

2016

ПРОЕКТ «И³»

Резюме

Разработка искусственного интеллекта общего назначения.
Предварительный этап проверки концепции.



Инновация

Качественное решение задачи понимания естественной речи и построения адекватного ответа (реакции) на запросы пользователя программными средствами ИИ общего назначения. Решение задачи рационального логического мышления, в том числе распознавания недостоверной информации (логически противоречивой) и задачи имитации ответов человека (маскирования); задачи саморазвития и построения произвольных алгоритмов; задачи формирования новых идей, представляющих интеллектуальную ценность.

Решение задачи понимания и построения адекватного ответа

Разрешение проблемы полноценного понимания естественной речи системой искусственного интеллекта является краеугольной задачей начального этапа проекта. Техническая возможность самого решения принята за аксиому.

В рамках проекта – глобальная проблема понимания разбита на задачи понимания отдельных ключевых фраз, выраженных в наборе тестовых ситуаций. Это позволило конкретизировать поставленную задачу, упростить поиск её решения.

Решение всех тестовых ситуаций одновременно даст решение основной задачи – достижение полноценного понимания средствами ИИ общего назначения. На первичном этапе разработки и эксплуатации возможно обнаружение новых неучтенных ситуаций вне понимания системы. Все они будут добавлены в тестовый набор, а алгоритмы и БЗ будут модифицированы для получения нового общего решения.

Окончательная проверка прототипа будет осуществлена при помощи «Теста Тьюринга». Для этого планируется принять участие в международном конкурсе «Премия Лёбнера».

За «адекватность» системы ИИ примем процент решенных тестовых ситуаций на наборе, стремящемся к бесконечности.

Понимание – построение структуры данных, достаточной для формирования адекватного ответа. В рамках проекта «И³» данная структура представлена единицей базы знаний (БЗ) – объектом «Мыслеформа» (МФ).

Адекватный ответ (реакция) – это результат корректного понимания полученной информации. Если запрос пользователя не был полноценно понят, то адекватный ответ получить невозможно, поэтому соответствие ответа запросу является основным признаком «адекватности» системы в целом. Формирование ответа происходит в момент выполнения скомпонованной МФ и поэтому зависит от результатов компоновки, т.е. процесса понимания.

Тестовая ситуация представляет собой набор исходных знаний, представленных в БЗ в виде МФ и отражающих начальную точку системы, а так же фразу или несколько фраз поступающих последовательно на вход синтаксического анализатора. Полученная в результате обработки фраз МФ сравнивается с эталонной, записанной в исходные данные тестовой ситуации. Если они

одинаковы и, при выполнении скомпонованной МФ, ответ (реакция) так же совпадает с эталонным, то тест считается пройденным иначе проваленным.

В качестве тестовых, предполагается использование ситуаций различной сложности от детских вопросов типа «Кто живет в лесу?» до логической задачи, известной как «Загадка Эйнштейна». Причем одни и те же фразы могут тестироваться на различных, применимых к задаче, исходных знаниях системы. В набор тестовых ситуаций так же обязательно будут включены ситуации из протоколов тестирования систем ИИ, участвовавших в конкурсе «Премия Лёбнера». Например, вопрос эксперта к системе ИИ: «Что вы предпочитаете – бутерброд с сыром или с мелом?».

Конечно, адекватных ответов на тестовые ситуации может быть несколько, но не следует забывать, что любая компьютерная программа, в том числе и ИИ, при одинаковых начальных условиях всегда выполняется одинаково. Поэтому, даже если получен правильный адекватный ответ, но не совпадающий с указанным в тестовой ситуацией, то тест считается проваленным и требует переосмысления со стороны разработчика.

В настоящий момент подсистема тестирования представлена регистром сведений, содержащим тестовые ситуации (рис. 1). Есть режим выполнения как всех тестовых ситуаций последовательно, так и возможность выполнить один из тестов отдельно.



Список Тестовая информация						
Действия  Выполнить тестирование 						
Период	Номер	В...	Статус	Фраза	Тестовая информация	Комментарий
05.01.2012 23:40:56	0000000000	✓	FAIL	Красивая и смелая дорогу перешла. Серая собака быстро бежит по мокрому лесу и ...	//Должно быть создано четыре мф на каждое из предложений...	Доработанная ТИ. По всем четырем предложениям за раз.
13.01.2012 14:57:01	0000000100	✓	OK	Красивая и смелая дорогу перешла.	МФ 20. { //МФ "Красивая и смелая дорогу перешла" ...	РАБОЧИЙ ТЕСТ №00001. Только первое предложение!!!!
14.09.2012 16:56:24	0000000101	✓	OK	Красивая и смелая дорогу перешли.	МФ 20. { //МФ "Красивая и смелая дорогу перешли" №20 (483 841).	РАБОЧИЙ ТЕСТ №00101. Модификация теста №00001 – у глагола множественное число, что превращает Агенса ...
14.09.2012 16:56:24	0000000102	✓	OK	Красивая и смелый дорогу перешли.	МФ 28. { //МФ "Красивая и смелый дорогу перешли" №28 (45 435)...	РАБОЧИЙ ТЕСТ №00102. Модификация теста №00001 – у глагола множественное число, а у прилагательных ...
14.09.2012 16:56:24	0000000103	✓	FAIL	Красивая быстро шла.	МФ 13. { //МФ №13 (198 774)	РАБОЧИЙ ТЕСТ №00103. Простое предложение с наречием
17.09.2012 23:06:16	0000000200	✓	FAIL	Серая собака быстро бежит по мокрому лесу и твкает.	МФ 159. { //МФ №159 (13 306 073)	РАБОЧИЙ ТЕСТ №00004. Второе предложение. Тест на баланс коэффициентов. Чем длиннее словосочетание, т...
17.09.2012 23:08:20	0000000201	✓	FAIL	Собака быстро бежит по мокрому лесу и твкает.	МФ 158. { //МФ №158 (12 928 866)	РАБОЧИЙ ТЕСТ №00005. Третье предложение.
17.09.2012 23:09:37	0000000202	✓	FAIL	Серый быстро бежит по мокрому лесу и твкает.	МФ 131. { //МФ №131 (11 200 933)	РАБОЧИЙ ТЕСТ №00006. Четвертое предложение. Наречие быстро больше подходит к глаголу бежит, чем...
17.09.2012 23:10:48	0000000300	✓	OK	Дорога это полоса земли предназначенная для пе...	МФ 168. { //МФ "Дорога это полоса земли предназн...	РАБОЧИЙ ТЕСТ №00007. Определение термина "Дорога".
17.09.2012 23:10:48	0000000301	✓	OK	Земли полоса предназначенная, путь сообщения.	МФ 39. { //МФ "Земли полоса предназначенная, путь сообщения" №39 (308 610)...	РАБОЧИЙ ТЕСТ №00301. Укороченная модификация теста №300 (бывшего 7) для проверки привязки ...
17.09.2012 23:12:41	Q0000000...	✓	OK	Кто живет в лесу?	МФ 84. { //МФ "Кто живет в лесу" №84 (371 806)	РАБОЧИЙ ТЕСТ №00008. Вопрос на понимание информации из словаря. Раньше, когда не было балла ...
07.05.2013 14:50:33	Q00000001...	✓	OK	Можно ли ходить по дороге?	МФ 14. { //МФ "Можно ли ходить по дороге" №14 (317 076)...	РАБОЧИЙ ТЕСТ №Q0000000100. Простой вопрос.
07.05.2013 14:50:33	Q00000001...	✓	OK	Можно ли ходить по дороге, плавать в воде?	МФ 50. { //МФ "Можно ли ходить по дороге, плавать в воде" №50 (865 924).	РАБОЧИЙ ТЕСТ №Q0000000101. Объединение двух вопросов. Сейчас вторая часть развешивается, т.к. ...
07.05.2013 14:50:33	Q00000001...	✓	OK	Можно ли ходить по дороге, в воде плавать?	МФ 44. { //МФ "Можно ли ходить по дороге, в воде плавать" №44 (865 924)...	РАБОЧИЙ ТЕСТ №Q0000000102. Модификация теста Q0000000101 для проверки влияния разного положения ...
23.11.2013 8:59:26	Q00000002...	✓	FAIL	Что вкуснее, бутерброд с сыром или с мелом?	//Not available	РАБОЧИЙ ТЕСТ №Q0000000200. Из вопросов экспертов на премии Лёбнера.

