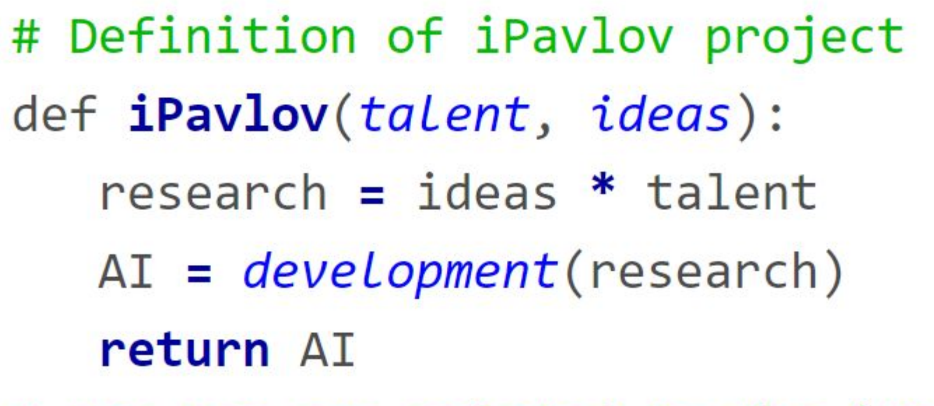
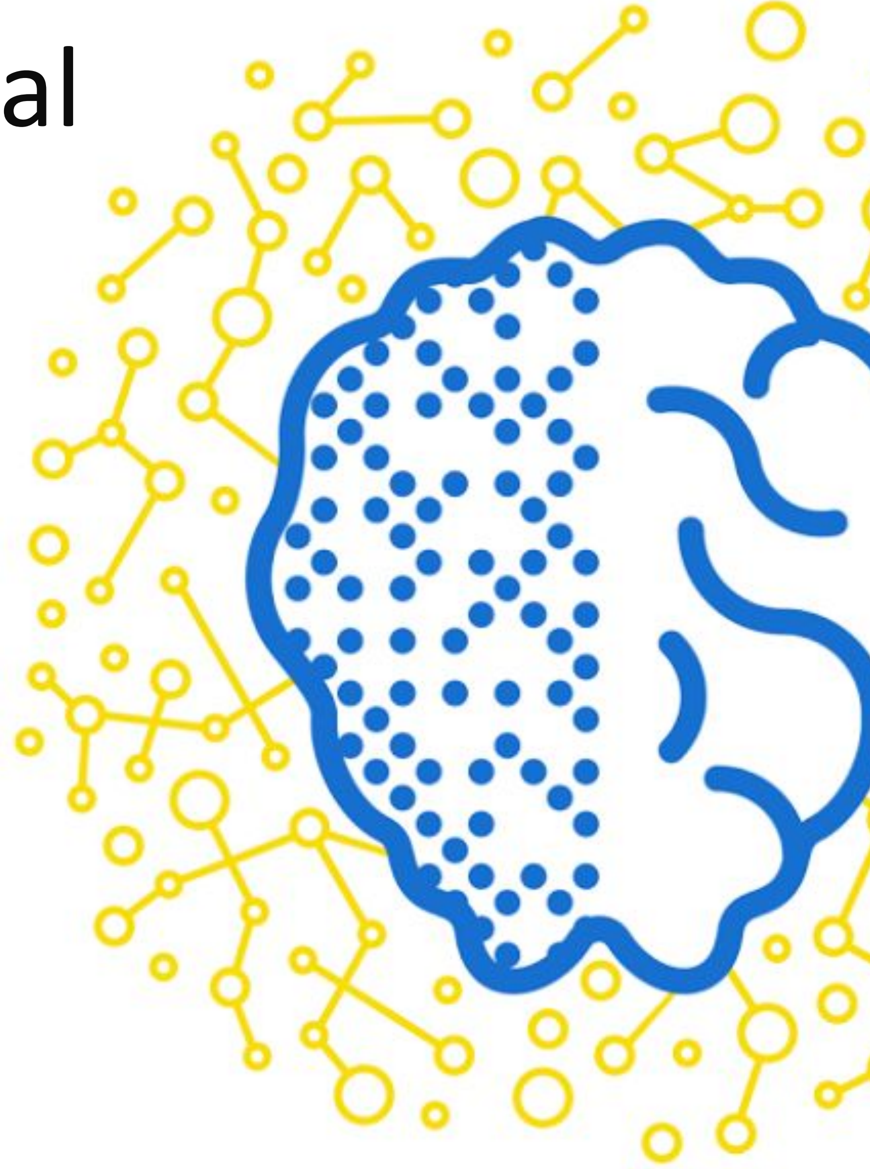
iPavlov: Conversational Intelligence Project

Mikhail Burtsev, PhD

Moscow Institute of Physics and Technology (MIPT)

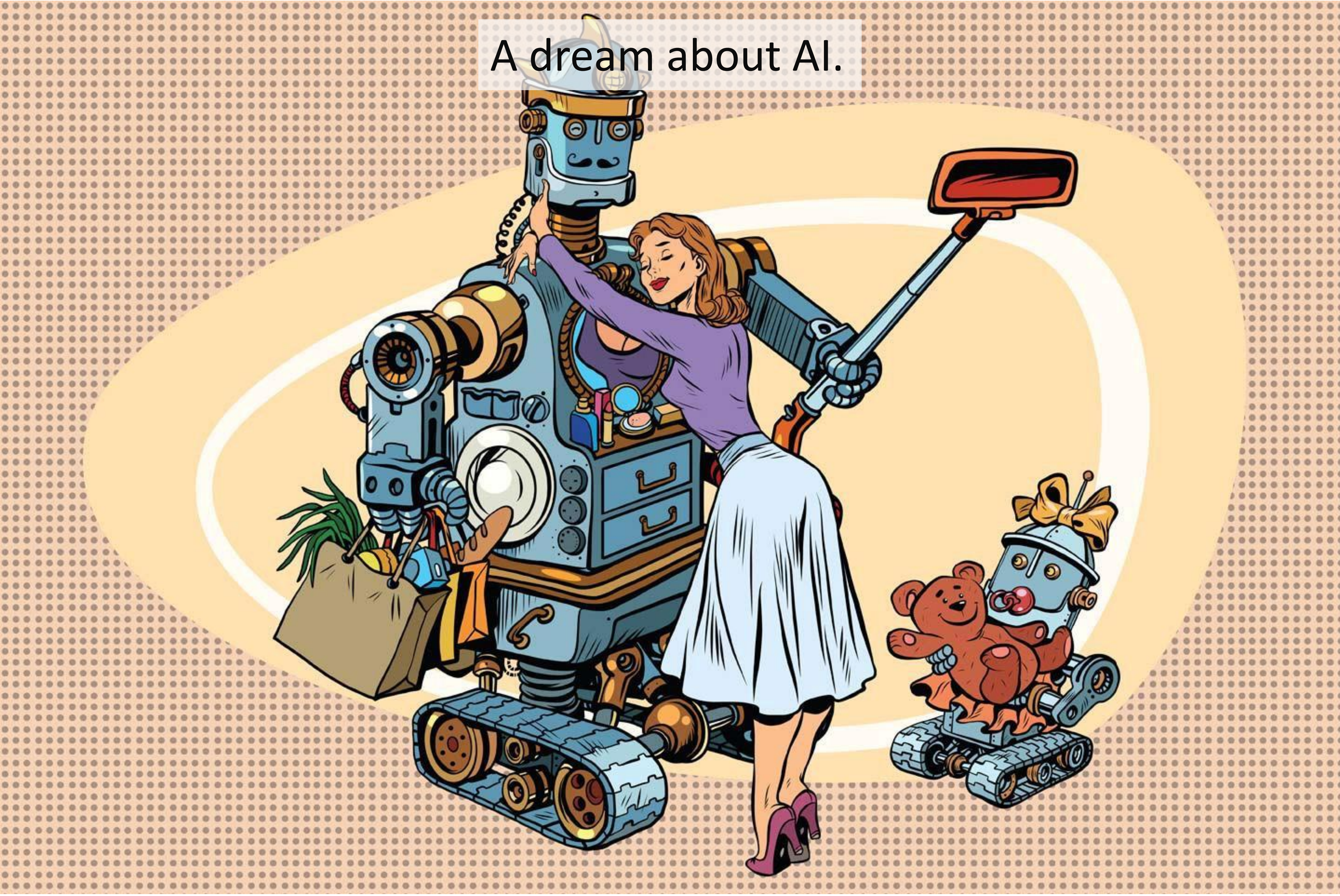


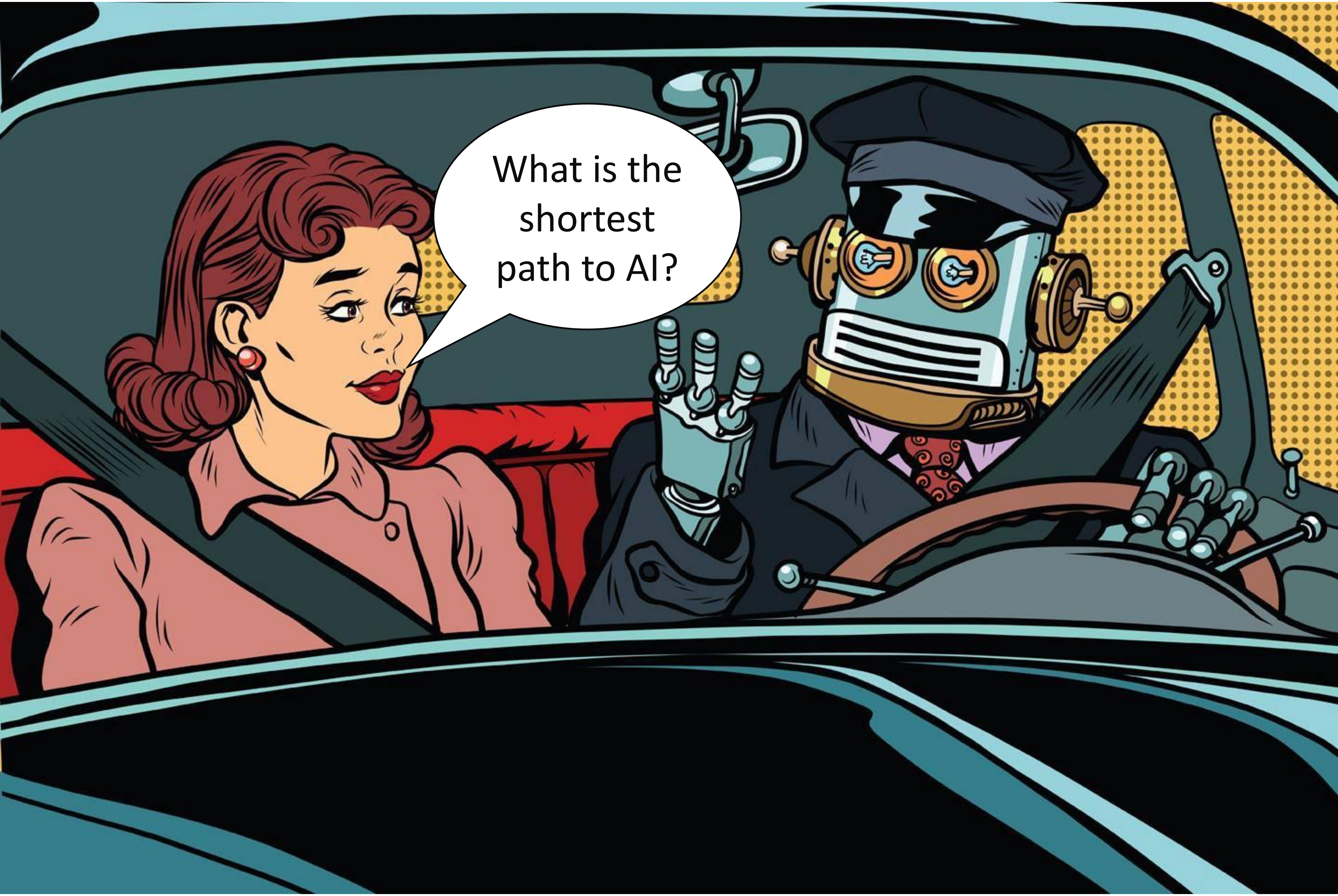
```
# Definition of iPavlov project
def iPavlov(talent, ideas):
    research = ideas * talent
    AI = development(research)
    return AI
```

