

МОДЕЛЬ ДИАЛОГА С ЧАТ-БОТОМ В МУЛЬТИАГЕНТНОЙ СРЕДЕ

DIALOGUE MODEL WITH CHAT BOT
IN A MULTIACTIVE ENVIRONMENT

S. Morugin

Summary. The article discusses the principles of conducting a client's dialogue with a bot based on the methodology of the multi-agent approach. Each agent is a mini-expert system capable of conducting a dialogue in its own part of the subject area.

The purpose of the study is to determine the general model of a dialogue with a multi-agent bot and the principles of agent interaction during a dialogue in the conditions of changing the topic of the dialogue and the unrecognizability of some of the client's responses.

The article considers a recurrent scenario-situation model of an agent and formulates requirements for agents to conduct a dialogue.

Dialogue management from the controller's side is reduced to the choice of an agent that is best able to continue the dialogue based on its assessment of the client's responses or requests.

The Markov model of the dialogue and the activation of observer agents allow the topic of the dialogue to be changed without losing context and taking into account the user's goals specified in previous responses.

Keywords: Multi-agent approach, software agents, situational scenario approach, dialogue bot, Markov models.

Моругин Станислав Львович

Доктор техн. наук, доцент,
Нижегородский государственный технический
университет им. Р.Е. Алексеева
smorugin@yandex.ru

Аннотация. В статье рассмотрены принципы ведения диалога клиента с ботом, основанным на методологии мультиагентного подхода. Каждый агент является мини-экспертной системой, способной вести диалог в своей части предметной области.

Цель исследования — определить общую модель диалога с мультиагентным ботом и принципы взаимодействия агентов при ведении диалога в условиях смены темы диалога и нераспознаваемости части ответов клиента. Рассмотрена рекуррентная сценарно-ситуационная модель агента и сформулированы требования к агентам для ведения диалога.

Управление диалогом со стороны контроллера сводится к выбору агента, который наилучшим образом, исходя из оценки своего состояния по отношению к ответам или запросам клиента, способен продолжать диалог.

Марковская модель диалога и активизация агентов-наблюдателей позволяет менять тему диалога, не теряя контекста и учитывая цели пользователя, указанные в ранее данных ответах.

Ключевые слова: мультиагентный подход, программные агенты, ситуационно-сценарный подход, диалоговый бот, Марковские модели.

Введение

Для канала предложения услуг, продаж и маркетинга через сайты и мессенджеры постоянно увеличивается. Чат-боты являются инструментом для такого канала. Продвижение клиентов в режиме диалога с чат-ботом имеет ряд существенных преимуществ, поскольку бот может оперативно перестраивать сценарий диалога, используя данные о клиенте, полученные от внешних источников и в процессе диалога. Персонализация клиентов вместо ранее используемой сегментации повышает эффективность продаж за счет попадания в потребности клиента и большей убедительности аргументов, что его проблема будет решена.

Обычный бот отвечает по заданному сценарию и теряется при смене темы или нестандартных вопросах, что требует обращения к оператору.

Боты с элементами искусственного интеллекта (ИИ боты) понимают слова и контекст. Одним из самых развитых искусственных интеллектов является ChatGPT способный решать разные задачи, в том числе написание кода, создание текстов, перевода текстов, получения точных ответов и использование контекста диалога для

ответов [6, с. 458]. Ограничениями даже для таких сложных решений являются: фактическая точность ответов, склонность к переобучению [1, с. 5] и отсутствие возможности целевого разговора [2, с. 9], под которым понимается способность ИИ общаться на определённую тему и контролировать контекст диалога с собеседником.

В статье рассмотрим принципы ведения диалога ботом, основанным на мультиагентном подходе [3, с. 6] и на подходе экспертной системы [8, с. 1].

Такой бот представляет собой набор вопросно-ответных систем (агентов). Агенты отвечают на вопросы клиента, используя каждый свою базу знаний или часть общей базы, учитывают контекст диалога. База знаний для каждого агента уникальна и специализирована, но в силу ограниченности предметной области агента достаточно проста, её формированием занимается инженер. В качестве модели знаний используют дерево решений, продукционные модели и фреймы [4, с. 7].

Бот — часть общей системы, набор средств для средств ведения диалога и предоставления информации пользователю. Кроме бота в системе есть база данных по объектам предметной области (например, по фи-

нансовым инструментам), база знаний в виде решающих правил ведения диалога, словарь в виде набора доменов, средства обновления информации, накопления статистики и др.

Цель исследования — определить общую модель диалога с мультиагентным ботом и принципы взаимодействия агентов при ведении диалога в условиях смены темы диалога и нераспознаваемости части ответов клиента.