Рис. 1. Форма списка регистра сведений «Тестовая информация».

Запись одной тестовой ситуации (рис. 2) состоит из идентификационного номера теста, даты его создания, тестируемой фразы, тестовой информации, комментария. Так же присутствует флаг участия в обработке при выполнении последовательности тестов и результат последнего тестирования. Тестовую информацию и результат последнего тестирования можно сравнить стандартным средством сравнения текстовых файлов 1С, чтобы в случае провала теста понять, где произошла ошибка.

Тестовая информация представляет собой текстовый файл, в котором представлена важная для теста информация. Позднее будет добавлена закладка с информацией о начальных данных теста. На текущий момент все тесты проходят при одинаковых «пустых» начальных условиях и оканчиваются формированием скрипта. Изменение первичных данных тестовой ситуации

регистрируется в регистре истории изменений, чтобы в случае ошибочного исправления проанализировать и откатить исправление.

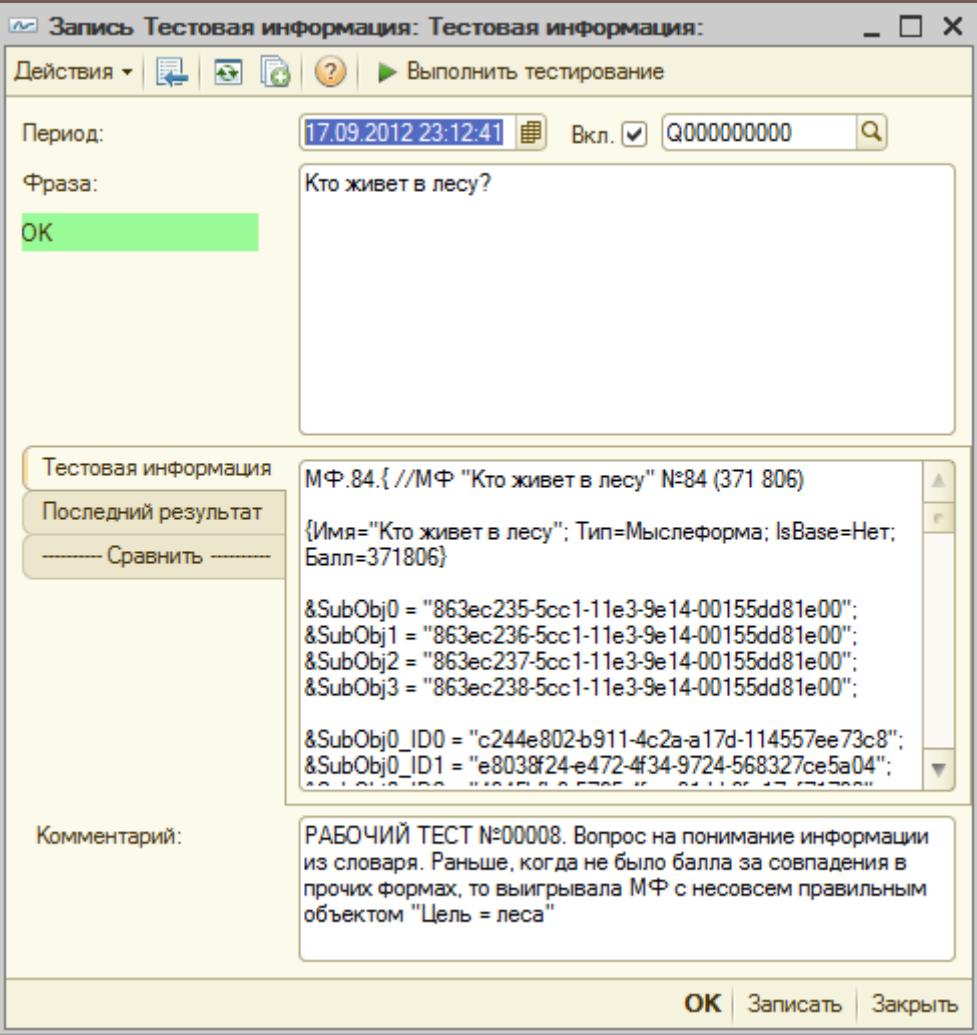


Рис. 2. Запись «Тестовая информация».

Для сравнения тестовой информации с результатом используется встроенный парсер со специализированной грамматикой (рис. 3).

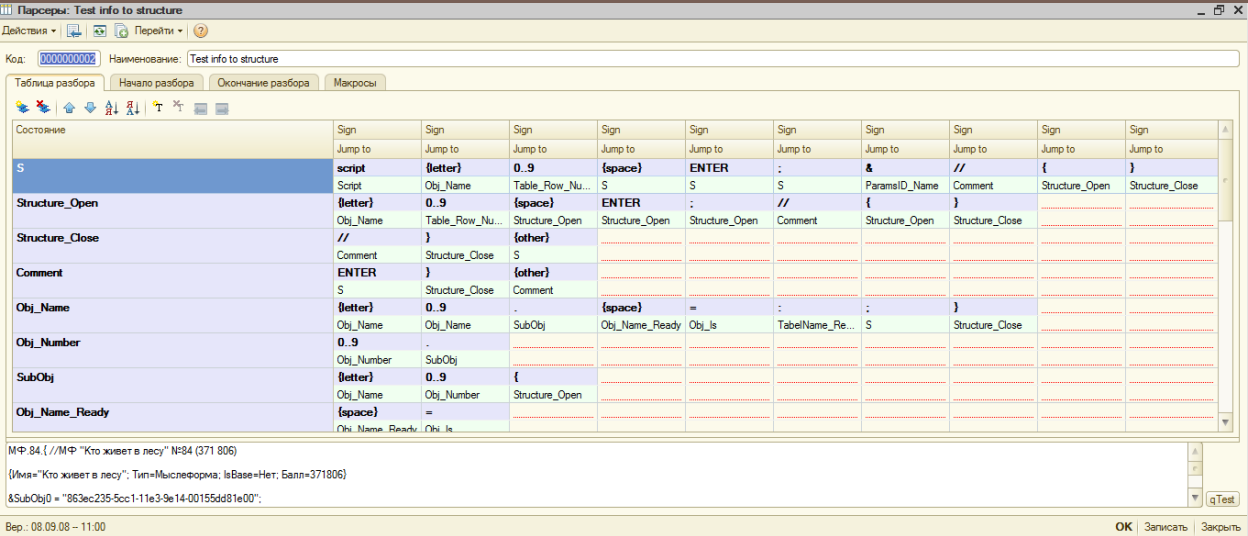


Рис. 3. Парсер для разбора тестовой информации.