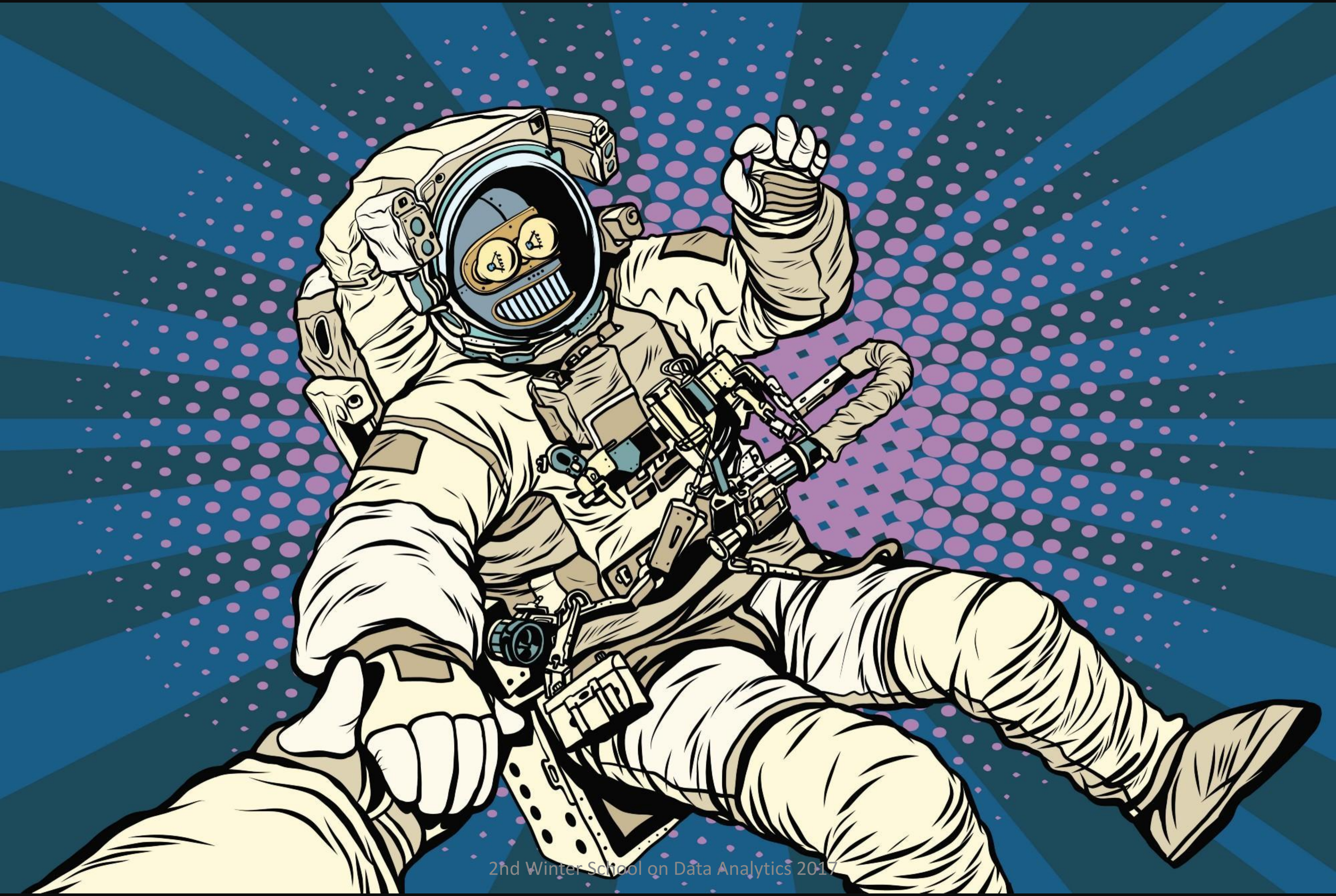


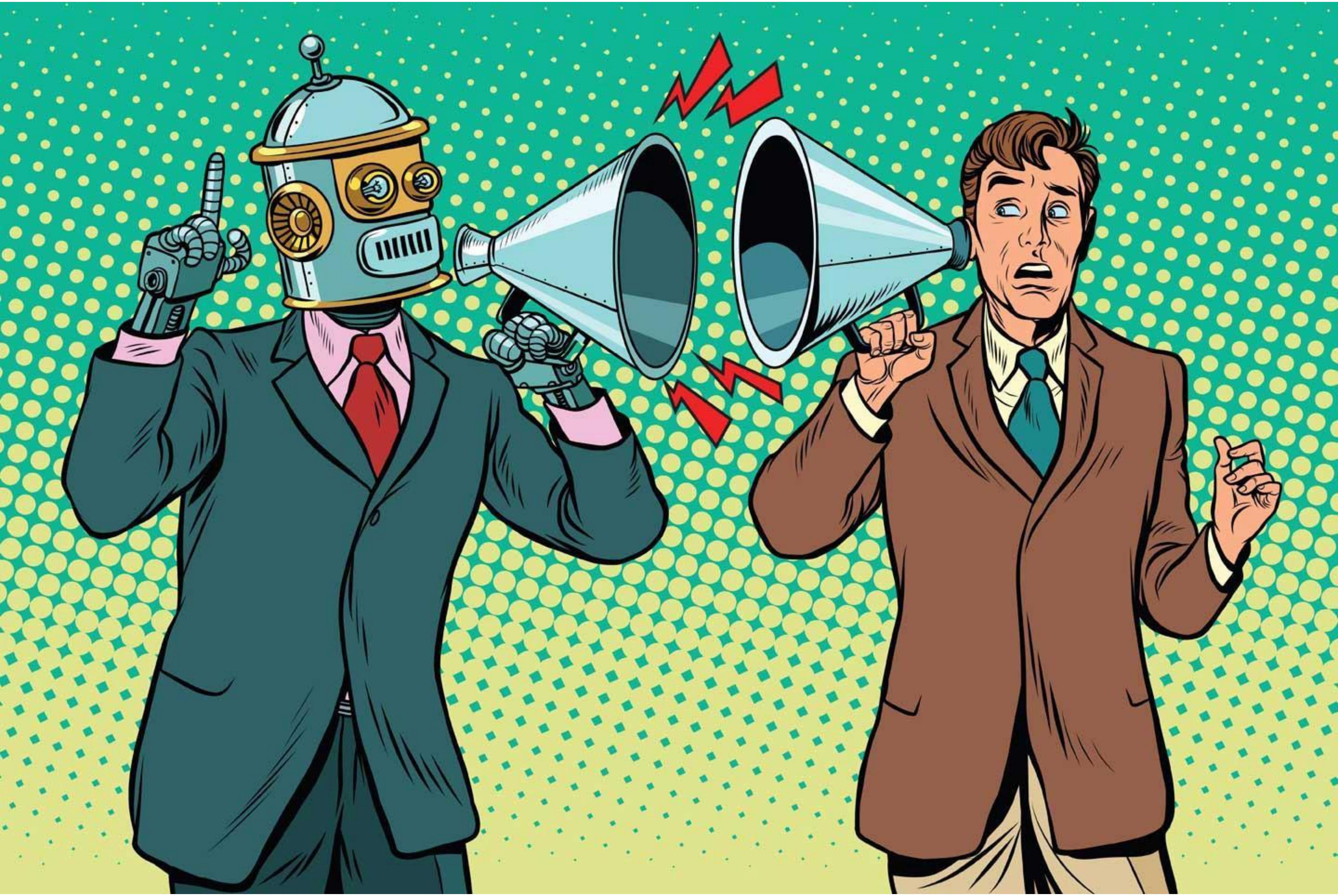
Everybody has a dream

A dream about AI.




What is the shortest path to AI?

