

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»**

**Программные среды разработки
искусственного интеллекта**

Выполнил:

Студент гр. 3540901/02001

Бараев Д.Р.

Руководитель:

Доцент К.Т.Н.

Бендерская Е.Н.

Санкт-Петербург 2020

Первые языки

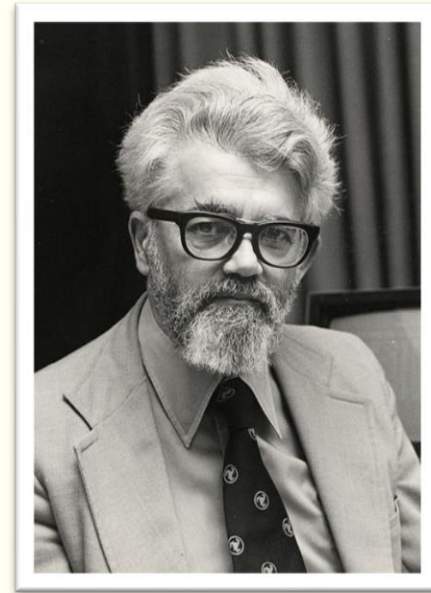


LISP

Появился в 1958

Разработчики Джон Маккарти и Стив Рассел

Традиционный Лисп имеет динамическую систему типов. Язык является функциональным, но начиная уже с ранних версий обладает также чертами императивности, к тому же, имея полноценные средства символьной обработки, позволяет реализовать объектно-ориентированность; примером такой реализации является платформа CLOS.



Джон Маккарти

Автор термина «искусственный интеллект», изобретатель языка Лисп, основоположник функционального программирования, лауреат премии Тьюринга за огромный вклад в область исследований искусственного интеллекта

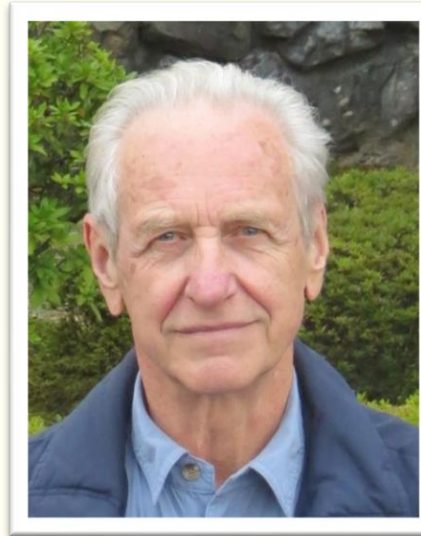
Первые языки

PROLOG

Появился в 1972

Разработчики Ален Колмероз и Роберт Энтони Ковальский

Prolog является декларативным языком программирования: логика программы выражается в терминах отношений, представленных в виде фактов и правил. Для того чтобы инициировать вычисления, выполняется специальный запрос к базе знаний, на которые система логического программирования генерирует ответы «истина» и «ложь». Для обобщённых запросов с переменными в качестве аргументов созданная система Пролог выводит конкретные данные в подтверждение истинности обобщённых сведений и правил вывода.



**Роберт Энтони
Ковальский**



**Ален
Кольмерауэр**

Средства создания систем ИИ

**Общие требования к
современным
инструментальным
средствам создания систем
искусственного интеллекта**

- Специализация
- Использование языков
традиционного программирования
- Интегрированность
- Открытость и переносимость
- Архитектура клиент/сервер

Разработка экспертных систем

Три основные разновидности исполнения экспертных систем:

Экспертные системы, выполненные в виде отдельных программ, на некотором алгоритмическом языке, база знаний которых является непосредственно частью этой программы

Оболочки экспертных систем - программный продукт, обладающий средствами представления знаний для определенных предметных областей

Генераторы экспертных систем - мощные программные продукты, предназначенные для получения оболочек, ориентированных на то или иное представление знаний в зависимости от рассматриваемой предметной области

Языки программирования ИИ



Уровень развитие языков в области ИИ на сегодняшний день— это библиотеки для разработки ИИ, которые в основном требуют контролируемого обучения. Тем не менее такие технологические гиганты, как Microsoft, Facebook и Google, работают над созданием программ, которые будут работать поверх существующих библиотек разработки ИИ, чтобы дать им кроссплатформенность и поддержку самообучения.

Фреймворки и библиотеки для разработки ИИ

Член
команды

Спринт



TensorFlow — это комплексная платформа для машинного обучения с открытым исходным кодом.

Плюсы:

- Берёт на себя оптимизацию ресурсов для вычислений.
- Огромное комьюнити.

Минусы:

- Сложен в использовании и освоении.
- Необходимо постоянно контролировать используемую видеопамять.
- Имеет свои стандарты.

Проекты, которые используют фреймворк TensorFlow:

DeepSpeech — система распознавания речи.

Mask R-CNN — модель, которая генерирует ограничительные рамки и маски сегментации для каждого объекта на изображении.

BERT — предобученная нейронная сеть, используемая для решения задач обработки естественного языка.

Фреймворки и библиотеки для разработки ИИ

Член
команды

Спринт



PyTorch — это среда машинного обучения на языке Python с открытым исходным кодом, обеспечивающая тензорные вычисления с GPU-ускорением.

Плюсы:

- **Имеет множество модульных элементов, которые легко комбинировать.**
- **Легко писать собственные типы слоев и работать на GPU.**
- **Имеет широкий выбор предварительно обученных моделей.**

Минусы:

- **Вам придётся самостоятельно писать тренировочный код.**
- **Плохая документация, то и дело будут попадаться функции и методы, документация которых существует исключительно на форумах сообщества и получена эмпирическим путём.**

Проекты, которые используют фреймворк PyTorch:

PyText — библиотека для обработки устной и письменной речи.

vid2vid — генеративная нейросеть для подмены деталей и свойств видеопотока.

pix2pix — алгоритм, который превращает пользовательские наброски в фотографии.