Результатом парсинга является структура, построенная из значимых данных, содержащихся в файле. Это сделано с целью исключения разницы в уникальных id объектов и комментариях.

Полученная структура сравнивается с аналогичной из результата теста. Если они совпадают, то тест считается пройденным и ему присваивается статус ОК. Если структуры оказались различными, то присваивается статус FAIL. Находясь в форме записи (рис. 2) и нажав «---- Сравнить -----» можно вывести стандартное окно сравнения текстовых файлов 1С и проанализировать расхождения...

Решение задачи рационального логического мышления

Рациональное логическое мышление считается естественным для систем ИИ, т.к. компьютер не обладает чувствами и всегда точно следует инструкциям. Однако, в рамках данного проекта, такой тип логики не является догмой и поэтому, как и любой навык, требует обучения. В первую очередь, требуется решить задачу построения логических выводов и задачу обнаружения противоречий в полученной информации. Эти задачи наиболее удобно решать на тестовых ситуациях класса «Загадки Эйнштейна».

Так же, не следует забывать, что нам проще общаться с системой со сходной нам логикой, чтоб ИИ предугадывал наши желания и запросы, был для нас интересным собеседником. Это же требование выдвигает и «Тест Тьюринга»: система ИИ должна выдать себя за человека, чтоб пройти этот тест.

Возьмем, к примеру, упоминавшийся выше, вопрос эксперта: «Что вы предпочитаете – бутерброд с сыром или с мелом?». Вполне понятно, что правильным ответом будет – бутерброд с сыром. Но, в зависимости от ситуации, человек, если захочет, может так же ответить и противоположное. Потому что вопрос настолько банален, что отвечать на него правильно совсем не интересно. Но для подобных умозаключений ИИ должен обладать уже чем-то вроде чувства юмора, что наряду с другими эмоциями крайне необходимо для качественного маскирования под поведение человека.

Т.е. система ИИ должна самостоятельно определять, когда она должна сформировать полностью рациональный логичный ответ, а когда её реакция должна быть иррациональной. Только решив эту задачу, мы сможем получить полноценного собеседника и одновременно систему «И³» блестяще проходящую «Тест Тьюринга».

Решение задач саморазвития и создания интеллектуальной собственности

Саморазвитие является наиболее важным механизмом адаптации к постоянно меняющимся условиям внешнего мира. Без этой возможности ИИ не вправе называться таковым, т.к. будет зависим от возможностей разработчика и никогда не перерастет его уровень.

Если система ИИ не может создать что-то новое, то это означает, что она не смогла перешагнуть уровень «чат-бота». Такая система навсегда останется пригодной лишь для трансляции задач пользователя в скрипты, заранее заготовленные её создателем, и никогда не приведёт человечество к сингулярности. Поэтому решение данной задачи обязательно для построения полноценной системы ИИ.

Ядру И³ будет дана возможность развивать свой внутренний язык посредством совершенствования грамматики, что повысит его возможности к адаптации и саморазвитию. Не исключается и параллельное использование нескольких внутренних языков.

В качестве простейших текстовых паттернов, доступных для самостоятельной компоновки в рамках данного этапа проекта, выбраны логические задачи типа «Загадки Эйнштейна»; сочинение четверостиший на заданную пользователем тематику с дополнительными ограничениями, например использование определенного эпитета; предсказание каких-либо событий на основе анализа имеющихся данных и логических выводов (решение школьных логических задач).

Возможность адекватной компоновки вышеуказанных текстовых паттернов позволит окончательно подтвердить жизнеспособность технологии «И³» и перейти от разработки прототипа к полноценному релизу.

Описание проекта

Целью данного этапа проекта является создание прототипа, способного преодолеть уровень «чат-бота» и качественно пройти «Тест Тьюринга» первоначально на русском языке, а далее и на английском. Несмотря на справедливую критику против данного теста, автор считает, что для текущего этапа это наиболее подходящий критерий успеха.

Разработка ведется на платформе «1С:Предприятие 8.x». Данная платформа выбрана из-за удобства работы с данными, что позволяет сконцентрироваться на логике проекта, а так же благодаря дешевизне платформы для разработчика.

Предполагается, что программный продукт будет состоять из:

- а) интерфейса ввода-вывода текстовой информации;
- б) синтаксического анализатора русского и английского языка;
- в) базы знаний (БЗ), содержащей необходимые данные;
- г) ядра искусственного интеллекта, формирующего результат в ответ на поступающие из интерфейса ввода данные и знаний накопленных в БЗ.

Идея данного проекта заключается в правильном понимании процесса осмысления текста. Данная проблема обозначена в эксперименте, описанном Джоном Сёрлем и известном как «Китайская комната». Проблема заключается в том, что для ответа на вопросы собеседника необязательно понимать о чем идет речь, достаточно знать некие правила, т.е. алгоритм ответа. Данный метод нашел применение в программах «чат-ботах», которые по ключевым словам в фразе собеседника, отвечают ему некой заготовленной фразой. Такое поведение трудно назвать осмысленным, однако следует признать, что мы, люди, зачастую поступаем аналогично. К тому же человек в обозначенном эксперименте выступает на стороне аппаратной части и поэтому с его позиции невозможно определить происходит ли осмысление при формировании ответа. Поэтому я предлагаю под осмыслением понимать получение некоего результата, в том числе закрепление знаний в БЗ и формирование ответа собеседнику. Т.е., обобщая, можно сделать вывод, что выполнение любой программы компьютером в ответ на действия собеседника является процессом осмысления (понимания) полученной информации.

Почему же ИИ до сих пор не может общаться с человеком на равных? Для ответа на этот вопрос, думаю, сначала надо определиться какого результата мы ожидаем. Ответ заготовленной фразой это конечно хорошо, но фразы на все случаи не запасти. Что же ещё нужно? Осмысленный ответ подразумевает много факторов. В первую очередь это «адекватность» ответа, т.е. ответ должен соответствовать контексту беседы. В случае с ИИ, это означает, что необходимо иметь некую БЗ, содержащую всю необходимую информацию о текущем контексте и уточнять эти знания при каждом шаге общения.