Модель диалога с ботом

Рассмотрим диалог, который ведется с ботом в письменном виде на естественном языке.

Клиент (КЛ) пишет в адрес бота (БТ) высказывание (набор фраз) и далее отвечает на вопросы и выражает свои пожелания. Обозначим:

$W_1, W_2 \dots W_n$ — наборы высказываний (ответов, предложений, справок, вопросов) КЛ;

$V_1, V_2 \dots V_n$ — наборы высказываний (вопросов, фраз, слов, данных) БТ.

Обозначим $R = \{RV, RW\}$ — результат диалога, который включает (подтверждает) подмножество ранее сделанных высказываний и формируется ботом в конце диалога.

Диалог (D) — это последовательность высказываний КЛ и БТ

$$D = \{V_1, W_1, V_2, W_2 \dots V_n, W_n \dots R\}. \quad (1)$$

Немного более общая форма диалога может включать несколько высказываний с одной из сторон подряд

$$D = \{V_1, W_1, V_2, V_3, W_2 \dots V_{n-m} \dots V_n, W_n \dots R\},$$

но её рассматривать не будем в силу того, что она приводится к первой форме диалога.

Цель диалога для БТ — получить результат диалога $R = \{RV, RW\}$ с наиболее высоким рейтингом (баллами, оценкой).

Цель диалога для КЛ — получить информацию для решения своих проблем, определиться со способом и порядком их решения. Иными словами, оценка состояния модели КЛ, выполненная со стороны системы (бота) и в её интересах, должна повыситься. Бот должен так вести диалог, чтобы КЛ закончил диалог удовлетворенным, исходя из оценки системы.

Пример. Формально последние высказывания могут звучать так

«КЛ — я согласен с условиями кредита

БТ — договорились, высылаю Вам документы для оформления

КЛ — да, жду»

В $R = \{RV, RW\}$ включаются все ранее сделанные высказывания и формулировки, которые относятся к результату диалога.

Клиент может провести с ботом несколько диалогов. Модель пользователя M до начала диалога $n+1$

$$M_{in} = \{M_0, FR_1, \dots, F_n R_n\}$$

где M_0 — усредненная начальная модель пользователя, построенная на основе типовых решений и общей статистики диалогов,

$FR_1, \dots, FR_n$ — отфильтрованные результаты предыдущих диалогов,

F — обозначение фильтра состояния пользователя (например, если клиент полгода не звонил, и его данные теряют актуальность, например, в части суммы инвестирования, степени заинтересованности и др.)

Модель клиента после окончания диалога с номером $n+1$

$$M_{out} = \{M_0, FR_1, \dots, F_n R_n, R_{n+1}\}$$

Ситуационно-сценарный подход ведения диалога

Ситуация характеризуется набором параметров состояния активных агентов и общедоступных параметров системы [3, с. 96; 4, с. 167]. На основе обработки параметров ситуации (состояния системы) бот выбирает очередной шаг сценария диалога.

Обозначим:

k — номер шага (например, шаг включает вопрос и ответ),

s_k — состояние системы (совокупность состояний всех активных объектов) на шаге k ,

r_k — распознанный ответ клиента,

q — теоретико-множественный функционал формирования вопроса к КЛ,

f — теоретико-множественный функционал перехода от одного состояния к другому,

h — теоретико-множественный функционал распознавания ответа КЛ по его исходному высказыванию W ,

$A()$ — функция активации объектов по состоянию s_{k+1} ,

δ_k — вектор признака активных объектов на шаге k , содержит 0 (объект не активен) или 1 (объект активен), также этот вектор определяет порядок объектов в очереди на обработку данных от КЛ.

Рекуррентная сценарно-ситуационная модель агента выглядит так

$$\begin{aligned} V_{k+1} &= q(s_k, \delta_k), \\ r_{k+1} &= h(W_{k+1}), \\ s_{k+1} &= f(s_k, \delta_k, r_{k+1}), \\ \delta_{k+1} &= A(s_{k+1}, \delta_k). \end{aligned} \quad (2)$$

Модель (2) обладает марковским свойством [11, с. 1530] и позволяет моделировать псевдослучайно изменяющиеся системы [9, с. 678]. В Марковской модели предполагается, что будущие состояния диалога $s_{k+1}, \delta_{k+1} \dots$ зависят только от текущего состояния s_k, δ_k , а не от событий, которые произошли до него, что используется при управлении такими системами [10, с. 20].

Реализовать универсальный бот в виде одного агента, который отвечал бы на все возможные вопросы КЛ и компетентно предоставлял необходимую информацию в разных предметных областях весьма сложно. Это возможно на основе больших языковых моделей, которые, однако они будут проигрывать специализированным средствам, настроенным на конкретные предлагаемые продукты.