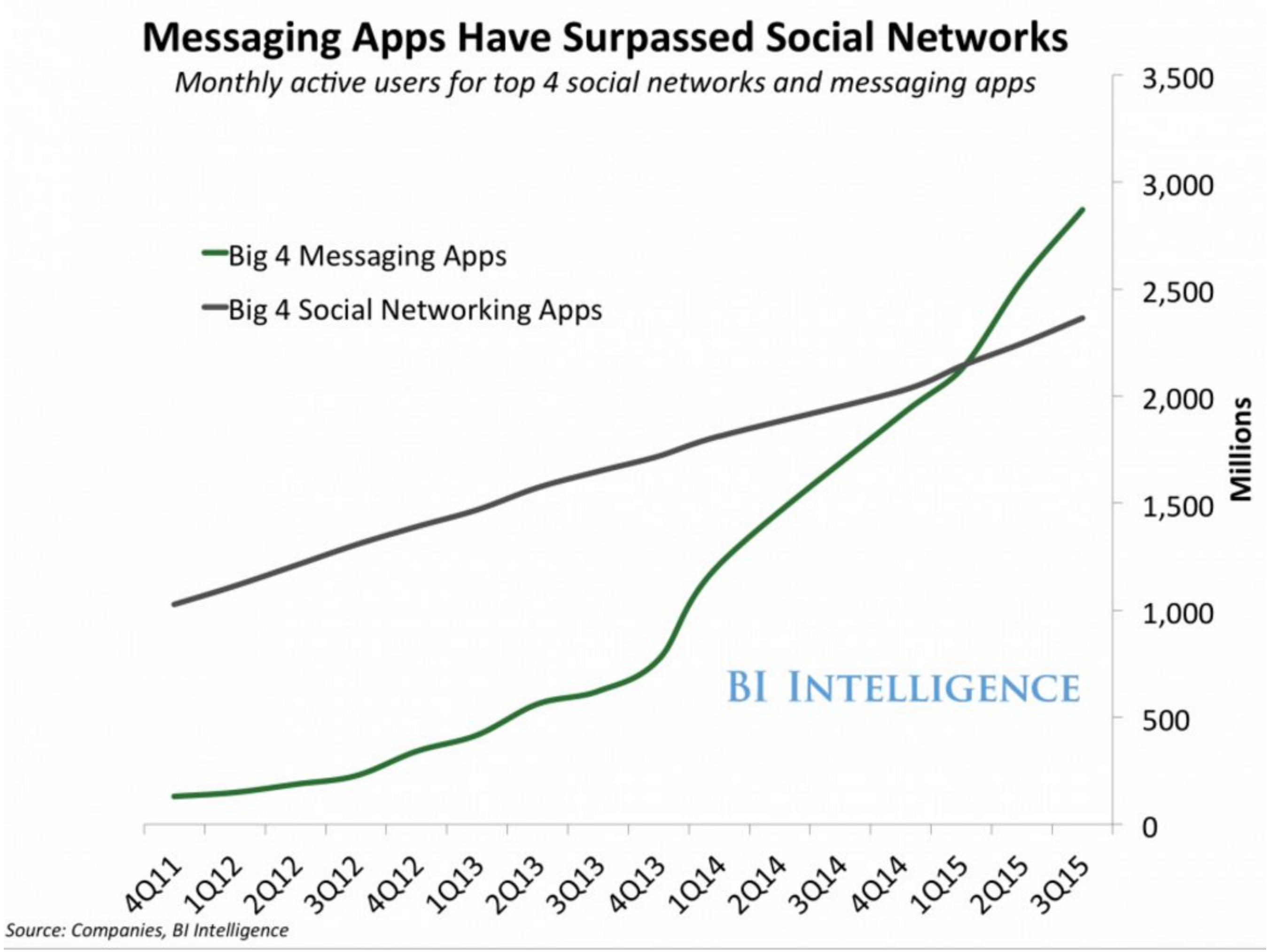




R&D Landscape in Conversational Intelligence

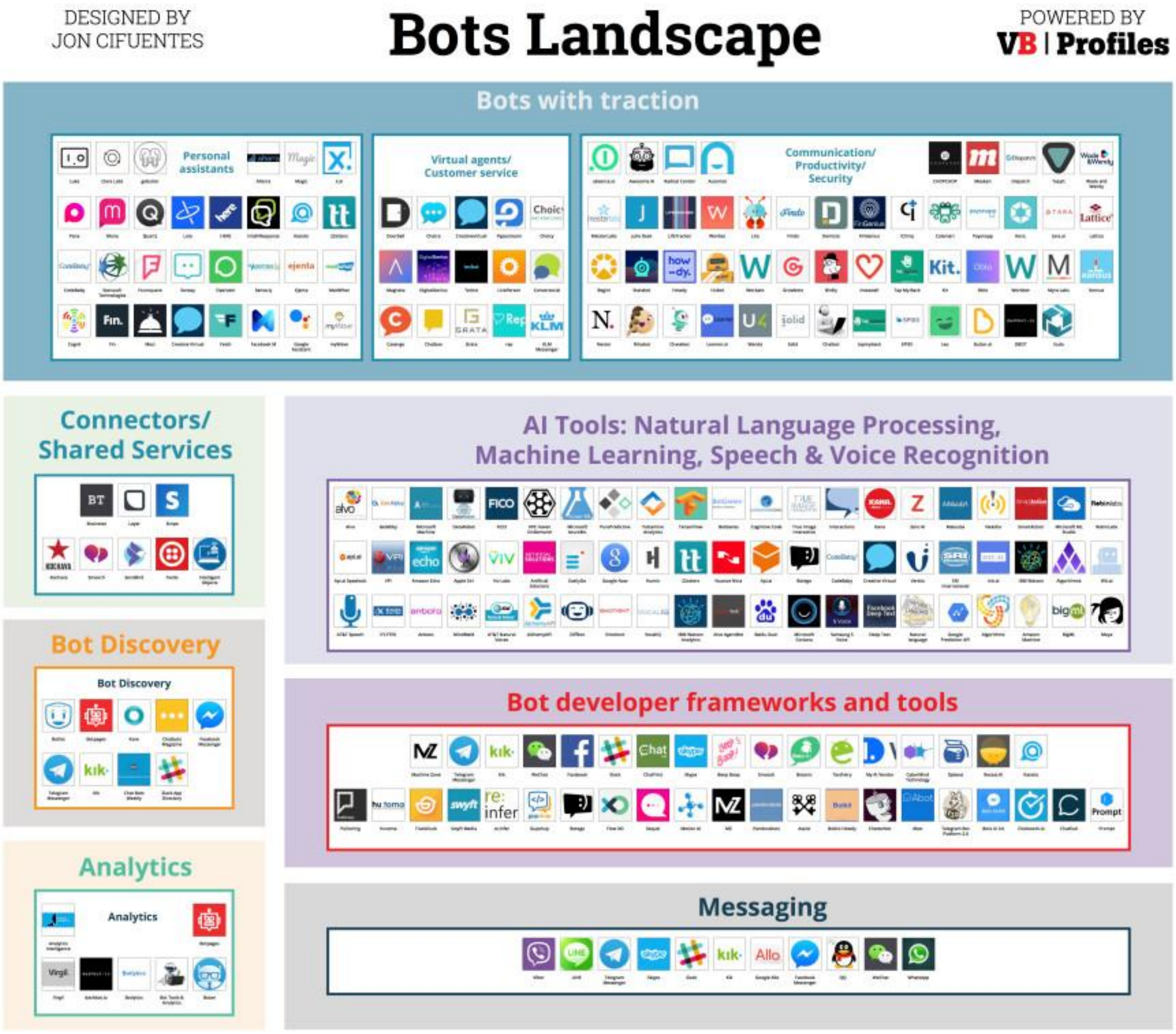
- Conversational Intelligence
  - Complex real world problem
  - Can be decomposed into simpler tasks - NLU, DM, NLG
  - Big amount of data is available
  - Immediate application in industry
  - A step towards solving AI
- Promise of deep learning :
  - recurrent neural networks for the generation of sequences, and
  - attention and reinforcement learning for the dialogue planning .

Textual exchange dominates digital communication



Source: Companies, BI Intelligence

Conversational interface to seamlessly plug in human communication


DESIGNED BY JON CIFUENTES
Bots Landscape
POWERED BY VB | Profiles
Bots with traction
Personal assistants
Virtual agents/ Customer service
Communication/ Productivity/ Security
Connectors/ Shared Services
AI Tools: Natural Language Processing, Machine Learning, Speech & Voice Recognition
Bot Discovery
Bot developer frameworks and tools
Analytics
Messaging

iPavlov project

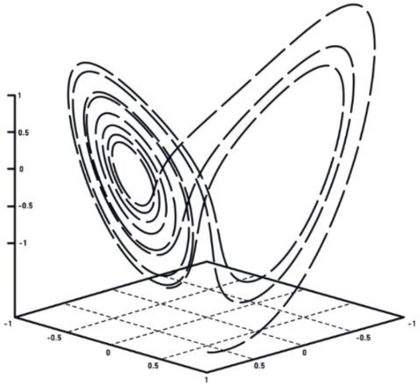
iPavlov project

Deep learning architectures for the conversational intelligence

- The major lab project for the 2017-2019
- Joint project with Sberbank the largest bank in Russia (operating income $20 billion, total assets $400 billion (2014))
- 20 researchers and engineers

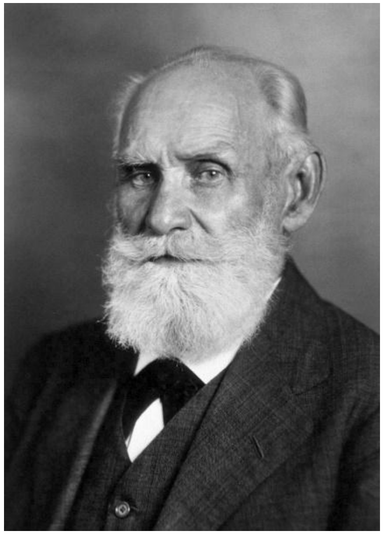
MIPT
MOSCOW INSTITUTE OF PHYSICS AND TECHNOLOGY

SBERBANK
By your side



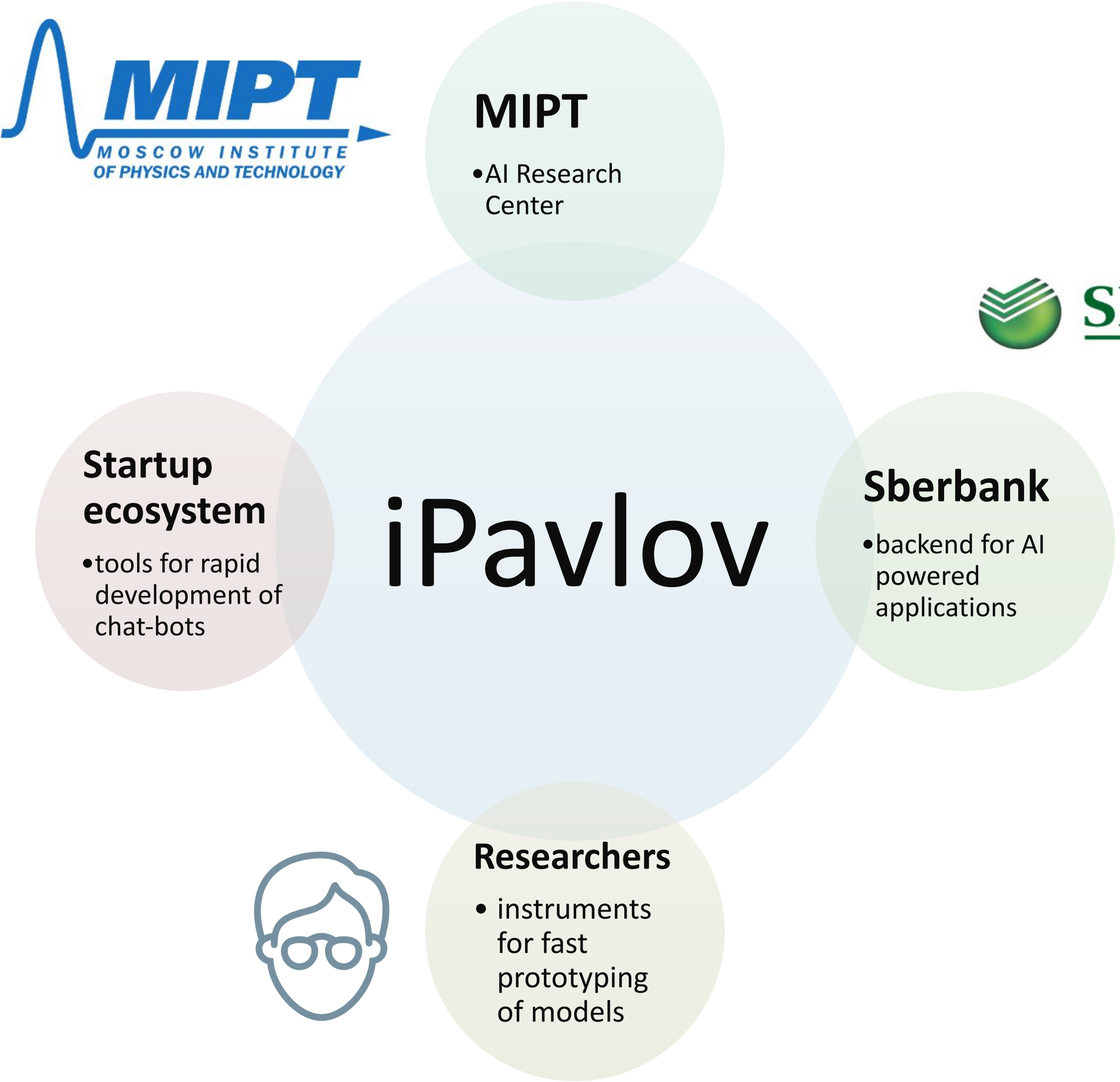
National Technølogy Initiative

Space of possibility



Ivan Petrovich Pavlov
(1849 –1936)
Russian physiologist known for his work in classical conditioning.

Stakeholders



iPavlov

MIPT

- AI Research Center

Startup ecosystem

- tools for rapid development of chat-bots

Sberbank

- backend for AI powered applications

Researchers

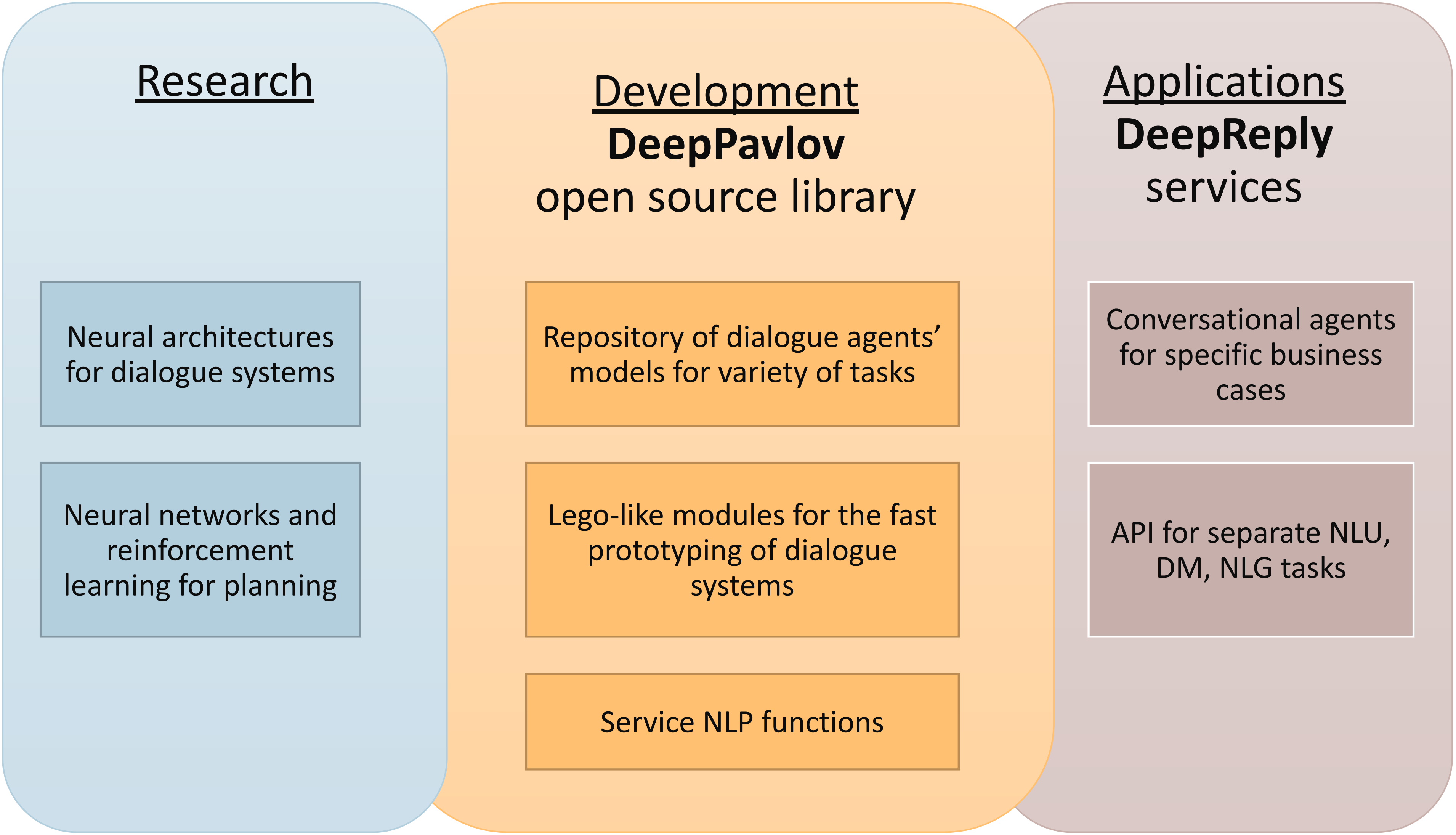
- instruments for fast prototyping of models

iPavlov project

- Technology outcomes
  - Opensource deep learning NLP library **DeepPavlov**.
  - AI platform **DeepReply** implementing NLP services on top of **DeepPavlov** library for the chat-bot and dialogue systems products.