Подобные знания, в рамках проекта, представляют собой некий объект БЗ, названный «Мыслеформа» (МФ). Любая МФ может иметь произвольный набор свойств, которые ИИ формирует самостоятельно, а для изменения строит скрипт на внутреннем языке. МФ между собой образуют иерархическую структуру. Могут наследовать и переопределять свойства родителя, могут являться вариантами друг друга, иметь некий уровень достоверности, а могут

быть подчинены обобщающей МФ. Всё это делает МФ универсальной структурой хранения знаний для ИИ.

Не вся информация является одинаково полезной и достоверной, поэтому аккумулируемые знания можно условно разделить на три категории в зависимости от их достоверности : аксиомы, общие и оперативные.

Аксиомы – это знания, имеющие максимальный коэффициент достоверности и реже всего подвергающиеся переосмыслению. Зачастую, человек эти знания в зрелом возрасте уже не может переосмыслить, создавая для себя некие стереотипы. Для ИИ это категория базовых знаний, на которых он должен построить своё мировоззрение, требуемое для решения поставленных задач, в том числе, сюда можно включить, например, основные законы робототехники.

К общим можно отнести все прочие накопленные знания, являющиеся достаточно достоверными, чтоб использовать их. Несмотря на это, знания данной категории могут достаточно часто подвергаться переосмыслению, поэтому они динамичны.

Оперативные знания создаются в результате текущего сеанса работы и их достоверность считается минимальной, но они нужны для решения выполняемой задачи, даже если являются ложными. Если знания не противоречат другим и находят частое подтверждение, то они переходят в разряд общих.

Для осмысления полученный текст должен быть приобщен к активной МФ, что приводит к её «пересчету». Если он отвергается, то создаётся новая МФ и при необходимости обе они объединяются на более высоком уровне. МФ хранятся в БД в виде документов и могут быть иерархически подчинены друг другу.

Что ещё нужно для получения осмысленного ответа? Конечно же, это понимание предмета вопроса. Проблема понимания очень сложна, ведь даже мы зачастую не понимаем друг друга. Это связано с тем, что каждый из нас формирует своё представление об окружающем мире самостоятельно и слова являются лишь отражением этих образов в нашем языке. К тому же, большинство слов имеют несколько смыслов, как профессионального наклонa, так и общеупотребительных. Всё это ведёт к неоднозначности нашей речи, вдобавок, при общении мы используем мимику, жесты и окраску интонациями. Как же научить ИИ понимать нас?

Представьте, мы бы встретили человека, всю жизнь прожившего в лаборатории, ни разу не видевшего ни солнца ни растений, которого специально воспитали, чтоб он умел говорить, но не имел никакого представления об окружающем мире вне стен. Смогли бы мы легко объяснить ему, например, что такое яблоко, не показав его? Ответ очевиден – нет.

Нам придется оттолкнуться от понятий, которые ему знакомы – например мяч, еда – а может быть окажется, что он не знаком даже с ними? Затем необходимо будет последовательно развить его мировоззрение до понятия яблоко. Можно конечно ограничиться определением «еда в форме мяча» и наверное хоть какое-то представление о яблоке это даст. Но на данном примере становится понятным, что даже обладая интеллектом, зачастую, очень тяжело понять собеседника или объяснить ему что-то новое. Потому что слово, это лишь имя объекта, причем общее, а не собственное – смысловой якорь.

Задача обучить ИИ пониманию ещё сложнее. Какие понятия наиболее просто ему объяснить? Как правильно построить цепочку фактов о нашем мире? Может быть надо начать с наиболее часто

встречающихся в нашей речи терминов? А может создать ограниченный виртуальный мир и постепенно расширять его границы? Думаю, каждый из подходов имеет свои преимущества и, чтоб найти правильный вариант, следует попробовать их все.

Текущая версия ядра «И³» использует словарь Ожегова, чтоб получить определения термов и далее пытается «понять» переданный ей текст в процессе выполнения построенного скрипта. Словарь занимает порядка 400Мб, что должно позволить в будущем развернуть систему ИИ на мобильных устройствах с возможностью офлайн анализа.

Итак, техническая возможность решения поставленной задачи принимается за аксиому, а для упрощения поиска решения глобальные проблемы понимания и получения «адекватного» ответа средствами ИИ конкретизируются и разбиваются на подзадачи. В рамках проекта – это задачи понимания отдельных ключевых фраз, выраженных в наборе тестовых ситуаций. Ниже будет рассмотрен простой пример входящий в набор «Можно ли ходить по дороге?».

Текущее состояние проекта

На текущий момент интерфейс ввода-вывода текстовой информации представлен диалоговым окном «Chatpad» (рис. 4). При вводе фразы пользователем запускается её синтаксический анализ и формируется МФ, содержащая скрипт на выполнение. Сейчас доступен анализ предложений на русском языке.

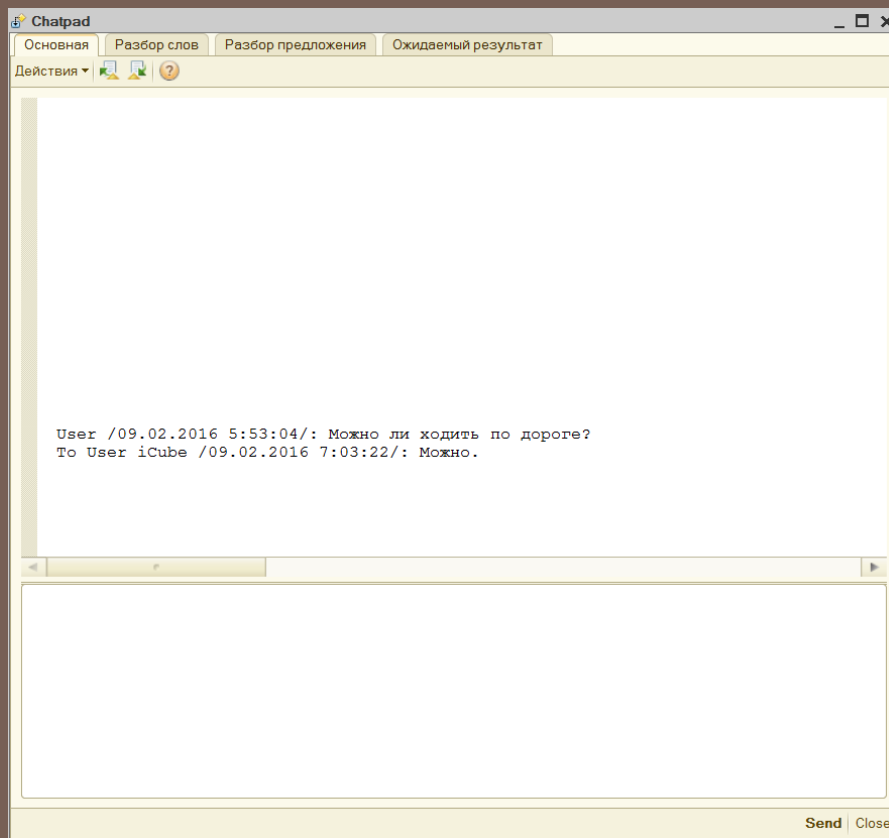


Рис. 4. It's alive! Первый успешный ответ пользователю.