Рассмотрим мультиагентный подход [6, с. 460], когда диалог ведется за счет множества агентов, специализированных в отдельных областях (например, кредиты, инвестиции в акции, облигации, фонды и др.). Каждый специализированный агент может выдавать КЛ точную, актуальную и убедительную информацию в своей области. Мультиагентная среда эквивалентна привлечению к диалогу с клиентом различных специалистов помимо менеджера по общим вопросам.

Пусть в системе существует множество агентов, которые достаточно развиты с точки зрения автономности и достижения своих целей. Активные агенты стоят в очереди на право задать вопрос и обработать ответ КЛ.

Первый из очереди активных агентов формирует вопрос V_{k+1} к КЛ. КЛ дает ответ W_k . Порядок вызова агентов для обработки ответа и список активных агентов определяется вектором δ_k .

Состояние s_{k+1} на шаге $k+1$ определяется состоянием активных агентов s_k на предыдущем шаге k и ответом

КЛ W_{k+1} . В зависимости от совокупного состояния s_k активных агентов, контроллер агентов принимает решение об активизации новых агентов для шага $k+1$ и об изменении порядка агентов в очереди.

Требования к агентам для ведения диалога

Под агентом в работе будем понимать программный (в перспективе еще и аппаратный) комплекс, который характеризуется рядом свойств [2, с.396].

Таблица 1.

Требования к агентам и их реализация в диалоговом боте

Требование к агенту [2, с. 396]	Возможная реализация агента в диалоговом боте
Четко определенная миссия в системе	Роль или миссия каждого агента системы диалога определена
Множеством сформированных целей	Цели агентов заданы, каждый агент решает свои задачи. Критерии достижения целей здесь нестрогие в силу взаимодействия с КЛ. Кроме формального результата, что требует сбора статистики, нужна удовлетворенность КЛ, оценка отношения КЛ
Наличие математической или эвристической модели функционирования	Модель диалога достаточно строгая математически и алгоритмически. Разные задачи требуют разного математического аппарата
Детерминированный алгоритм работы	Алгоритмы диалога работают на основе локальных целевых функций, но могут содержать элементы стохастики для разнообразия диалога
Информационное обеспечение реализации агентного алгоритма работы	Все исходные данные формализованы, могут храниться в базе данных, XML — файлах и др. Формализация в виде набора параметров, доменов и др.
Организационное обеспечение (например, поддержка процесса разрешения конфликтов между агентами)	В процессе текстового общения почти не возникает проблем несовместимости данных. Агенты выстраиваются в очередь на обработку ответов пользователя, что определяет приоритеты обработки
Высокая степень автономности — способностью контролировать свое состояние и действовать без вмешательства оператора	Бот полностью автономен, вмешательство оператора не требуется. Агенты автономны, каждый из них реализует диалог в определенной области

Возможность взаимодействия с другими агентами и обслуживаемой системой	Агенты могут взаимодействовать через обработку ответов, меняя свое состояние. Предусмотрен механизм обмена параметрами агентов родственных классов (Конкретный кредит и Запрос на кредит, например). Разнородные агенты не должны напрямую взаимодействовать, только через контроллер и общедоступные параметры (системные переменные)
Чувствительность, т.е. способностью адекватно воспринимать окружающую среду и реагировать на ее изменения	Среда определяется ответами пользователя, параметрами ответов КЛ и оценкой его состояния. Агенты могут менять общедоступные параметры среды, читать и использовать их
Возможность вмешиваться в работу бота (активное состояние агента)	Решение о продолжении диалога принимается контроллером агентов в составе БТ по состоянию активных агентов. Это делается на каждом шаге и явно входит в модель
Наличие универсальной теоретической основы, которая не меняется в течение жизненного цикла агента и не зависит от предметной области использования агентов	Основа ориентирована на диалог с КЛ, но от предметной области диалога почти не зависит

Таким образом, предлагаемые части бота можно считать агентами в определении [2, с. 396]. Вместе с тем есть отличия в постановке задач для диалогового бота по сравнению с методологиями и сценариями проектирования автоматизированных систем.

Процесс управления диалогом

Представим себе, что все ответы КЛ однозначные, интерпретируемые и даны в терминах модели диалога (агентов и объектов). Однозначности ответов можно добиться, если КЛ выбирает ответы переключателем на форме. Тогда алгоритм диалога будет организован так:

- ведущий агент БТ задает вопрос. КЛ дает один из базовых ответов,
- ведущий агент что-то предлагает и задает следующий вопрос, сформулированный на основе свойств агентов. КЛ дает один из базовых ответов...,
- КЛ или БТ предлагает подвести итоги,
- БТ формирует Результат.