Фреймворки и библиотеки для разработки ИИ

Член
команды

Спринт



Фреймворк нацелен на оперативную работу с нейросетями и является компактным, модульным и расширяемым. Подходит для небольших проектов.

Плюсы:

- Удобен в использовании.
- Лёгок в освоении.
- Быстроразвивающийся фреймворк.
- Хорошая документация.
- Встроен в TF.

Минусы:

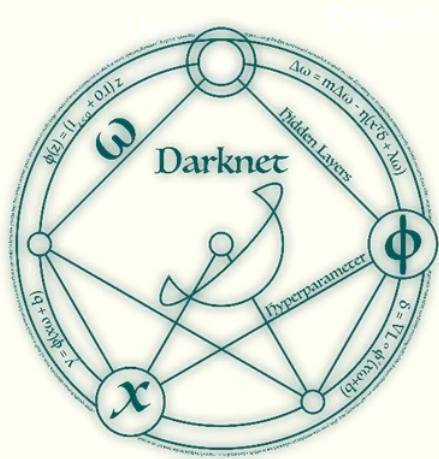
- Не подходит для больших проектов.

Проекты, которые используют фреймворк Keras:

Mask R-CNN — модель, которая генерирует ограничительные рамки и маски сегментации для каждого объекта на изображении
face_classification — алгоритм для распознавания лиц в режиме реального времени и классификации эмоций и пола.

YOLOv3 — нейронная сеть для обнаружения объектов в режиме реального времени.

Фреймворки и библиотеки для разработки ИИ



Darknet — это фреймворк с открытым исходным кодом, написанный на языке C с использованием программно-аппаратной архитектуры параллельных вычислений CUDA. Также Darknet поддерживает вычисления на базе CPU и GPU.

Плюсы:

- Простой.
- Быстрый.
- Удобный.

Минусы:

- Кроме задач с обнаружением больше нигде не используется.
- Не рекомендуется для больших проектов.
- Плохая документация.

Проекты, которые используют фреймворк Darknet:

YOLOv3 — нейронная сеть для обнаружения объектов в режиме реального времени.

Tiny-YOLO 3 — компактная нейронная сеть для обнаружения объектов.

Фреймворки и библиотеки для разработки ИИ



Accord.Net — это фреймворк, основанный на C#, занимающийся развитием нейросетей, используемых для обработки аудио и изображений.

Плюсы:

- **Постоянно обновляется и поддерживается активной командой разработчиков**
- **Стабильная система, которая эффективно обрабатывает огромное количество вычислений и визуализирует их**
- **Крайне удобная работа с алгоритмами и обработкой сигналов**
- **Легко справляется с числовой оптимизацией и искусственными нейросетями**

Минусы:

- **Малоизвестный, по сравнению с другими фреймворками**
- **Производительность гораздо ниже, в сравнении с другими фреймворками**

Сравнение Фреймворков

Framework	Язык	Достоинства	Недостатки
TensorFlow	Python JavaScript C++ Java	<ul style="list-style-type: none"> Берёт на себя оптимизацию ресурсов для вычислений. Огромное комьюнити. 	<ul style="list-style-type: none"> Сложен в использовании и освоении. Имеет свои стандарты. Плохая документация. Необходимо контролировать видеопамять.
PyTorch	Python C++ Java	<ul style="list-style-type: none"> Имеет множество модулей, которые легко комбинировать. Легко работать на GPU. Имеет широкий выбор предварительно обученных моделей. 	<ul style="list-style-type: none"> Плохая документация
Keras	Python	<ul style="list-style-type: none"> Удобен в использовании. Лёгок в освоении. Хорошая документация. 	<ul style="list-style-type: none"> Не подходит для больших проектов.
Darknet	C	<ul style="list-style-type: none"> Простой. Быстрый. Удобный. 	<ul style="list-style-type: none"> Кроме задач с обнаружением больше нигде не используется. Не рекомендуется для больших проектов. Плохая документация.
Accord.Net	C#	<ul style="list-style-type: none"> Постоянно обновляется и разрабатчиками Стабильная система, которая обрабатывает огромное количество вычислений Крайне удобная работа с алгоритмами 	<ul style="list-style-type: none"> Производительность гораздо ниже, в сравнении с другими фреймворками

Заключение

На сегодняшний день тема «Искусственный интеллект» очень популярна. Она является частым предметом разговоров в сфере IT. С развитием интеллектуальных систем развивается и софт для его создания. Благодаря множествам проектов с открытым исходным кодом очень часто появляются разные фреймворки и библиотеки, как и от крупных компаний так и от разработчиков-энтузиастов, которые решают область конкретных задач. Так как тема весьма глубока в изучение, развитие искусственного интеллекта еще находится в начальной стадии. ИИ имеет очень большие перспективы так как уже на сегодняшний день решает весьма сложные задачи. С развитием ИИ будут и развиваться языки программирования для его написания. Будут разрабатываться большое количество библиотек и фреймворков, а также различные вспомогательные инструменты. Во многих IT-компаниях формируются специальные команды и лаборатория по изучению ИИ и разработки разных систем по решению конкретных задач. Поэтому ИИ имеет большое будущее также, как и разработчики в этой сфере и языки, которые используются для разработки этих систем.

Спасибо за внимание !