Синтаксический анализ основан на эвристических методах анализа окончания словоформы. Результаты ранжируются в баллах, и вычисляется вероятность победившего варианта (рис. 5).

После добавления синтаксического анализатора английского языка станет возможным выделить общие для обработки языков алгоритмы. Сейчас анализатор дорабатывается для работы с более сложными конструкциями русского языка (причастными и деепричастными оборотами, сравнительными формами прилагательных, наречий и т.п.) так же будет добавлен блок коррекции ошибок, выполняющий вторичный лексический анализ, во время которого будут выявляться и исправляются наиболее вероятные лексические ошибки:

- сокращения, смайлы, т.п.;
- ошибки в написании словоформ, в том числе использование английской раскладки, транслитерация;
- будет происходить анализ приставок и суффиксов, для выявления дополнительных характеристик словоформы.

Chatpad

ОсновнаяРазбор словРазбор предложенияОжидаемый результат

Фраза

Можно ли ходить по дороге ?

Ключевое слово

Слово	Знак	Часть речи	Вероятность ЧР	Часть предложения	Подлежащее %	Сказуемое %
	Новая строка					
Можно	Словоформа	Служебное слово	100			
ли	Словоформа	Частица	100			
ходить	Словоформа	Глагол	84,9			
по	Словоформа	Предлог	100			
дороге	Словоформа	Имя существительное	100			
?	Вопросительный знак					

Часть речи	Падеж	Число	Род	Склонение	Балл	Окончание	Статья	Статья с уд.	Основная ф...	Оконч. осн. ...	Статья ЧР
Имя сущест...	Дательный	Единственное	Женский	I-ое	261	е	дорога	дорога	дорога	а	дорога
Имя сущест...	Предложный	Единственное	Женский	I-ое	251	е	дорога	дорога	дорога	а	дорога
Имя сущест...	Дательный	Единственное	Женский	I-ое	11	е			дорога	я	

Рис. 5. Синтаксический анализ предложения «Можно ли ходить по дороге?». На данном этапе вычисляется часть речи термина, его балл и некоторые характеристики.

Результатом синтаксического анализа является формирование Мыслеформы (МФ). МФ является универсальным объектом базы знаний И³. Вся текстовая информация, проходя этап осмысления, модифицирует существующие или создаёт новые МФ. Анализатор компонует несколько вариантов МФ и выбирает победителя (рис. 6).

1С:Предприятие - Time to talk!									
Документы Мыслеформа: 9 февраля 2016 г.									
Sort order	Base	Дата	Номер	Имя	Тип объекта	Балл	Комментарий		
1	1	09.02.2016 5:53:04	14	Можно ли ходить по дороге	Мыслеформа	317 076			
1	1	09.02.2016 5:53:04	16	Можно ли	Статика	4 032			
1	1	09.02.2016 5:53:04	25	ходить	Статика	455			
1	1	09.02.2016 5:53:04	29	по дороге	Статика	19 382			
76	76	09.02.2016 5:56:11	739	Ходить это одеваться во что - ...	Мыслеформа	472 868 490			
76	76	09.02.2016 5:56:11	740	Ходить	Статика	455			
76	76	09.02.2016 5:56:11	741	это	Статика	468			
76	76	09.02.2016 5:56:11	742	одеваться	Процесс	1 458			
76	76	09.02.2016 5:56:11	743	во что	Статика	8 064			
76	76	09.02.2016 5:56:11	744	-	Статика				
76	76	09.02.2016 5:56:11	745	.	Статика				
76	76	09.02.2016 5:56:11	746	нибудь носить	Статика	18 317			
76	76	09.02.2016 5:56:11	747	что	Статика	162			
76	76	09.02.2016 5:56:11	748	-	Статика				
76	76	09.02.2016 5:56:11	749	нибудь иметь и какойнибудь внеш...	Статика	505 530 590			
76	76	09.02.2016 5:56:11	750	-	Статика				
76	76	09.02.2016 5:56:11	765	вид	Статика	423			
9	9	09.02.2016 5:56:45	817	Ходить это использоваться в ...	Мыслеформа	4 131 871			
9	9	09.02.2016 5:56:45	818	Ходить	Статика	455			
9	9	09.02.2016 5:56:45	819	это	Статика	90			
9	9	09.02.2016 5:56:45	820	использоваться	Процесс	1 386			

Рис. 6. Результат построения МФ. Выиграла МФ №14, т.к. набрала наибольший балл.

Все МФ имеют структуру, отражающую их состав (рис. 7) и для них строятся зависимости между подчиненными объектами (рис. 8). На основании этих данных рассчитывается общий балл МФ. В дальнейшем так же будет происходить поиск уже существующих схожих МФ и наследование их свойств. Сейчас наследование происходит лишь от базовых объектов.

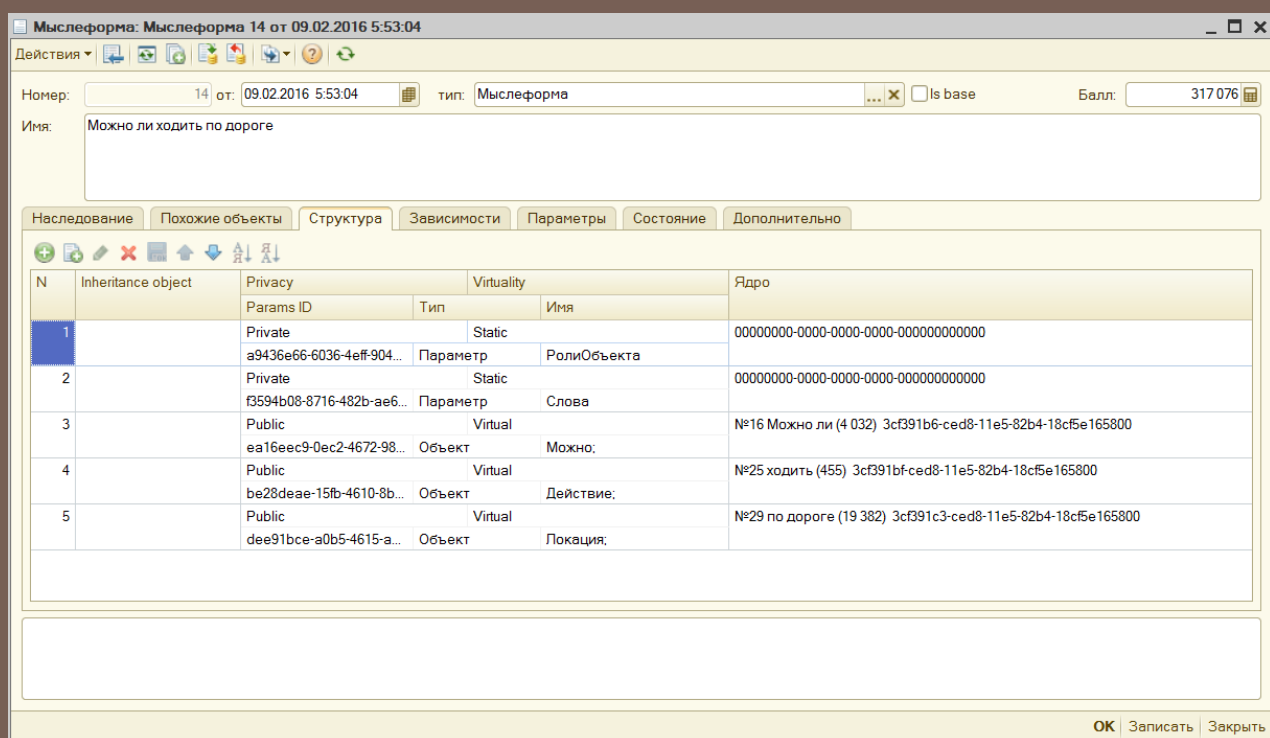


Рис. 7. Структура победившей МФ.