Если КЛ выходит за рамки базовых ответов агента, что имеет место при ответах или вопросах КЛ в свободной форме, то ответ КЛ может обрабатываться очередью агентов по следующим правилам ведения диалога.

- 1) Если КЛ дает один из базовых (заранее известных) ответов в пределах модели диалога ведущего агента, то ответ обрабатывает ведущий агент и он продолжает диалог. Если ведущий агент решил, что диалог с ним исчерпан, то контроллер передает управление другим агентам.

Каждый активный агент, наблюдая за ответом КЛ, параллельно с ведущим обрабатывает ответ при необходимости меняет свое состояние сам, выступая в роли наблюдателя. Он может выставить признак результата обработки («я не нужен», «я удовлетворен», «не понял ответ» и др.). Контроллер агентов считывает эти компоненты состояния и принимает решение о назначении агента ведущим, продолжении активации агента, его деактивации, удалении из диалога или активации новых агентов.

- 2) Все вопросы к КЛ генерируются методами ведущего агента на основе его текущего состояния и могут касаться только параметров входящих в него объектов.
- 3) Ответы на вопросы меняют состояние ведущего агента и, возможно, других агентов. Если ведущий агент не распознал ответ или не может на него ответить, он выставляет соответствующий параметр своего состояния, доступный контроллеру агентов и другим агентам. В таком случае контроллер должен выбрать другого ведущего агента.

Если результаты обработки ответа очередью агентов неудовлетворительны, уточняющий вопрос может быть задан агентом, ориентированным на решение самых общих вопросов.

- 4) На основе оценки состояния агентов контроллер принимает решение о переходе к другому агенту или другой стадии диалога, меняет очередь агентов, активирует новых агентов.

Таким образом, общая структура диалога — это последовательность передачи управления от одного активного агента к другому. Контроллер БТ не вникает в суть вопросов агента, но он следит за его состоянием. У каждого агента n ($n \in 1, \dots, N$) для контроллера и других агентов доступны параметры состояния, в частности K_n — критерий качества ожидаемого ответа агента n .

На основе этих параметров контроллер выстраивает очередь активных агентов, на каждом шаге диалога решая задачу выбора ведущего агента на основе критериев K_n

$$\{K_1, K_2, \dots, K_n\} \rightarrow \max_{n \in 1 \dots N}$$

Выставляемая агентом величина K_n зависит от степени распознавания предыдущего ответа КЛ, наличия

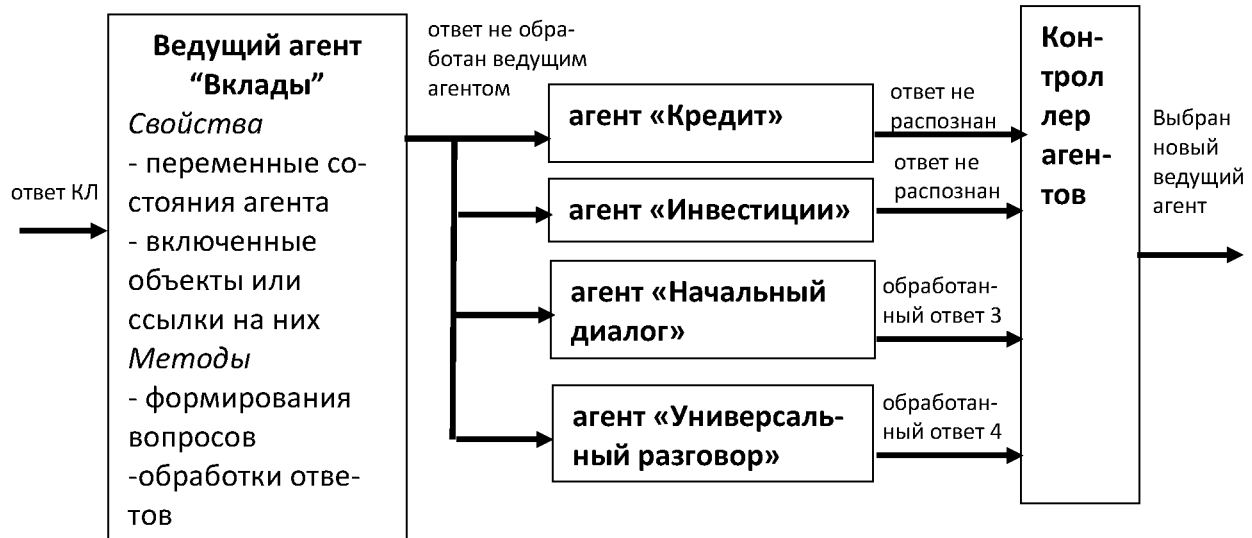


Рис. 1. Структура агента и пример обработки ответа очередью агентов

у агента информации для ответа, накопленной статистики удовлетворения другими КЛ ответами агента на запрос КЛ.