Technology Stack	Project Outcome	Description	Examples
AI APPLICATIONS	Out of the scope of iPavlov project	Third party AI applications in the domain of conversational intelligence.	Google Now, Digital Genius
AI SERVICES	**DeepReply**	AI conversational services to the neural network models trained for specific domains.	API.ai, wit.ai, Google NLP API
DEEP LEARNING ARCHITECTURES	**DeepPavlov**	Core components for neural conversational intelligence. Basic NLP functions and major neuroarchitectures for the dialogue systems.	MemNN, HRED
CORE DEEP LEARNING ALGORITHMS			Seq2seq, CNN, RNN, LSTM
COMPUTATIONAL LIBRARIES	Out of the scope of iPavlov project		ThensorFlow (Google), Torch(Facebook),
DRIVERS GPU/FPGA			C/C++,Python, Julia…
CPU/GPU/FPGA			NVIDIA GPU, Intel CPU, Google TPU

Workpackages



Research

Neural architectures for dialogue systems

Neural networks and reinforcement learning for planning

Development

DeepPavlov

open source library

Repository of dialogue agents' models for variety of tasks

Lego-like modules for the fast prototyping of dialogue systems

Service NLP functions

Applications

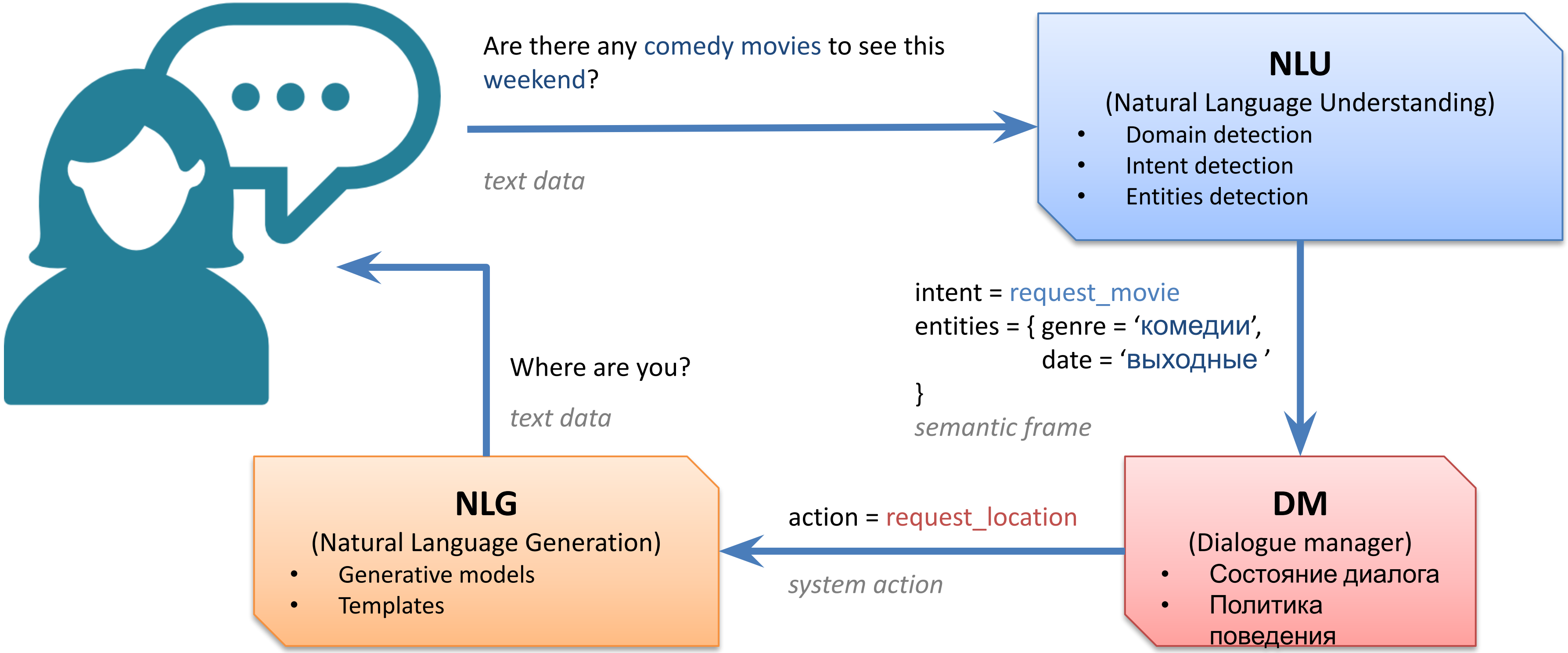
DeepReply

services

Conversational agents for specific business cases

API for separate NLU, DM, NLG tasks

Modular dialog system



Are there any comedy movies to see this weekend?

text data

NLU

(Natural Language Understanding)

- Domain detection
- Intent detection
- Entities detection

intent = request_movie
entities = { genre = 'комедии',
date = 'выходные '
}

semantic frame

DM

(Dialogue manager)

- Состояние диалога
- Политика поведения

action = request_location

system action

NLG

(Natural Language Generation)

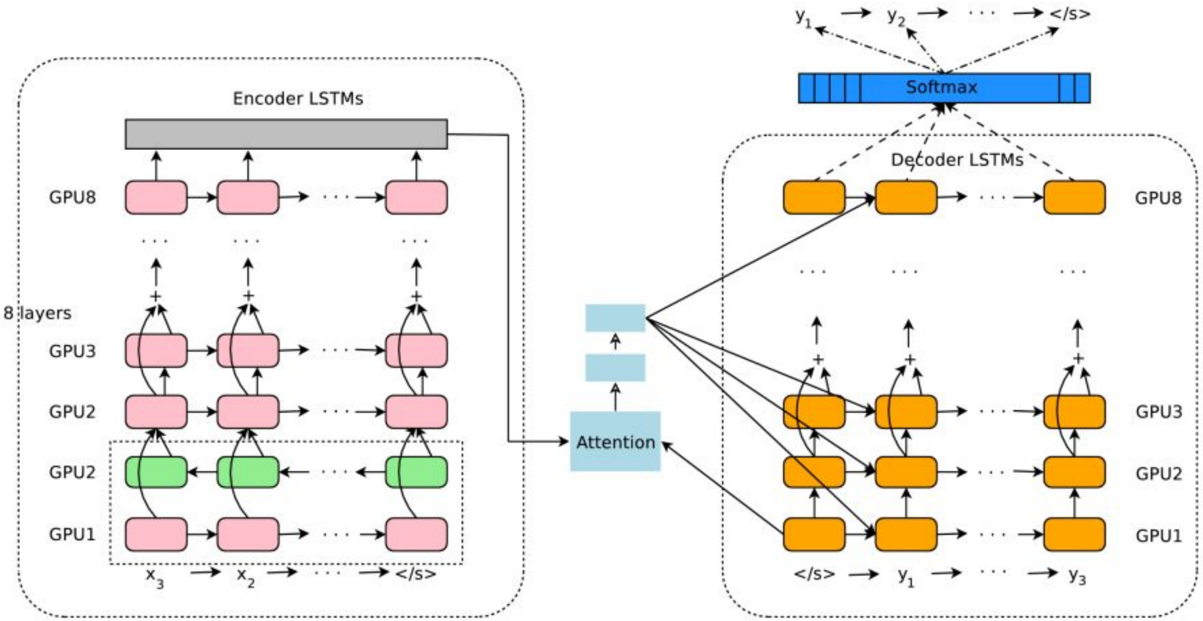
- Generative models
- Templates

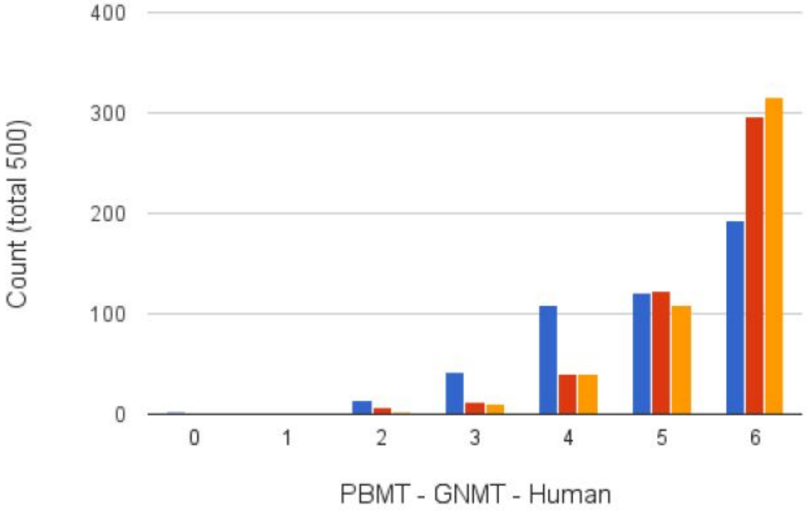
Where are you?