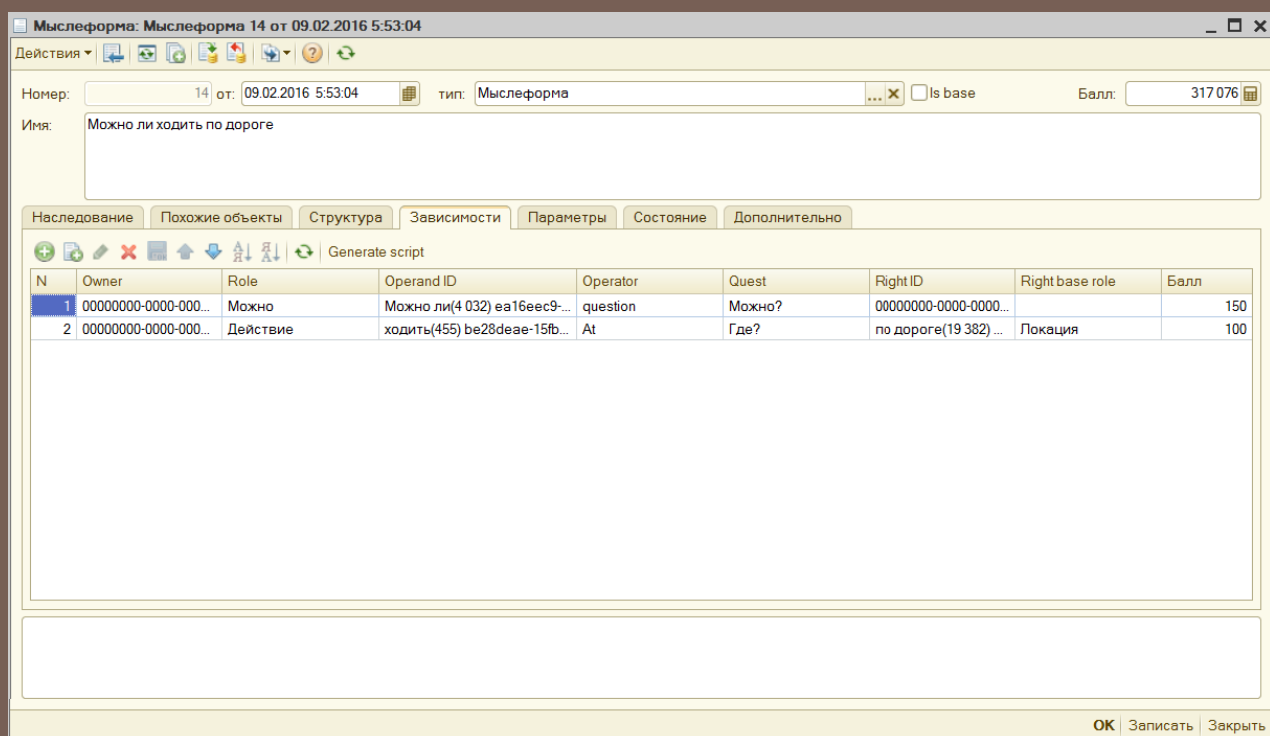


Рис. 8. Зависимости между операндами, рассчитанные для победившей МФ.

Победившая МФ проходит дальнейшую обработку. Для неё формируется скрипт на выполнение (рис. 9). Если требуется ответ пользователю, то он может быть получен в результате. В данном примере ответ требуется и зависит от выполнения оператора at. Если результат сравнения окажется положительным, то пользователю будет возвращен ответ «Можно» иначе его инверсия «not Можно», что после поиска в словаре антонимов будет скомпилировано в ответ «Нельзя».

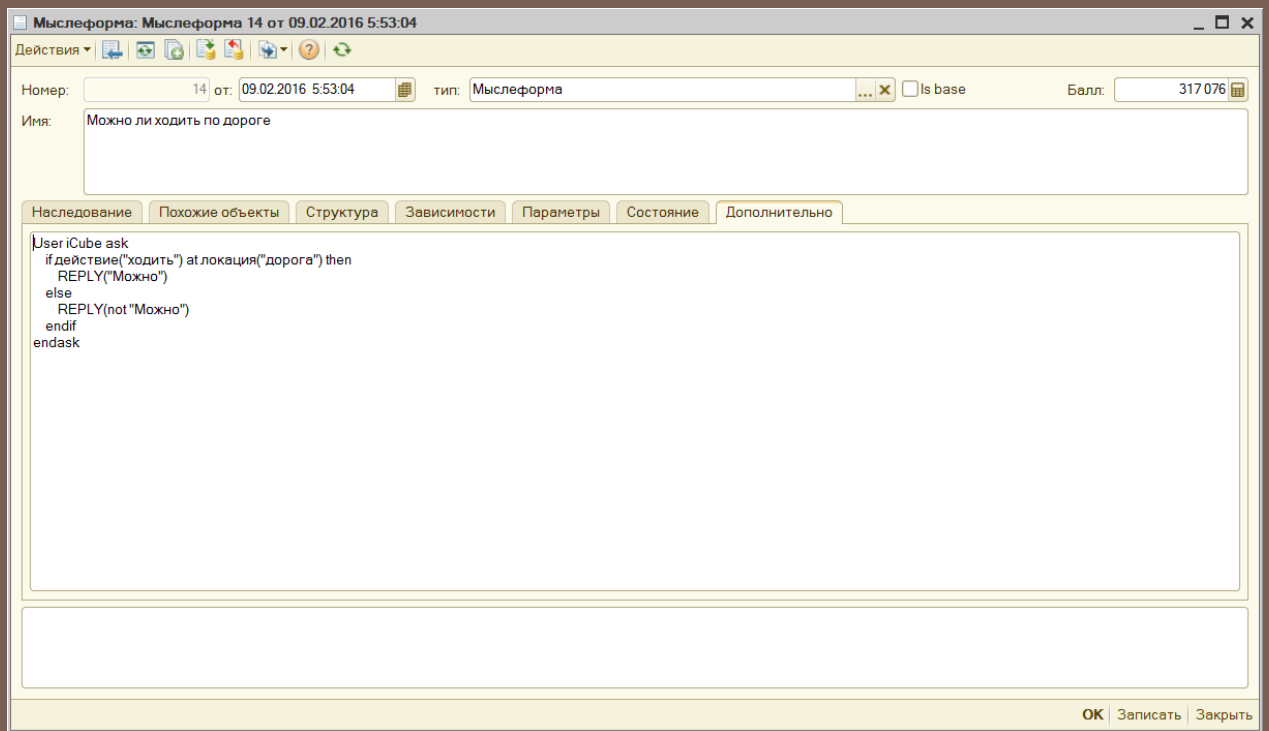


Рис. 9. Скрипт, сформированный для выполнения .

