

## ЯЗЫК РАЗМЕТКИ ДЛЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

**В. Н. Савицкая**

**Н. П. Шутько**, научный руководитель, канд. техн. наук, доцент

Белорусский государственный технологический университет

г. Минск, Республика Беларусь

Искусственный интеллект и всевозможные чаты с его использованием с каждым днем набирают все большую популярность. Данный доклад посвящен языку AIML.

Artificial Intelligence Markup Language (AIML) – язык на основе XML, который создает чат-ботов и виртуальных помощников. Программа на основе данного языка была создана в 1995 году и называлась A. L. I. C. E. в честь компьютера, на котором она впервые была использована. Alicebot имитировал общение с реальным человеком через Интернет.

A. L. I. C. E. была разработана на основе чат-бота ELIZA. Разработчики первых чат-ботов, таких как ELIZA и Parry, поняли, что тестировщики легко доверяют программе секреты и сокровенные мысли, и чат-боты стали имитировать работу психотерапевта. Благодаря этому A. L. I. C. E., в свою очередь, смогла обрабатывать естественный язык и имитировать реальный человеческий разговор [1]. Пример диалога представлен на рисунке 1.



Рис. 1. Пример общения человека с A. L. I. C. E

В настоящее время AIML не является популярным инструментом для создания чат-ботов. Однако данный язык интересен с точки зрения простоты своего строения и использования.

Язык разметки для искусственного интеллекта работает на основе набора правил и шаблонов для создания ответов на вводимые пользователем данные

[2]. Эти правила написаны в формате XML и используются чат-ботами для создания человекоподобных бесед. Когда пользователь взаимодействует с чат-ботом на основе AIML, ввод его данных анализируется и сопоставляется с соответствующим шаблоном. Затем чат-бот извлекает ответ, связанный с этим шаблоном, и возвращает его пользователю. В данном языке всего 14 тегов [3], этого хватает, чтобы программа могла полноценно функционировать и выдавать корректные ответы на вопросы. Список тегов показан на рисунке 2.

|             |                                                                                                          |
|-------------|----------------------------------------------------------------------------------------------------------|
| <aiml>      | Defines the beginning and end of an AIML document.                                                       |
| <category>  | Defines the unit of knowledge in Alicebot's knowledge base.                                              |
| <pattern>   | Defines the pattern to match what a user may input to an Alicebot.                                       |
| <template>  | Defines the response of an Alicebot to user's input.                                                     |
| <star>      | Used to match wildcard * character(s) in the <pattern> Tag.                                              |
| <srai>      | Multipurpose tag used to call/match the other categories.                                                |
| <random>    | Used <random> to get random responses.                                                                   |
| <li>        | Used to represent multiple responses.                                                                    |
| <set>       | Used to set value in an AIML variable.                                                                   |
| <get>       | Used to get the value stored in an AIML variable.                                                        |
| <that>      | Used in AIML to respond based on the context.                                                            |
| <topic>     | Used in AIML to store a context so that later conversation can be done based on that context.            |
| <think>     | Used in AIML to store a variable without notifying the user.                                             |
| <condition> | Similar to switch statements in a programming language. It helps ALICE to respond to matching the input. |

Рис. 2. Список тегов и расшифровка

Указанные выше программы (ELIZA, Parry и A. L. I. C. E.) проходили тест Тьюринга, однако ни одной из них так и не удалось пройти его успешно. Всегда находились люди, которые во время тестирования понимали, что они общаются с компьютерной программой. Из этого вытекает основной минус данного языка – ограниченность. AIML-боты сильно связаны с заданными шаблонами и не всегда могут адаптироваться к новым сценариям или вопросам. Также AIML не поддерживает сложные алгоритмы машинного обучения, и такие чат-боты могут давать ответы, которые звучат не совсем естественно для человека.

Одним из преимуществ использования AIML является простота и доступность. AIML предоставляет простой и понятный синтаксис для создания правил диалога. Это делает его доступным даже для новичков в области искусственного интеллекта. Также AIML позволяет создавать чат-ботов на разных языках, что полезно для многоязычных приложений.

В свое время AIML был очень полезным и инновационным языком. С его помощью были созданы хорошие и простые чат-боты. Однако для более сложных и гибких решений стоит использовать более актуальные программы.

1. Эволюция чат-ботов: от ELIZA до ChatGPT: [Электронный ресурс]. – URL: <https://habr.com/ru/companies/timeweb/articles/704226/> (дата обращения: 05. 04. 2023). – Текст : электронный.

2. Шовин, В. А. Программа chat-bot – чат-бот или виртуальный собеседник: журнал / Математические структуры и моделирование. – 2016. – № 4(40). – С. 97.

3. AIML TUTORIAL: [Электронный ресурс]. – URL: <https://pema.grg.medium.com/aiml-tutorial-a8802830f2bf> (дата обращения: 05. 04. 2023). – Текст : электронный.

## ПРИМЕНЕНИЕ МАШИННОГО ОБУЧЕНИЯ ДЛЯ РАСПОЗНАВАНИЯ ЛИЦ

*Е. М. Саенко, А. О. Свириденко*

*Я. А. Игнаткова, научный руководитель, ассистент*

Белорусский государственный технологический университет  
г. Минск, Республика Беларусь

В современном мире, где данные и изображения становятся все более доступными и объемными, существует потребность в автоматическом и точном распознавании лиц для различных приложений, таких как системы видеонаблюдения, автоматическая идентификация, улучшение пользовательского опыта и другие. Таким образом, распознавание лиц становится все более актуальной задачей в области компьютерного зрения и искусственного интеллекта.

Применение машинного обучения для данной задачи позволяет создавать системы, способные автоматически обнаруживать и идентифицировать лица на изображениях или видео, что имеет огромную практическую значимость во многих областях, таких как безопасность и видеонаблюдение, биометрическая аутентификация и другие. Для выполнения поставленной задачи создается мощная модель, способная распознавать лица в различных ситуациях. Ее создание – сложная задача, состоящая из нескольких взаимосвязанных этапов. Программные обеспечения не могут делать обобщения на высоком уровне сразу. По этой причине их нужно детально обучать каждому этапу отдельно, чтобы они могли успешно выполнять поставленную задачу: распознавание лиц.

Целью работы было изучить возможности современных языков и библиотек для обучения искусственного интеллекта задаче распознавания лиц.

Python является одним из самых популярных и гибких языков программирования в области машинного обучения. OpenCV – это одна из библиотек, предоставляющих мощные инструменты для обработки различных изображений, анализа и распознавания требуемых объектов. Она позволяет работать с каскадным классификатором – методом машинного обучения, используемым для обнаружения объектов на изображениях или видео. Он основан на исполь-