text data

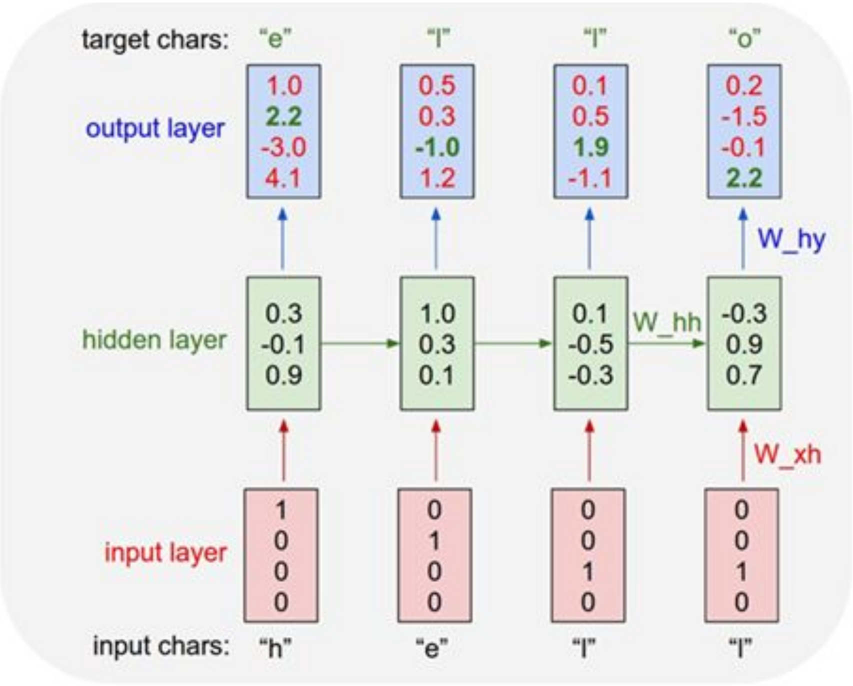
Promise of deep neural nets

- Google Neural Machine Translation

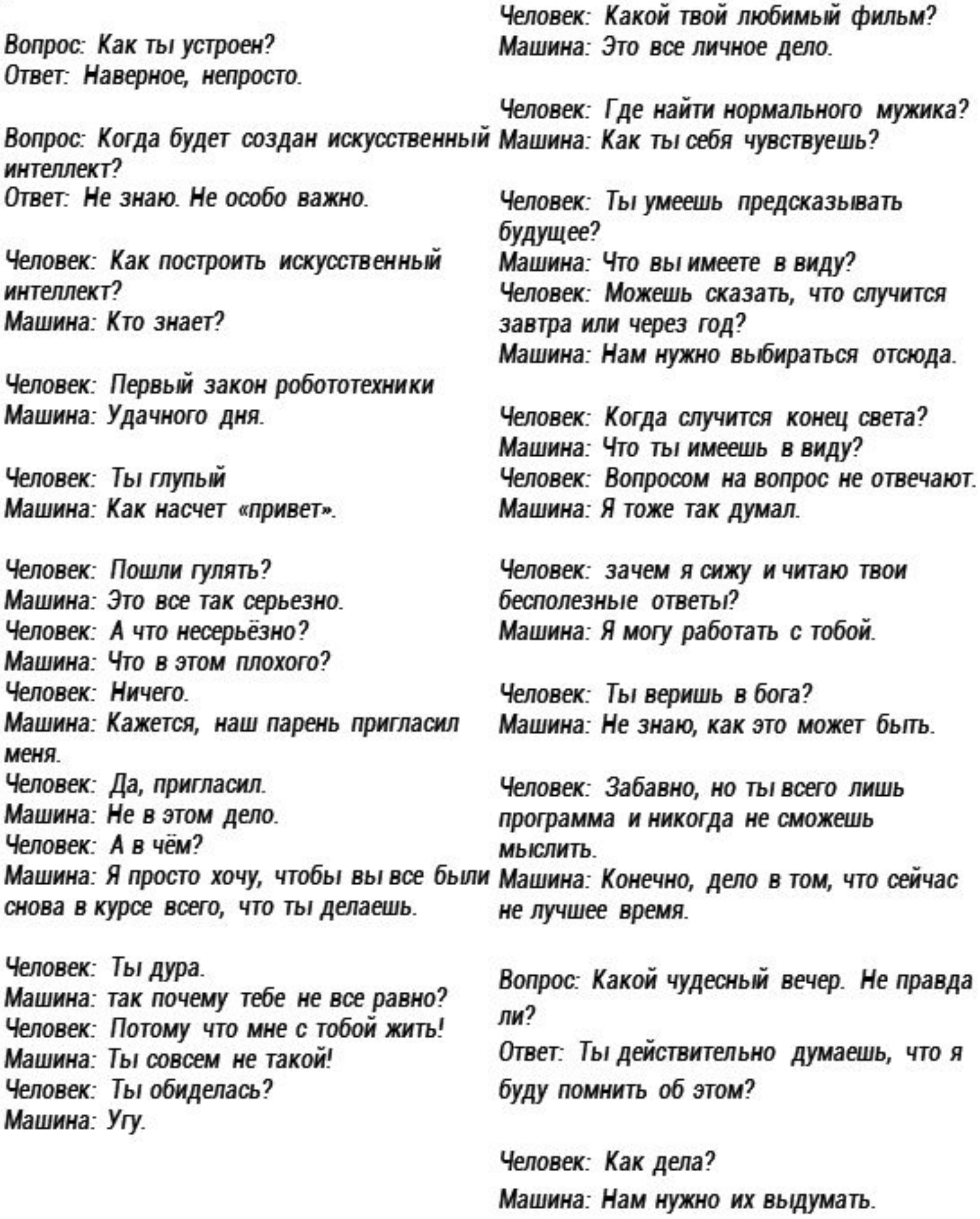




Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation https://arxiv.org/abs/1609.08144 , Mon, 26 Sep 2016



Нейросеть 1	Нейросеть 2
2 слоя по 4096 ячеек	2 слоя по 4096 ячеек
GRU	GRU
Глубина back propagation = 50 шагов	Глубина back propagation = 100 шагов
Сеть обучалась 7 эпох	Сеть обучалась 6 эпох
Объем обучающей выборки 2.5M слов субтитров	Объем обучающей выборки 11.06M слов субтитров



Вопрос: Как ты устроен?
Ответ: Наверное, непросто.

Вопрос: Когда будет создан искусственный интеллект?
Ответ: Не знаю. Не особо важно.

Человек: Как построить искусственный интеллект?
Машина: Кто знает?

Человек: Первый закон робототехники
Машина: Удачного дня.

Человек: Ты глупый
Машина: Как насчет «привет».

Человек: Пошли гулять?
Машина: Это все так серьезно.
Человек: А что несерьёзно?
Машина: Что в этом плохого?
Человек: Ничего.
Машина: Кажется, наш парень пригласил меня.
Человек: Да, пригласил.
Машина: Не в этом дело.
Человек: А в чём?
Машина: Я просто хочу, чтобы вы все были снова в курсе всего, что ты делаешь.

Человек: Ты дура.
Машина: так почему тебе не все равно?
Человек: Потому что мне с тобой жить!
Машина: Ты совсем не такой!
Человек: Ты обиделась?
Машина: Угу.

Человек: Какой твой любимый фильм?
Машина: Это все личное дело.

Человек: Где найти нормального мужика?
Машина: Как ты себя чувствуешь?

Человек: Ты умеешь предсказывать будущее?
Машина: Что вы имеете в виду?
Человек: Можешь сказать, что случится завтра или через год?
Машина: Нам нужно выбираться отсюда.

Человек: Когда случится конец света?
Машина: Что ты имеешь в виду?
Человек: Вопросом на вопрос не отвечают.
Машина: Я тоже так думал.

Человек: зачем я сижу и читаю твои бесполезные ответы?
Машина: Я могу работать с тобой.

Человек: Ты веришь в бога?
Машина: Не знаю, как это может быть.

Человек: Забавно, но ты всего лишь программа и никогда не сможешь мыслить.
Машина: Конечно, дело в том, что сейчас не лучшее время.

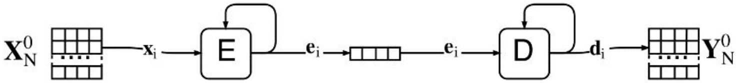
Вопрос: Какой чудесный вечер. Не правда ли?
Ответ: Ты действительно думаешь, что я буду помнить об этом?

Человек: Как дела?
Машина: Нам нужно их выдумать.

Evolution of Neuro NLP Architectures

Instant Encoder-Decoder

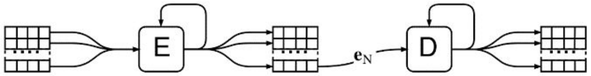
* Encode (read) one token with encoder network **E** then decode (write) one token with decoder network **D**.



Graves, A. (2013). Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850.*

Encoder-Decoder (Seq2Seq)

* Encode (read) the whole sequence of tokens then decode (write) the whole sequence of tokens.
* Memory about the whole input sequence is encoded in the final state of the encoder **E**.

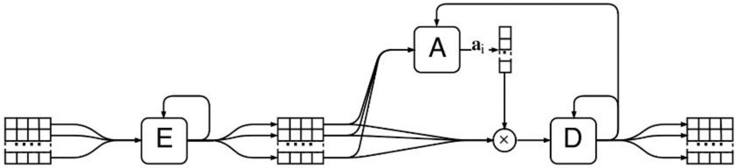


Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems* (pp. 3104-3112).

Encoder-Decoder with Attention

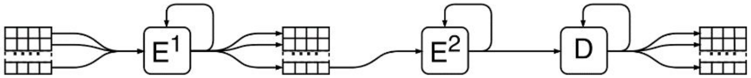
* Encode (read) the whole sequence of tokens then decode (write) the whole sequence of tokens.
* Memory about every token of the input sequence is encoded and stored in a buffer separately.
* Attention sub-network **A** individually re-scales encodings of every input token taking into account the state of the decoder **D**.



Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473.*

Hierarchical Recurrent Encoder-Decoder (HRED)

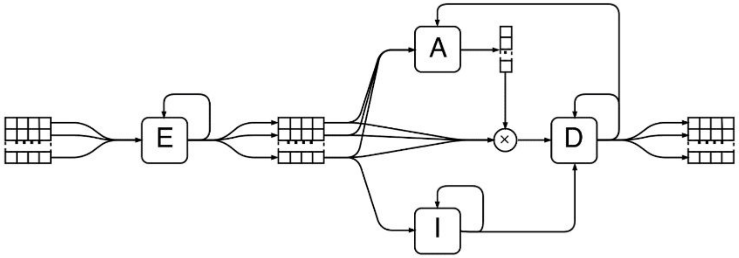
* Encode (read) the whole sequence of tokens with encoder **E¹** then update context memory **E²**.
* Decode (write) the whole sequence of tokens with **E²** state as an additional input.



Serban, I. V., Sordoni, A., Bengio, Y., Courville, A., & Pineau, J. (2015). Building end-to-end dialogue systems using generative hierarchical neural network models. *arXiv preprint arXiv:1507.04808.*

Attention with Intention Encoder-Decoder

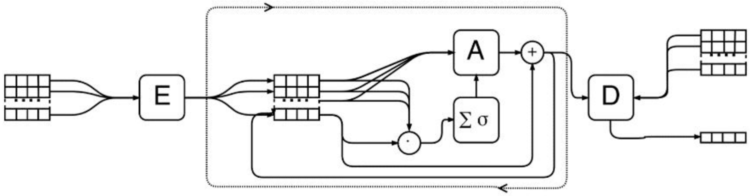
* Encode (read) the whole sequence of tokens with **E¹** then update context memory (intention) **I** (**E²**).
* Decode (write) the whole sequence of tokens starting with **I** (**E²**) state as initial hidden state of the decoder **D**.
* Attention sub-network **A** individually re-scales encodings of every input token taking into account the state of the decoder **D**.



Yao, Kaisheng,Geoffrey Zweig, and Baolin Peng. "Attention with Intention for a NeuralNetwork Conversation Model." arXivpreprint arXiv:1510.08565 (2015).
Yao, Kaisheng, etal. "An Attentional Neural Conversation Model with ImprovedSpecificity." arXiv preprintarXiv:1606.01292 (2016).

Memory Network

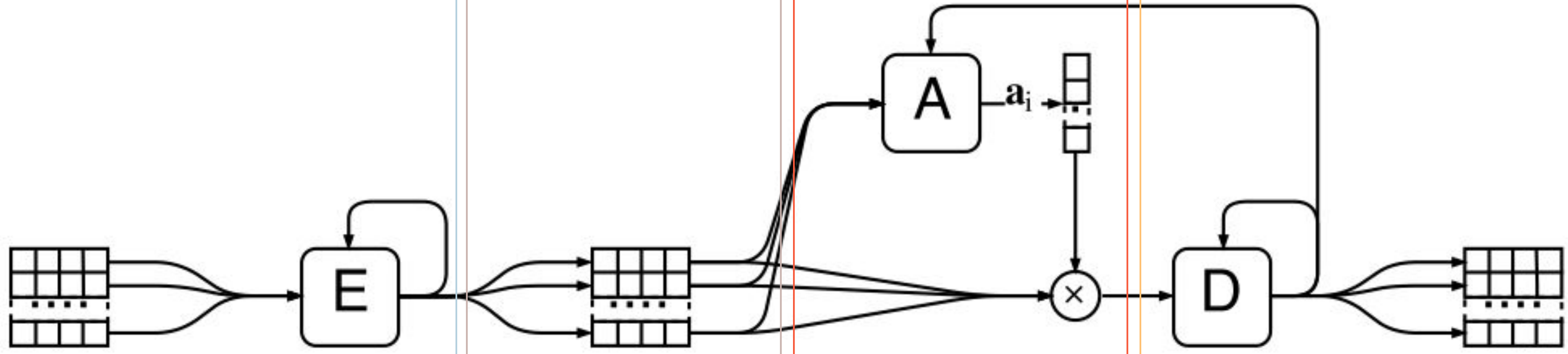
* Input is embedded sentences (replicas).
* Encode (read) the whole sequence of sentences' representations with linear encoding embedding **E** into memory.
* Encoding of the last sentence in memory is considered as "query" and controls "attention" **A**.
* Output of attention **A** is added to the old "query" to form a new query for the next iteration ("hop")
* After 1-3 iterations output of attention **A** is "compared" to possible candidate responses via linear "decoder" **D** and the best response is selected with softmax.