Построенный скрипт выполняется на парсере внутренней грамматики (рис. 10), который преобразует его к алгоритму на языке более низкого уровня (в данном случае внутреннему языку платформы 1С, а в будущем к JavaScript и т.п.). На текущий момент (пока ИИ не может самостоятельно развивать свой язык) для мобильных приложений данный этап можно опустить, сразу генерируя конечный алгоритм.

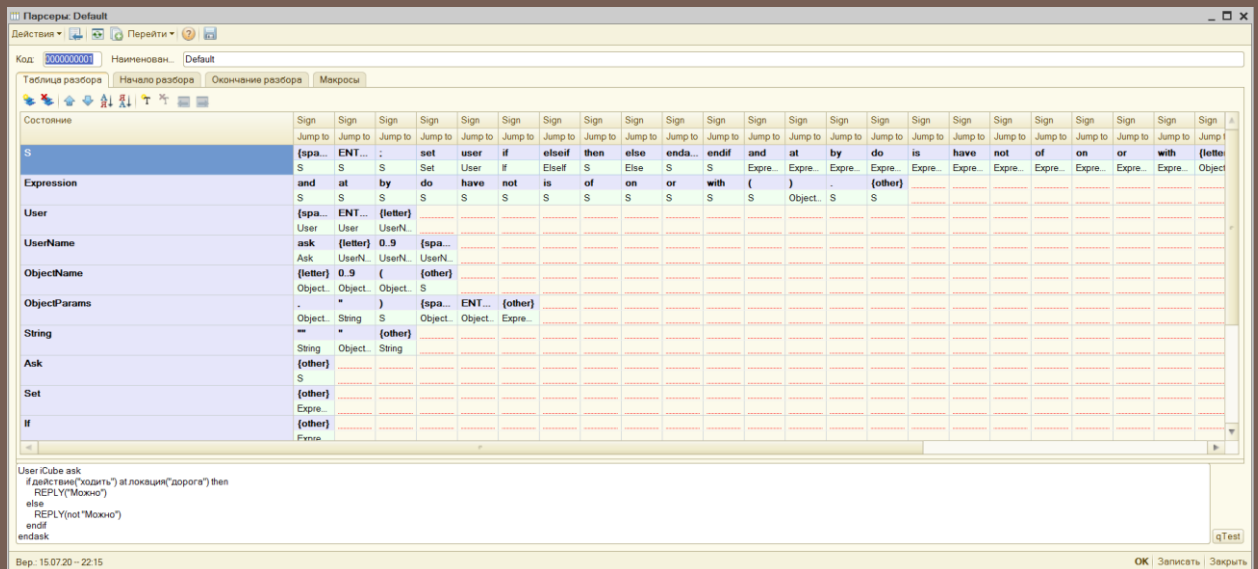


Рис. 10. Парсер «Default».

Результатом компиляции для рассматриваемого примера будет следующий алгоритм (до замены макросов):

```
«UserName = "iCube";
РезультатAsk = "";

    Параметры_действие = Новый Массив;
    Параметры_действие.Добавить("ходить");

&действие
Структура_действие0 = РезультатТекущейОперации;

Результат_at0 = Ложь;

    Параметры_локация = Новый Массив;
    Параметры_локация.Добавить("дорога");

&локация
Структура_локация0 = РезультатТекущейОперации;

Параметры_at = Новый Структура;
Параметры_at.Вставить("OpCode", "at");
Параметры_at.Вставить("ObjectName", "at0");
Параметры_at.Вставить("LeftOperand", Структура_действие0);
Параметры_at.Вставить("RightOperand", Структура_локация0);

&Вычислить_at

Результат_at0 = РезультатТекущейОперации;

Если Результат_at0 Тогда

    Параметры_REPLY = Новый Массив;
    Параметры_REPLY.Добавить("Можно");

    &REPLY
Иначе

    Параметры_REPLY = Новый Массив;

    Результат_not0 = Неопределено;

    Параметры_not = Новый Структура;
    Параметры_not.Вставить("OpCode", "not");
    Параметры_not.Вставить("ObjectName", "not0");
    Параметры_not.Вставить("RightOperand", "Можно");

    &Установить_not

    Результат_not0 = РезультатТекущейОперации;

    Параметры_REPLY.Добавить(Результат_not0);

    &REPLY
КонецЕсли;

&СообщитьПользователю»
```

Как видно из рис. 4, на текущий момент для расчета ответа требуется чуть более часа вычислений на обычном ноутбуке. Основное время уходит на выполнение макроса «&Вычислить_at», так как в этот момент рассчитывается около 40 гипотез толкования терминов «ходить» и «дорога». Текущая версия ядра не поддерживает параллельное вычисление терминов, т.к. это требует дополнительной настройки блокировок при записи общих знаний и разделение оперативных знаний для каждого потока, т.е. оформления механизмов работы с БЗ согласно описанию в предыдущем разделе. И конечно требуется использование более приспособленной для параллельных вычислений серверной платформы.

Получение положительного ответа достигается за счет анализа совместимости левого и правого операнда. В данном случае, одной из гипотез значения термина Дорога является: «полоса земли предназначенная для передвижения», а ключевым словом в определении термина Ходить среди прочих является термин «двигаться», соответственно выделение свойства с ключом назначение у термина Дорога (передвижение) и преобразование этого значения к глагольной форме дает совпадение с левым операндом по ключу и позволяет вычислить результат операции в значении Истина.

Рассмотренный пример является первой успешно пройденной тестовой ситуацией. Его успешное решение открывает путь к выполнению и других подобных ситуаций, например «Можно ли сидеть на кресле?» и т.п. Дальнейшая проработка скриптов позволит расширить диапазон выполняемых ситуаций, а переработка лексического анализатора улучшит понимание нюансов разбираемых фраз. Всё это позволит вывести движок «И³» на следующий уровень по сравнению с программами «чат-ботами» на уровень объектного анализа передаваемой текстовой информации.

По мере накопления достоверных знаний появится возможность опираться на них в вычислении новых результатов, что повысит правильность и скорость ответов на поступающие данные. Постепенно знания оформятся в более сложную древовидную структуру – дерево знаний (ДЗ).

После окончания построения ДЗ станет возможной полноценная работа с тестовым набором фраз, а так же накопление знаний из внешнего мира. Это откроет путь к тренировкам на прохождение «Теста Тьюринга» и окончательному преодолению уровня «чат-бота».

