан с длиной каждого модуля. Как показано на примере и в таблице 3, иногда слова с одинаковыми тегами могут быть объединены в один более крупный блок.

Не менее важным компонентом синтезатора ответов является набор правил согласования членов предложения. Поскольку система уже владеет достаточной информацией для синтеза ответа (сюда можно отнести данные, полученные в ходе анализа вопросов, а также структуры ответов), то набор правила согласования будет небольшим. В самом общем виде они выглядят следующим образом.

- *Если* подлежащее, выраженное нарицательным существительным или именем собственным, имеет тег NN или NNP (только для существительных не входящих в список исключений),

° *то* стоящий за ним глагол с тегом VB должен быть изменен на VBZ, в соответствии с таблицей 6.

- *Если* подлежащее, выраженное нарицательным существительным или именем собственным, имеет тег NNS или NNPS,

° *то* стоящий за ним глагол с тегом VB должен быть изменен на VBP, в соответствии с таблицей 6.

- *Если* подлежащее, выражено нарицательным существительным или именем собственным, а глагол с тегом VBD не является формой глагола *be*,

° *то* глагол с тегом VBD остается неизменным, в соответствии с таблицей 6.

- *Если* подлежащее, выраженное нарицательным существительным или именем собственным, имеет тег NN или NNP, а следующий за ним глагол – форма глагола *be* с тегом VBD,

° *то* глагол *be* принимает форму *was*.

- *Если* подлежащее, выраженное нарицательным существительным или именем собственным, имеет тег NNS или NNPS, а следующий за ним глагол – форма глагола *be* с тегом VBD,

° *то* глагол *be* принимает форму *were*.

- *Если* в вопросительном предложении предикат состоит из вспомогательного глагола *do* и смыслового глагола с тегом VB,

° *то* в ответе глагол с тегом VB принимает время глагола *do* и число, согласованное с предшествующим существительным.

- *Если* в вопросительном предложении предикат состоит из вспомогательного глагола *be* и смыслового глагола с тегом VBN,

° *то* в ответе предикат будет состоять из формы глагола *be,* согласованной в числе с предшествующим существительным, но в сохраненном из вопроса времени, а глагол с тегом VBN останется неизменным.

- *Если* в вопросительном предложении предикат состоял из смыслового глагола *be*,

° *то* в ответе предикат будет состоять из формы глагола *be,* согласованной в числе с предшествующим существительным, но в сохраненном из вопроса времени.

Приведенные выше правила составляют основу алгоритма работы синтезатора ответов. Рассмотрим его подробнее. Работа синтезатора ответа начинается с поиска текстов, а затем входящих в них предложений, которые содержат слова из поискового образа запроса (слова из *Блока Б* анализатора запроса пользователя). На выходе получается одно либо несколько наиболее подходящих для синтеза ответа предложений-кандидатов. В том случае, если предложений-кандидатов найдено не было, с помощью контекстуального словаря и правил нормализации производится изменение поискового образа запроса, который отправляется в систему для повторного поиска.

На следующем этапе с опорой на алфавитный тегированный словарь словам предложений-кандидатов приписываются теги частей речи. После этого найденные предложения проверяются на наличие в них модулей времени и места, а затем на содержание искомого тега модуля, который был получен на этапе анализа вопросительного предложения (запроса пользователя) и хранится в *Блоке Б*. Таким образом в системе остается лишь одно, самое вероятное предложение-кандидат. С этого момента начинается непосредственно синтез ответа пользователю. После того как была получена синтаксическая информация, содержащаяся в предложении-кандидате, компьютер находит в базе данных структуру из *Блока А* (анализатор запроса пользователя) и структуру ответа, которая ей соответствует. После этого структура ответа заполняется подходящими словами. Чтобы исключить вероятность ошибки при синтезе ответа, компьютер проводит проверку на наличие двух одинаковых рядом стоящих слов.

Заключительным этапом синтеза ответа на вопрос пользователя является согласование слов в предложении с опорой на хранящиеся в системе правила. Это позволяет сделать ответ грамматически более точным, непохожим на простой набор слов, что отражает определенную степень интеллектуальности вопросно-ответной поисковой системы.