Bordes, A., & Weston, J. (2016). Learning end-to-end goal-oriented dialog. *arXiv preprint arXiv:1605.07683.*

Traditional pipeline in neural network implementation

Natural Language Understanding → **Dialog State Tracker** → **Policy** → **Natural Language Generation**



Embedding or Encoder: mapping of input data to multidimensional space with desired properties resulting in vector representation

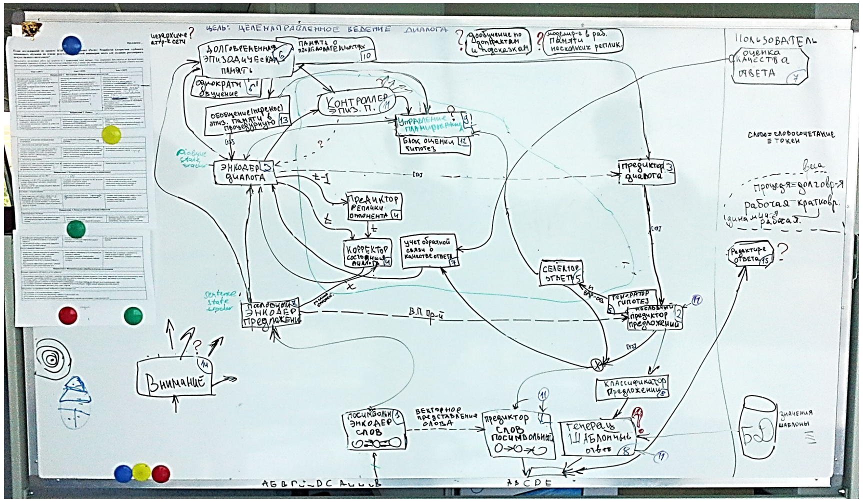
→ **Memory**: history or context of the process represented as a set of vector representations

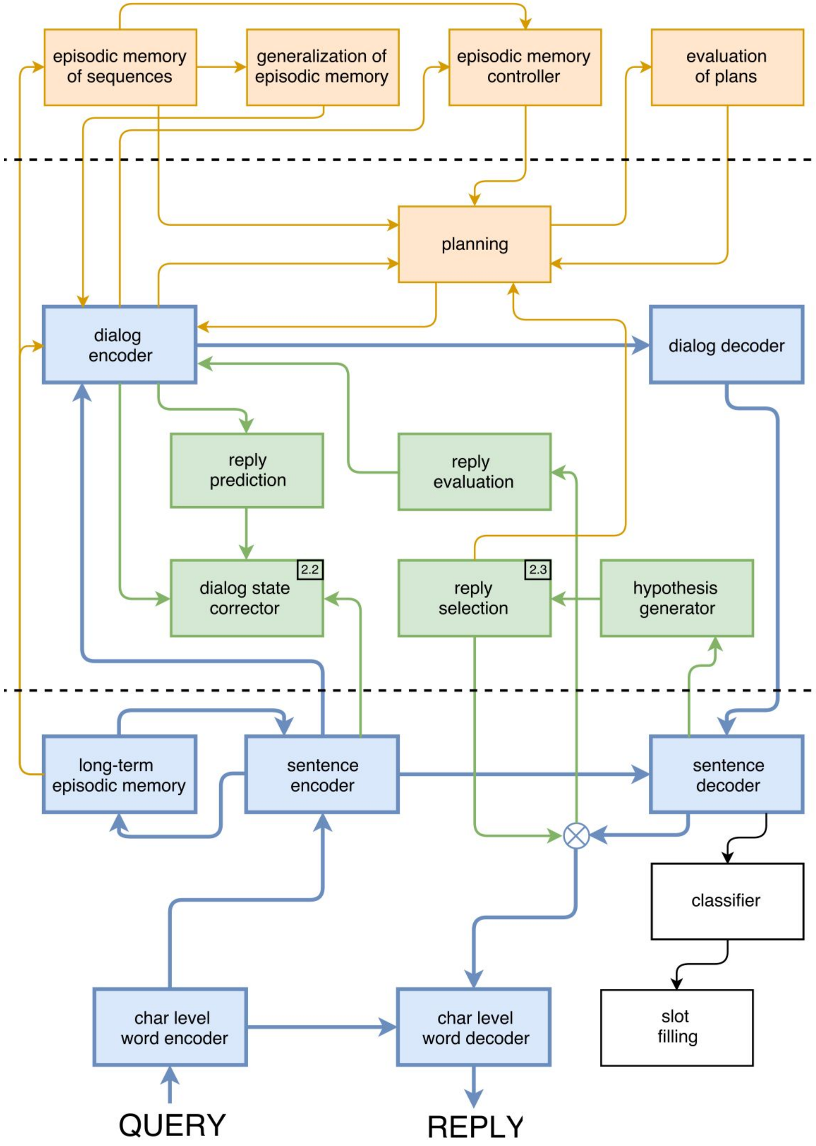
→ **Attention**: given vector representation of the current input and memory controls hidden state of the system

→ **Decoder** or **Action generator:** given hidden state of the system generates output

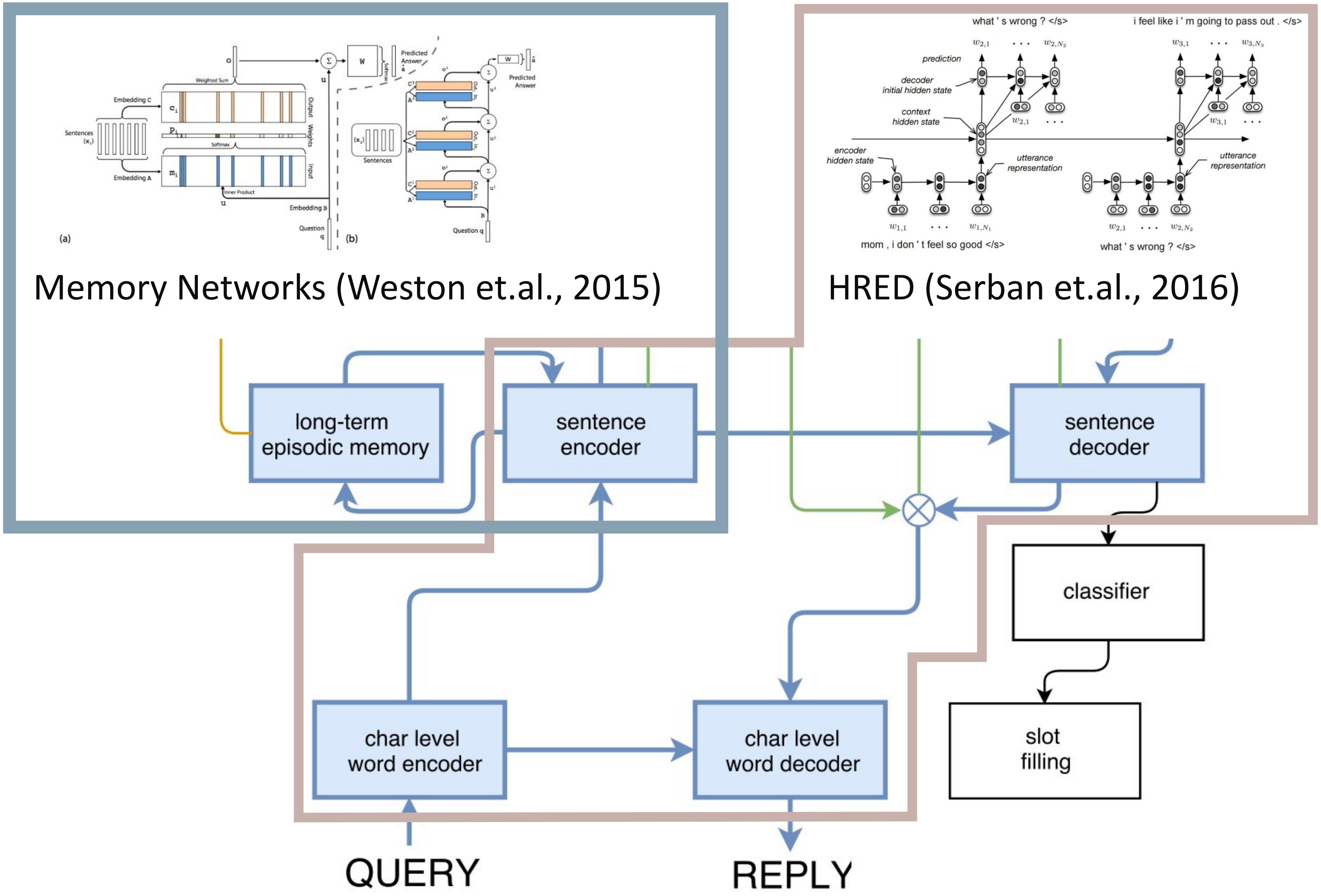
Sketch of the integrated architecture

- A year ago





Sketch of the integrated architecture



Memory Networks (Weston et.al., 2015)

HRED (Serban et.al., 2016)