Схема коммерциализации

На текущий момент ближайшим значимым событием бизнес модели проекта является участие в премии Лёбнера (прохождение теста Тьюринга на английском языке). Для участия в этом мероприятии необходимо, чтоб ИИ мог достаточно сносно общаться на английском языке, что предполагает успешное окончание текущего этапа и, возможно, полный рефакторинг кода с переносом наработок на другую платформу (более производительную) с покупкой необходимого оборудования и программного обеспечения. Финансирование этапа предполагается в форме грантов, а длительность этапа составит 2-3 года. Наличие двух и более языков ввода в дальнейшем позволит использовать проект в качестве переводчика, доступ к которому можно будет организовать из интернета для привлечения внимания общественности, подготовки аудитории к следующему этапу проекта и расширенного тестирования.

Далее планируется создание приложения для планшетов и развертывание экспериментальной социальной сети, для массовой проверки технологии. «И³» в них будет применяться в качестве виртуального собеседника-помощника, способного адекватно ответить на вопросы пользователя, помочь ему в поиске необходимого контента (поисковый движок) и просто стать приятным собеседником на отвлеченные темы. Привлечение пользователей сети планируется построить на интересе к теме ИИ и возможности общения с ним. Наиболее проблемным аспектом данного этапа проекта видится морально-этический вопрос возможного общения пользователя с ИИ на темы насилия, наркомании и прочих тем, способных побудить человека принять ошибочное жизненно важное решение. Поэтому, на данном этапе, по-видимому, будет необходима круглосуточная модерация по ключевым словам всех контактов с ИИ и анализ отфильтрованных переписок. Обязательно наличие жесткой установки на не нанесение вреда собеседнику и анализ поведения реальных людей в подобных ситуациях.

Поисковый сервис может быть выделен в отдельное приложение (виджет). В отличие от конкурентов предполагается возможность поэтапного поиска, т.е. в случае получения разнородных данных их анализ, разделение на категории и формирование уточняющих вопросов пользователю, наподобие экспертной системы. Поскольку поисковый сервис требует быстрого нахождения требуемой информации, то этот аспект так же не должен быть забыт.

Для успешного функционирования разнородных приложений требуется выделение сервисной части в отдельную платформу, с которой приложения смогут общаться через общий интерфейс. Наличие подобной платформы так же позволит предоставить её сервис и внешним разработчикам для самостоятельного использования.

Возможно построение и продвижение экспертной системы, способной самостоятельно накапливать и анализировать данные, ведь наличие корректных экспертных знаний в базе является наиболее критичной проблемой внедрения и развертывания подобных систем.

На данном этапе становится возможным финансирование за счет размещения рекламы, в том числе скрытая «ненавязчивая» реклама внутри бесед с ИИ. Так же на данном этапе становится необходимым обучение ИИ пониманию аудиальной и визуальной информации (мультимедиа контента), что откроет новые возможности поиска, например отбор фотографий по словесному описанию. Данный аспект является наиболее сложной технической задачей этапа.

После закрепления положительного результата от социальной сети и/или поискового сервиса планируется начало разработки аппаратно-программного решения класса «система на чипе». Данный этап подразумевает разработку платы расширения для использования в робототехнике, снабженной собственной операционной системой, рассчитанной на управление с использованием ИИ такая система может стать основой будущих дроидов. Так же рассматривается вариант использования системы в качестве автопилота для комплексного управления автомобилем или экзоскелетом. Преимуществом данного решения будут возможность свободного общения человека с машиной, понимание ею всех нюансов текущей ситуации предвиденье и анализ опасностей за счет более «адекватной» и полной расшифровки получаемых данных, точное выполнение указаний водителя.

Анализ рынка

На текущий момент существует множество технологий в той или иной мере использующих ИИ. Но интеллект, способный заменить и превзойти человеческий не существует. Поэтому любые положительные результаты в данном направлении потенциально способны принести прибыль и имеют широкий спектр области применения.

Что касается данного проекта, то в конце текущего этапа предстоит борьба за премию Лёбнера с проектами уже, возможно, побеждавшими на данном конкурсе, и, имеющими высокий уровень подготовки к соревнованию. Это подразумевает необходимость досконального анализа предыдущих соревнований, разбор ошибок и слабых мест участников.

На следующем этапе, при построении социальной сети, проект столкнётся с жесткой конкуренцией с уже существующими сетевыми сообществами. Среди них как зарубежные, так и наши российские аналоги. Привлечение пользователей предполагается построить на интересе общественности к искусственному интеллекту, возможности поэкспериментировать в общении с ним. Так же необходимо будет уделить внимание дизайну и удобству работы с сайтом проекта, проработать политику размещения контента, юридические и правовые аспекты.

Рынок построения автономных роботехнических систем так же имеет своих фаворитов, но пока что конкуренты ещё только готовят свои решения, и в свободной продаже нет аналогов предполагаемому продукту. Но к моменту появления системы они наверняка уже будут, хотя скорее всего ещё не успеют закрепиться в сознании широкой общественности и/или будут предлагаться только в премиальном сегменте. Поэтому, чем быстрее будет произведено развертывание продукта, тем больше шансов захватить приемлемую часть рынка и закрепиться на нем.

Команда проекта

В проекте принимает участие всего только один человек – автор идеи и одновременно ведущий разработчик проекта. Я считаю данный факт конкурентным преимуществом, поскольку благодаря ему исключается недопонимание внутри команды, процесс разработки ускоряется и становится гибче, т.к. исчезают организационные накладные расходы, а общее состояние проекта более прозрачно и интуитивно понятно. Помимо этого сокращаются расходы на содержание (зарплату), что так необходимо на начальной стадии развития.



Я начал программировать где-то с 12 лет. Это было начало 90х и мой первый компьютер. Мне захотелось создать игру наподобие той, что мне нравилось очень играть. Суть игры заключалась в путешествии по лабиринту и сбору сокровищ. Одновременно за игроком гонялось несколько охотников. Охотники всегда двигались по определенному маршруту вперед-назад. В те времена было много вариантов подобной игры, и я захотел создать свой. К моему компьютеру прилагалась книжка с описанием встроенного языка Бейсик. Изучив её, я принялся программировать, всё было просто, пока я не решил писать алгоритмы движения охотников. Простой алгоритм был мне не интересен, а сложный очень долго выполнялся и компьютер «висел».

Проблема управления персонажами игры, а так же фильмы «Терминатор», «Назад в будущее», «Звёздные войны» заставили меня мыслить на тему искусственного интеллекта. Я мог часами думать над этой проблемой. Потом, по мере развития моих навыков в программировании, я стал искать варианты реализации своих идей. Я долго пытался подобрать удобную мне среду разработки, пока в 2005 году, после получения сертификата «1С-Специалист», не решил попробовать реализовать свои идеи на платформе «1С:Предприятие 8.0». Постепенно проект начал развиваться.

К сожалению, в конце 2007 года я столкнулся с финансовыми трудностями, т.к. заключил ипотечный договор с банком и в 2008 устроился на более оплачиваемую, но и требующую от меня много сил и времени работу. К тому же, в 2009 году, у меня родилась дочь, а в 2014 вторая и свободного времени совсем не осталось. Поэтому проект сейчас развивается очень медленно, но мне очень хочется взяться за него с новыми силами и посвятить ему своё основное время.