Пример взаимодействия агентов при обработке ответа приведен на рисунке 1.

Взаимоотношение агентов можно описать такими словами:

- есть ведущий диалог агент (первый в очереди). Он получает ответ КЛ и обрабатывает его. Может заявить (выставить параметр K), что «не очень хочет вести диалог, не понял вопроса и т.п.». Далее ответ обрабатывается по очереди другими агентами. Другой агент может «поднять руку» и претендовать на ведение диалога (выставить параметр K , что готов ответить);
- контроллер, исходя из состояния и претензий агентов, степени удовлетворения их имеющимися ответами КЛ может передать управление другому агенту, который и станет ведущим диалог.

Заключение

Предложена структура чат-бота в виде мультиагентной системы, включающей в себя множество специализированных агентов в форме вопросно-ответных экспертных систем. Каждый агент способен вести диалог в своей части предметной области, что упрощает его структуру и повышает качество представления информации клиенту.

Рекуррентная сценарно-ситуационная модель агента является Марковской системой. Сформулированы требования к агентам ведения диалога для их успешного взаимодействия.

Управление диалогом со стороны контроллера сводится к выбору агента, который наилучшим образом, исходя из оценки своего состояния по отношению к ответам или запросам клиента, способен продолжать диалог.

Марковская модель диалога и активизация, кроме ведущего агента, агентов-наблюдателей позволяет менять тему диалога, не теряя контекста и учитывая цели пользователя, указанные в ранее данных ответах.

ЛИТЕРАТУРА

1. Джарратано Джозеф, Райли Гари Экспертные системы: принципы разработки и программирование. / Пер. с англ. — М.: Издательский дом «Вильямс», 2006. — 1152 с. с ил.
2. Мисевич П.В. Агентный подход к созданию и сопровождению автоматизированных систем: Методология, концепции, модели, алгоритмы, примеры (монография) Издательский Дом: LAP LAMBERT Academic Publishing, 2012. — 396 С.
3. Мисевич П.В. Сценарно-ситуационный подход к вопросам проектирования автоматизированных систем // Информационные технологии: труды НГТУ/Нижегород. гос.техн. ун-т.— Н.Новгород, 2005. Т.54. Кн. «Информатика». С.92–97.
4. Мисевич П.В. Сценарно-ситуационный подход к проектированию средств интеллектуальной поддержки процесса функционирования автоматизированных систем // Системы управления и информационные технологии. 2007. N2.1(28). С. 166–171.
5. Gao Leo; Schulman; Hilton, Jacob (2022). Scaling Laws for Reward Model Overoptimization. arXiv:2210.10760
6. Khranilov V.P., Andreev V.V.; Misevich P.V.; E.N. Pankratova. Identification Operators of Dynamic Control Models in the State Space for Engineering Applications Published in: XXI International Conference Complex Systems: Control and Modeling Problems (CSCMP) 01 Sep 2019 (IEEE) — Vol. 2019, pp 455–460

7. Khranilov V.P., Misevich P.V., Pankratova E.N. The Use of Hybrid Knowledge Bases in Designing Engineering Systems Published in: 2020 2nd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA), Lipetsk, Russia, 2020, pp. 479–482.
8. Knox W. Bradley; Stone, Peter. Augmenting Reinforcement Learning with Human Feedback (PDF). University of Texas at Austin. Архивировано (PDF) 8 декабря 2022. Дата обращения: 24 июня 2025.
9. Kochenderfer Mykel J.; Wheeler, Tim A.; Wray, Kyle H. (2022). Algorithms for decision making. Cambridge, Massachusetts; London, England: MIT Press. p. 678. ISBN 9780262047012.
10. Marcus Hutter (2009), Feature Reinforcement Learning: Part I. Unstructured MDPs, Journal of Artificial General Intelligence, 1: 3–24
11. Michael H. Lim, Tyler J. Becker, Mykel J. Kochenderfer, Claire J. Tomlin, and Zachary N. Sunberg (2023). «Optimality Guarantees for Particle Belief Approximation of POMDPs». Journal of Artificial Intelligence Research. 77: 1591–1636.

© Моругин Станислав Львович (smorugin@yandex.ru)

Журнал «Современная наука: актуальные проблемы теории и практики»