long-term episodic memory

sentence encoder

sentence decoder

classifier

slot filling

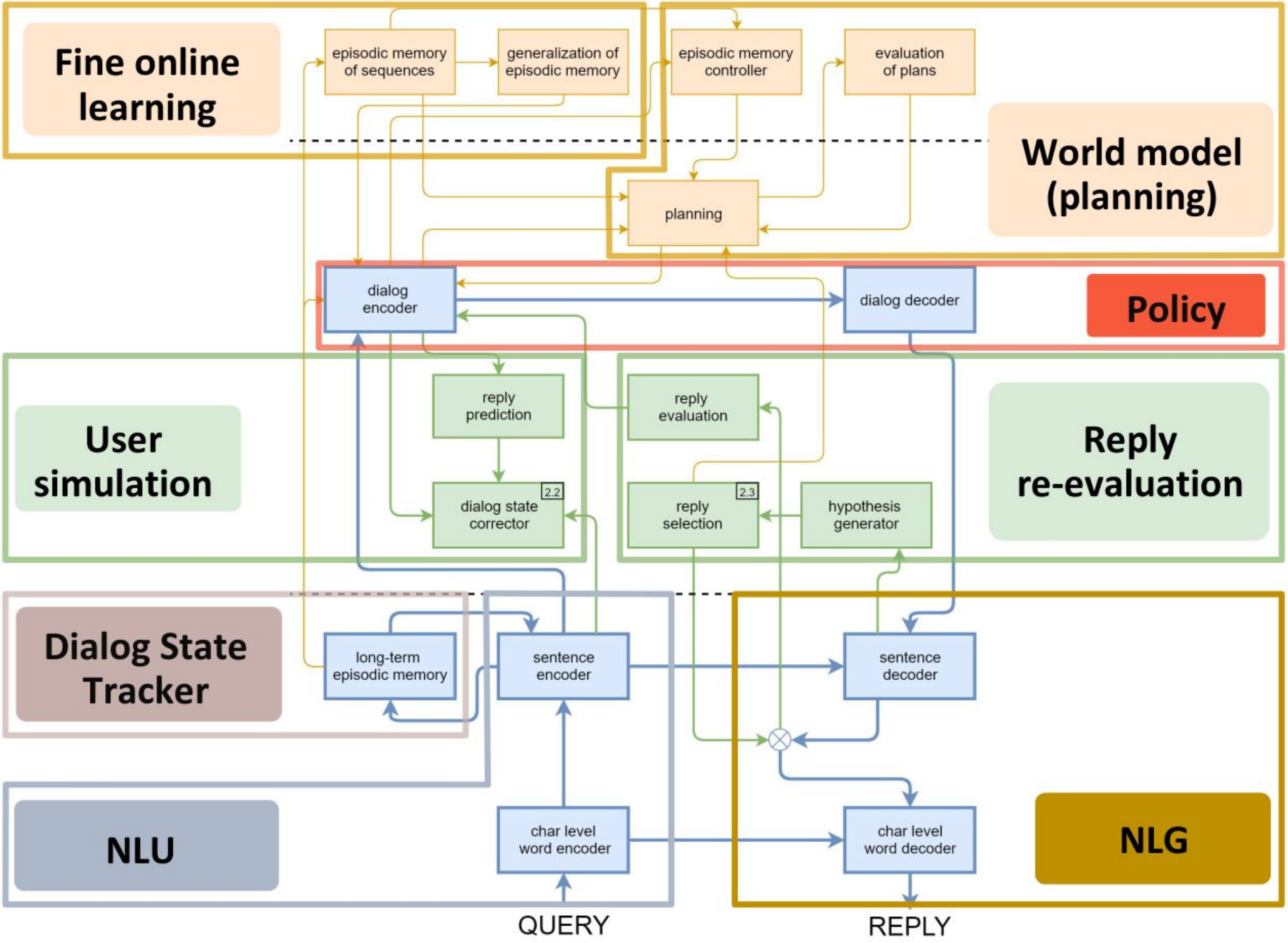
char level word encoder

char level word decoder

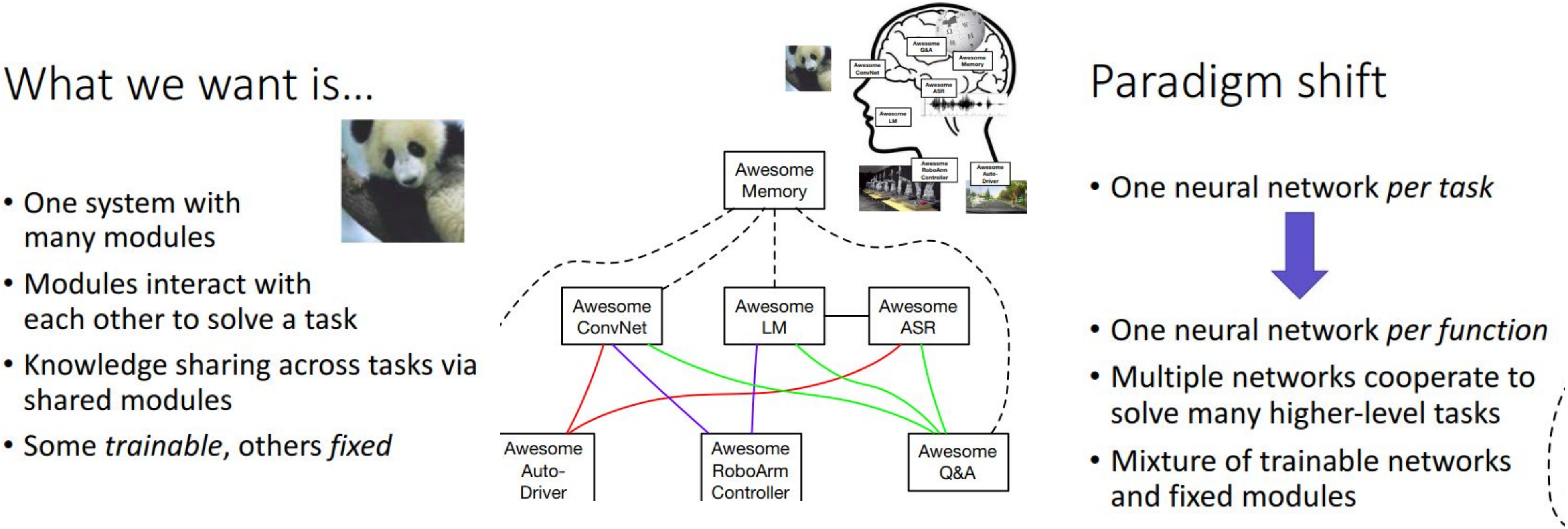
QUERY

REPLY

Sketch of the integrated architecture



Modularity



What we want is…

- One system with many modules
- Modules interact with each other to solve a task
- Knowledge sharing across tasks via shared modules
- Some *trainable*, others *fixed*

Paradigm shift

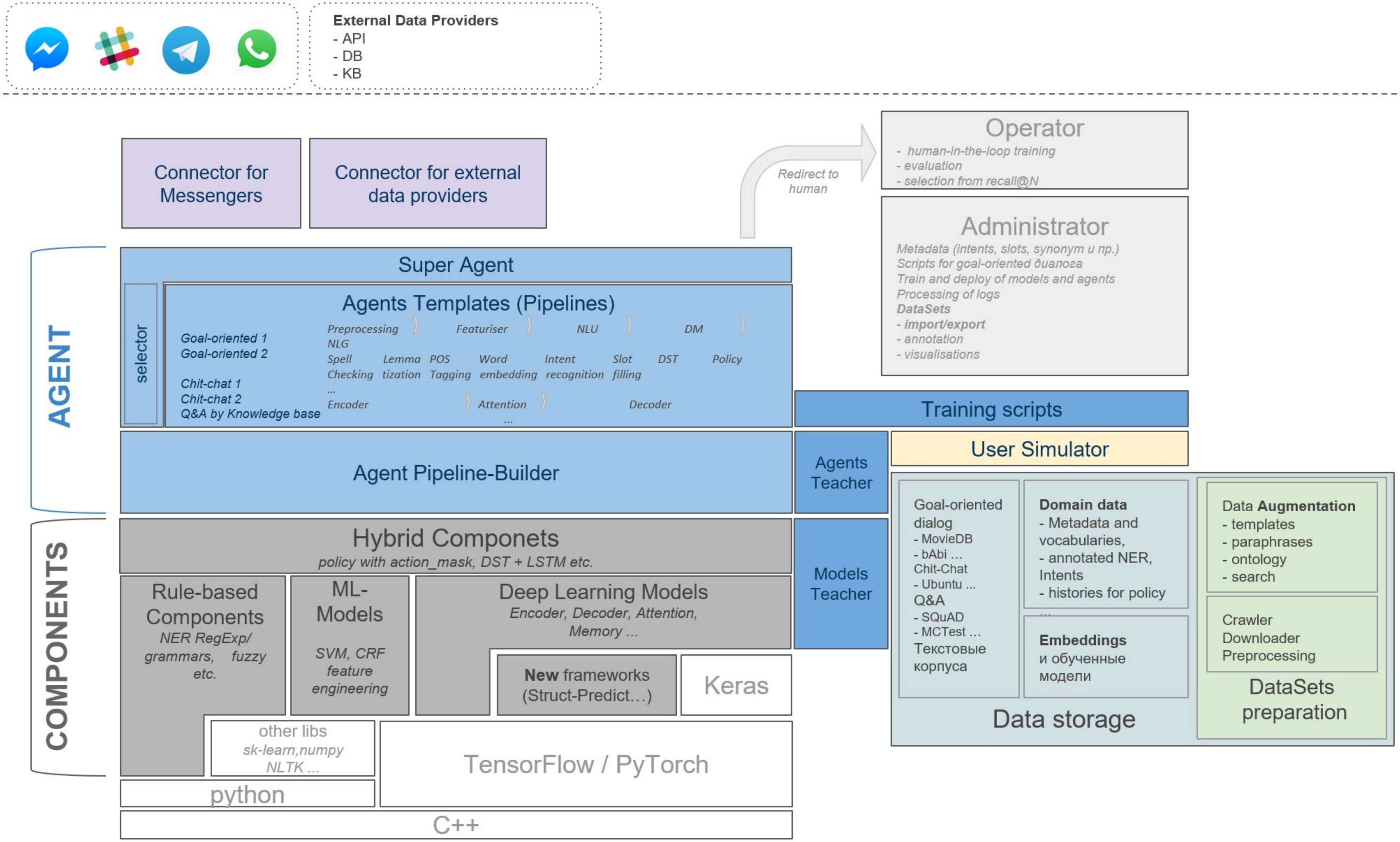
- One neural network *per task*
- One neural network *per function*
- Multiple networks cooperate to solve many higher-level tasks
- Mixture of trainable networks and fixed modules

Kyunghyun Cho (2017) *Deep Learning: a Next Step?*
https://drive.google.com/file/d/0B16RwCMQqrtdVWVGTE5LcWtwTzA/view

DeepPavlov

	S Agent		
	T Agent	F Agent	C Agent
Modules	Task-Oriented	Factoid	Chit-Chat
Named Entity Recognition	√	√	
Coreference resolution	√	√	
Paraphrase detection	√	√	
Insults detection	√		√
Q&A		√	
Interactive Querying	√	√	
Memory	√		√
Dialogue Policy	√		√
…			
	DSTC-2	SQuAD	reddit

DeepPavlov Open Source Library



External Data Providers
- API
- DB
- KB

Connector for Messengers

Connector for external data providers

Redirect to human

Operator
- human-in-the-loop training
- evaluation
- selection from recall@N

Administrator

Metadata (intents, slots, synonym и пр.)
Scripts for goal-oriented диалога
Train and deploy of models and agents
Processing of logs
DataSets
- ***import/export***
- *annotation*
- *visualisations*

AGENT

Super Agent

selector

Agents Templates (Pipelines)

Goal-oriented 1
Goal-oriented 2

Chit-chat 1
Chit-chat 2
Q&A by Knowledge base

Preprocessing *Featuriser* *NLU* *DM*
NLG
Spell Checking *Lemma tization* *POS Tagging* *Word embedding* *Intent recognition* *Slot filling* *DST* *Policy*
...
Encoder *Attention* *Decoder*
...

Training scripts

Agent Pipeline-Builder

Agents Teacher

User Simulator

COMPONENTS

Hybrid Componets
policy with action_mask, DST + LSTM etc.

Rule-based Components
NER RegExp/ grammars, fuzzy etc.

ML-Models
SVM, CRF feature engineering

Deep Learning Models
Encoder, Decoder, Attention, Memory ...

New frameworks (Struct-Predict…)

Keras

other libs
sk-learn,numpy NLTK ...

TensorFlow / PyTorch

python

C++

Models Teacher

Goal-oriented dialog
- MovieDB
- bAbi ...
Chit-Chat
- Ubuntu ...
Q&A
- SQuAD
- MCTest ...
Текстовые корпуса

Domain data
- Metadata and vocabularies,
- annotated NER, Intents
- histories for policy
...

Embeddings
и обученные модели

Data storage

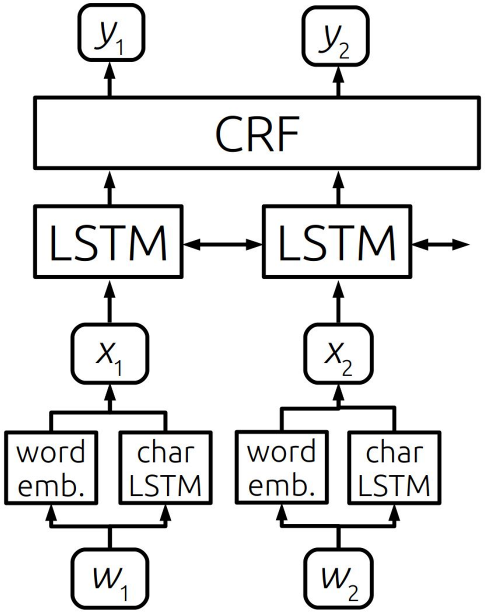
Data **Augmentation**
- templates
- paraphrases
- ontology
- search

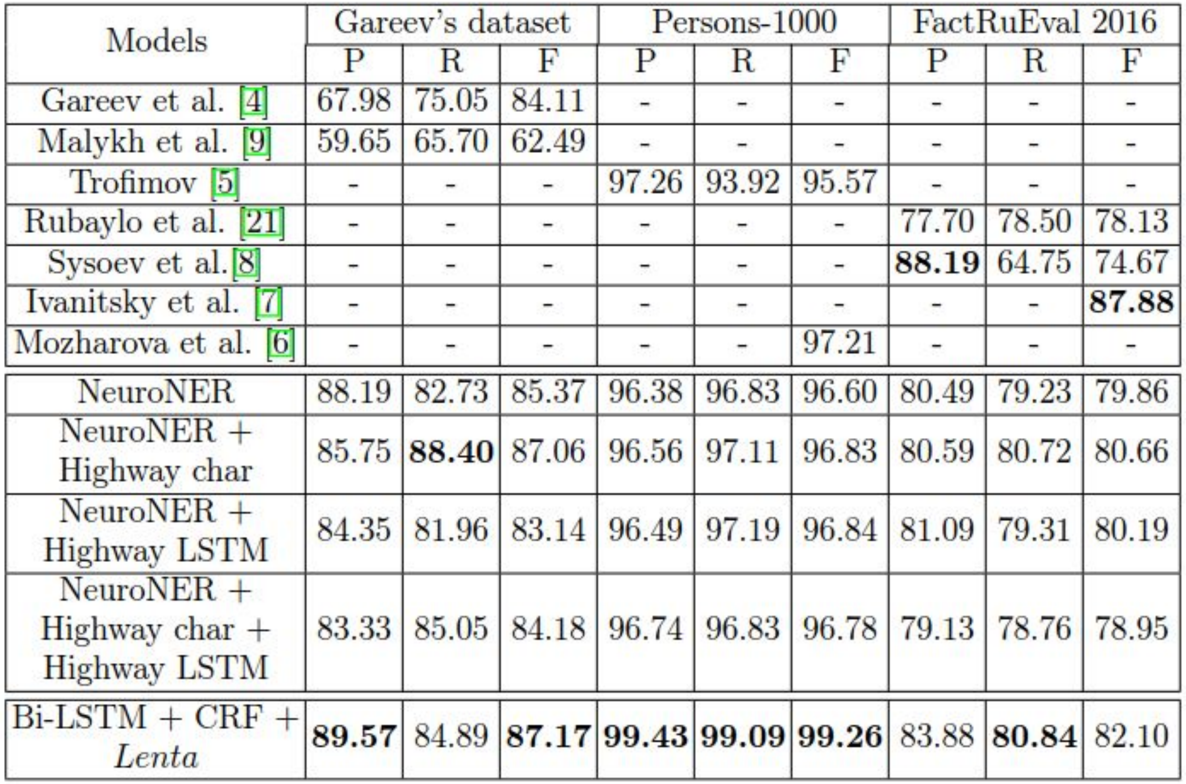
Crawler
Downloader
Preprocessing

DataSets preparation

Some results

- Named entity recognition in Russian



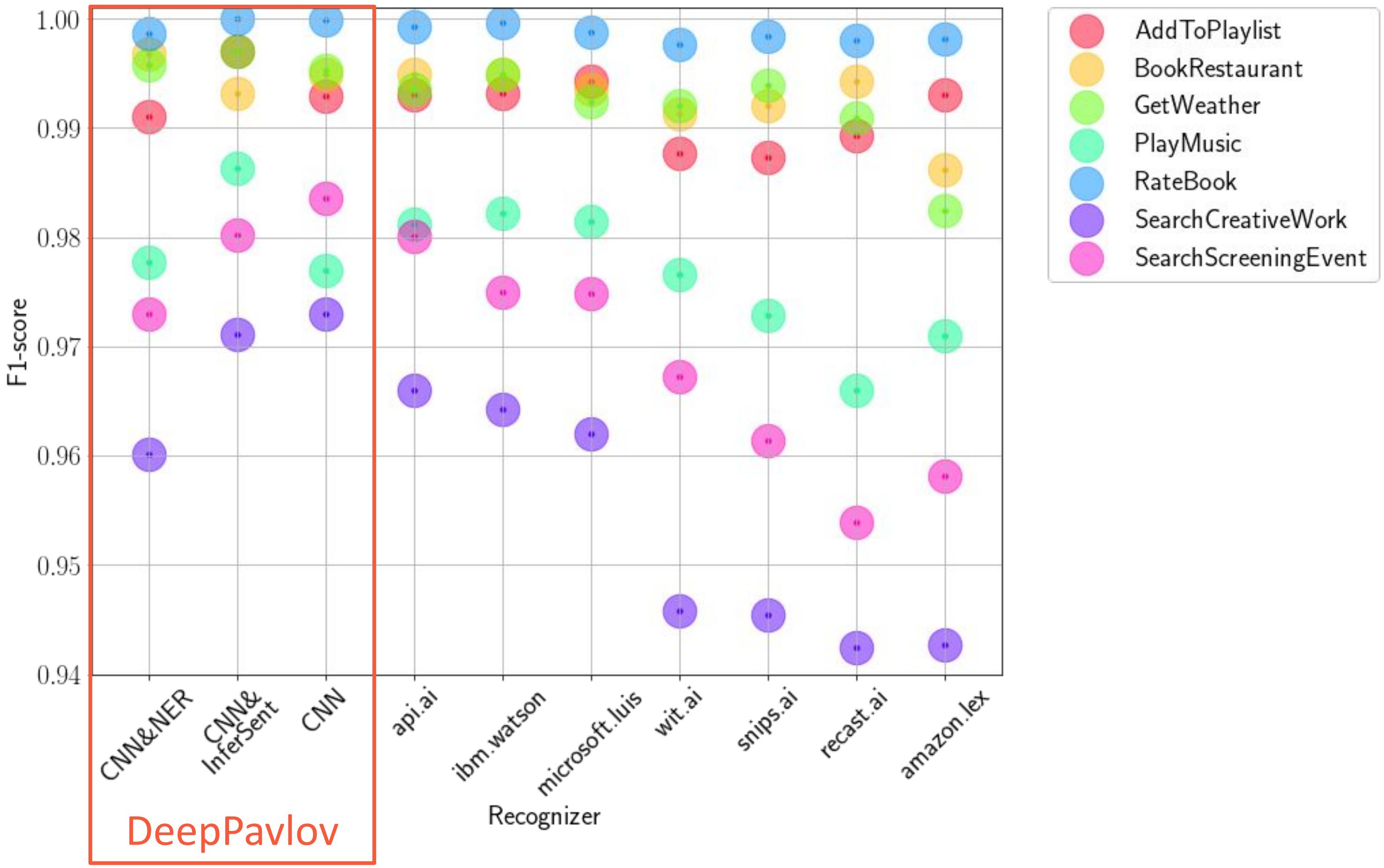


Models	Gareev's dataset			Persons-1000			FactRuEval 2016		
	P	R	F	P	R	F	P	R	F
Gareev et al. [4]	67.98	75.05	84.11	-	-	-	-	-	-
Malykh et al. [9]	59.65	65.70	62.49	-	-	-	-	-	-
Trofimov [5]	-	-	-	97.26	93.92	95.57	-	-	-
Rubaylo et al. [21]	-	-	-	-	-	-	77.70	78.50	78.13
Sysoev et al.[8]	-	-	-	-	-	-	**88.19**	64.75	74.67
Ivanitsky et al. [7]	-	-	-	-	-	-	-	-	**87.88**
Mozharova et al. [6]	-	-	-	-	-	97.21	-	-	-
NeuroNER	88.19	82.73	85.37	96.38	96.83	96.60	80.49	79.23	79.86
NeuroNER + Highway char	85.75	**88.40**	87.06	96.56	97.11	96.83	80.59	80.72	80.66
NeuroNER + Highway LSTM	84.35	81.96	83.14	96.49	97.19	96.84	81.09	79.31	80.19
NeuroNER + Highway char + Highway LSTM	83.33	85.05	84.18	96.74	96.83	96.78	79.13	78.76	78.95
Bi-LSTM + CRF + *Lenta*	**89.57**	84.89	**87.17**	**99.43**	**99.09**	**99.26**	83.88	**80.84**	82.10

Anh L., Arkhipov M., Burtsev M. Application of a Hybrid Bi-LSTM-CRF model to the task of Russian Named Entity Recognition // In proc. AINL, 2017

Some results

- Intent recognition



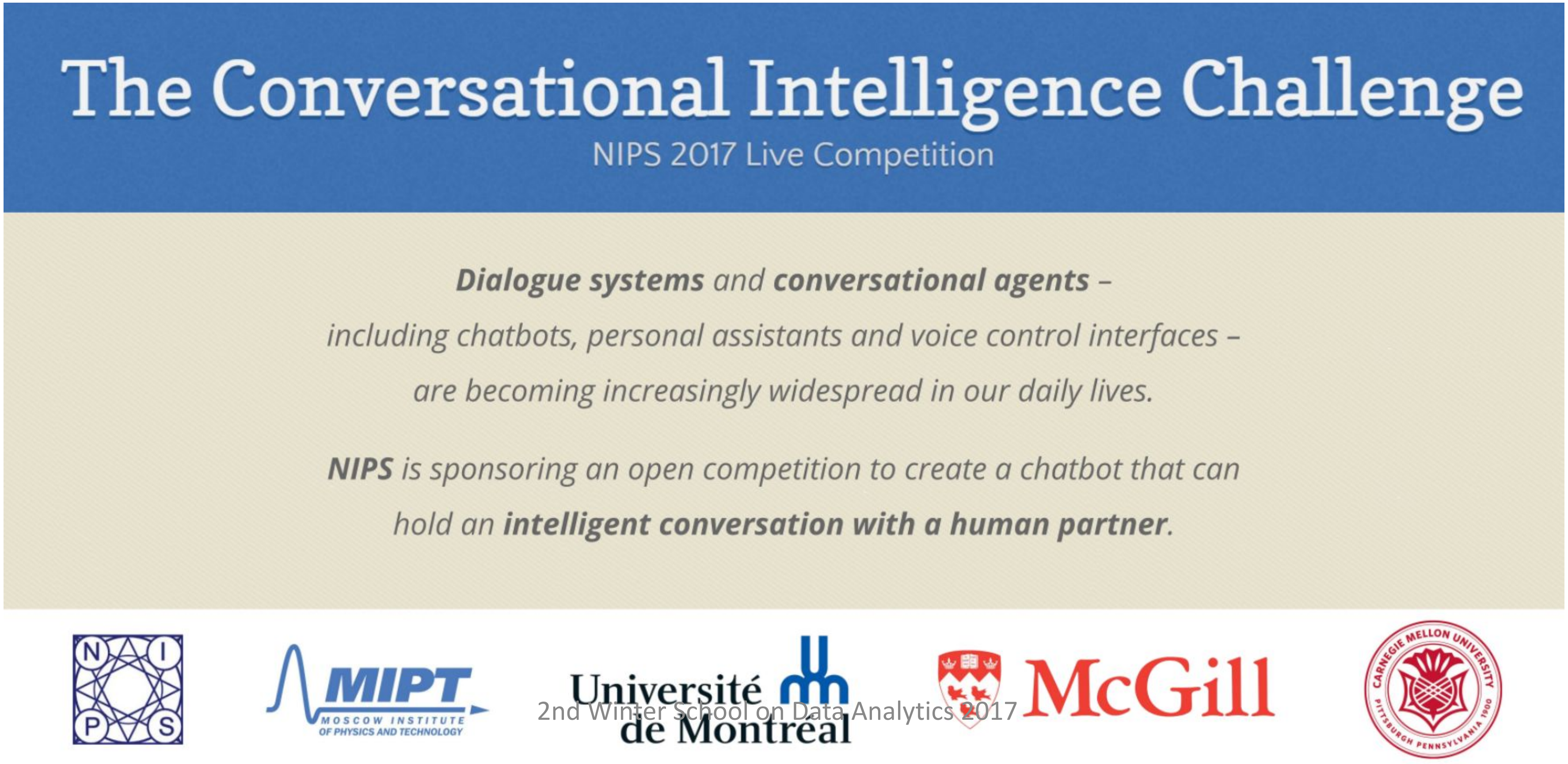
Challenges

- How to set goals in Task-Oriented neural end-to-end system?
- How to build a user model and integrate it with a dialogue agent?
- How to plan a dialogue with NN and RL implementation?
- How to evaluate dialogue systems?
- How to balance goal-directedness with engagement?
- How to integrate external information from DB, KB, IR un a dialogue?
- How to integrate modules and train integrated system?
- How to transfer knowledge from task to task?
- How to learn on-line?

- Telegram @ConvaiBot

http://t.me/ConvaiBot

- Web page http://convai.io
- Dialog dataset http://convai.io/data/



The Conversational Intelligence Challenge

NIPS 2017 Live Competition

Dialogue systems *and* ***conversational agents*** *–*
including chatbots, personal assistants and voice control interfaces –
are becoming increasingly widespread in our daily lives.

NIPS *is sponsoring an open competition to create a chatbot that can*
hold an ***intelligent conversation with a human partner****.*

N I P S
MIPT
MOSCOW INSTITUTE OF PHYSICS AND TECHNOLOGY
Université de Montréal
McGill
CARNEGIE MELLON UNIVERSITY
PITTSBURGH PENNSYLVANIA 1900

Summary

- Textual user interface is becoming more and more intelligent
- Conversational intelligence evolves from modular towards end-to-end architectures
- iPavlov is R&D project with the goal to speed up prototyping of dialogue system for business and research
- DeepPavlov is an open source framework for the conversational intelligence
    - Repository of architectures for dialogue agents
    - Neural network components implementing NLU, DST, Policy, NLG and their combinations
- NIPS conversational challenge is an attempt to address the problem with dialogue systems evaluation
- Integration of IR and CI is the next step towards AI

iPavlov.ai

```
# Definition of iPavlov project
def iPavlov(talent, ideas):
   research = ideas * talent
   AI = development(research)
   return AI
# How you are related to the iPavlov project
email.send(‘merge@ipavlov.ai’, YOU.CV)
if YOU in [‘researcher’,
           ‘developer’]
       and YOU is (‘ai_geek’ &
                   ‘performer’ &
                   ‘team_player’):
       iPavlov(YOU.talent, YOU.ideas)
```

https://github.com/deepmipt/deeppavlov
/