Politecnico di Torino

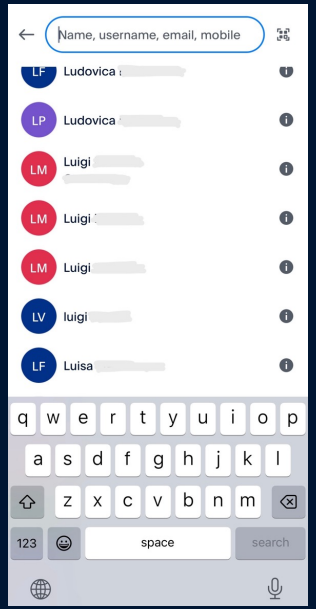
REPLY PAY

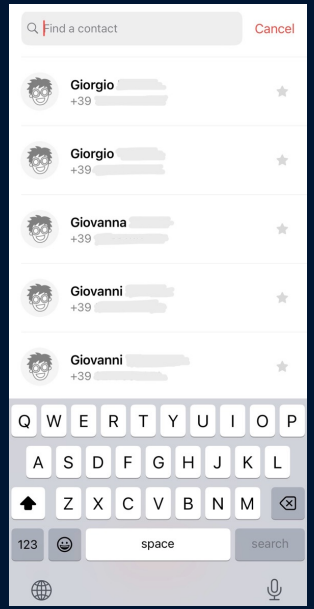
Developing an AI-Powered Voice Assistant for an iOS Payment App

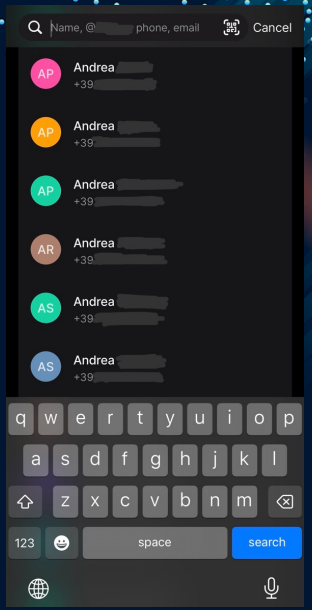
Master's Degree Thesis in Computer Engineering

Candidate: Mario Mastrandrea
Supervisors: Luigi De Russis, Andrea Loffredo

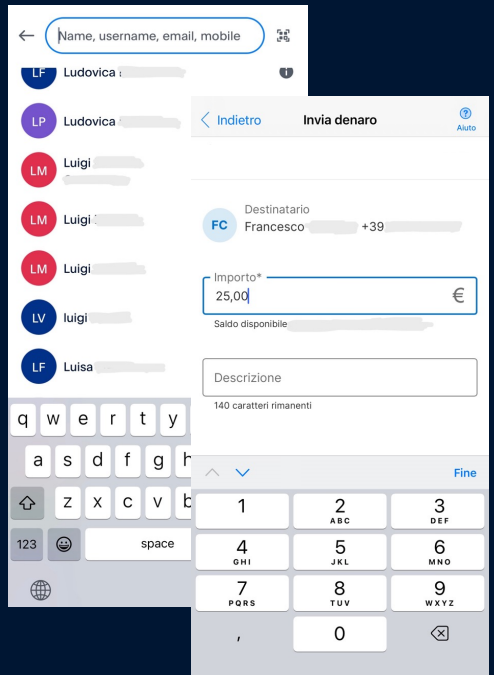
TRADITIONAL TOUCH-BASED INTERACTION

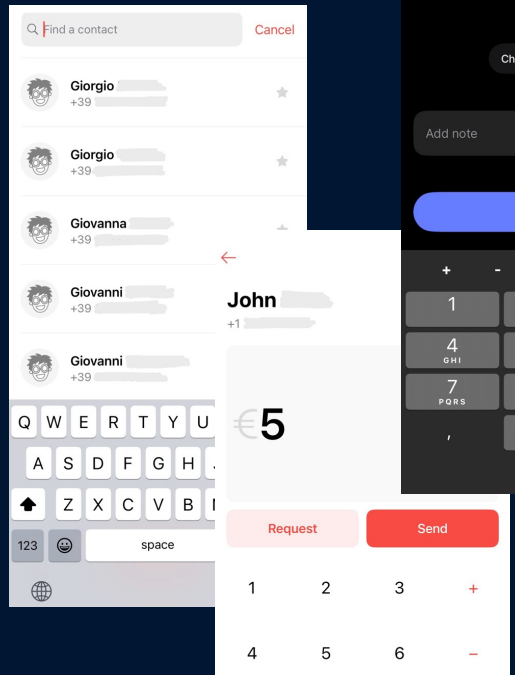


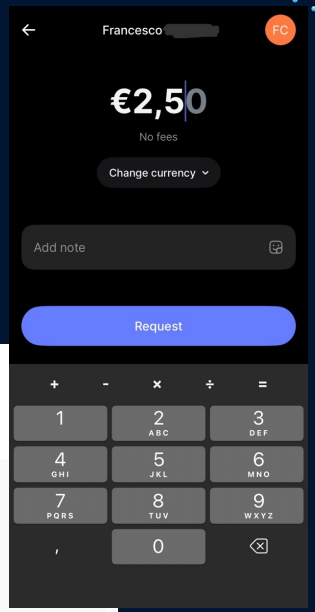


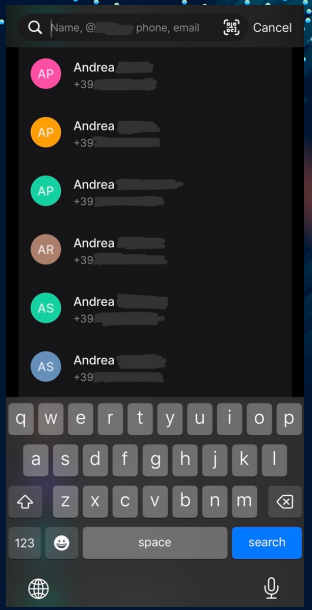


TRADITIONAL TOUCH-BASED INTERACTION

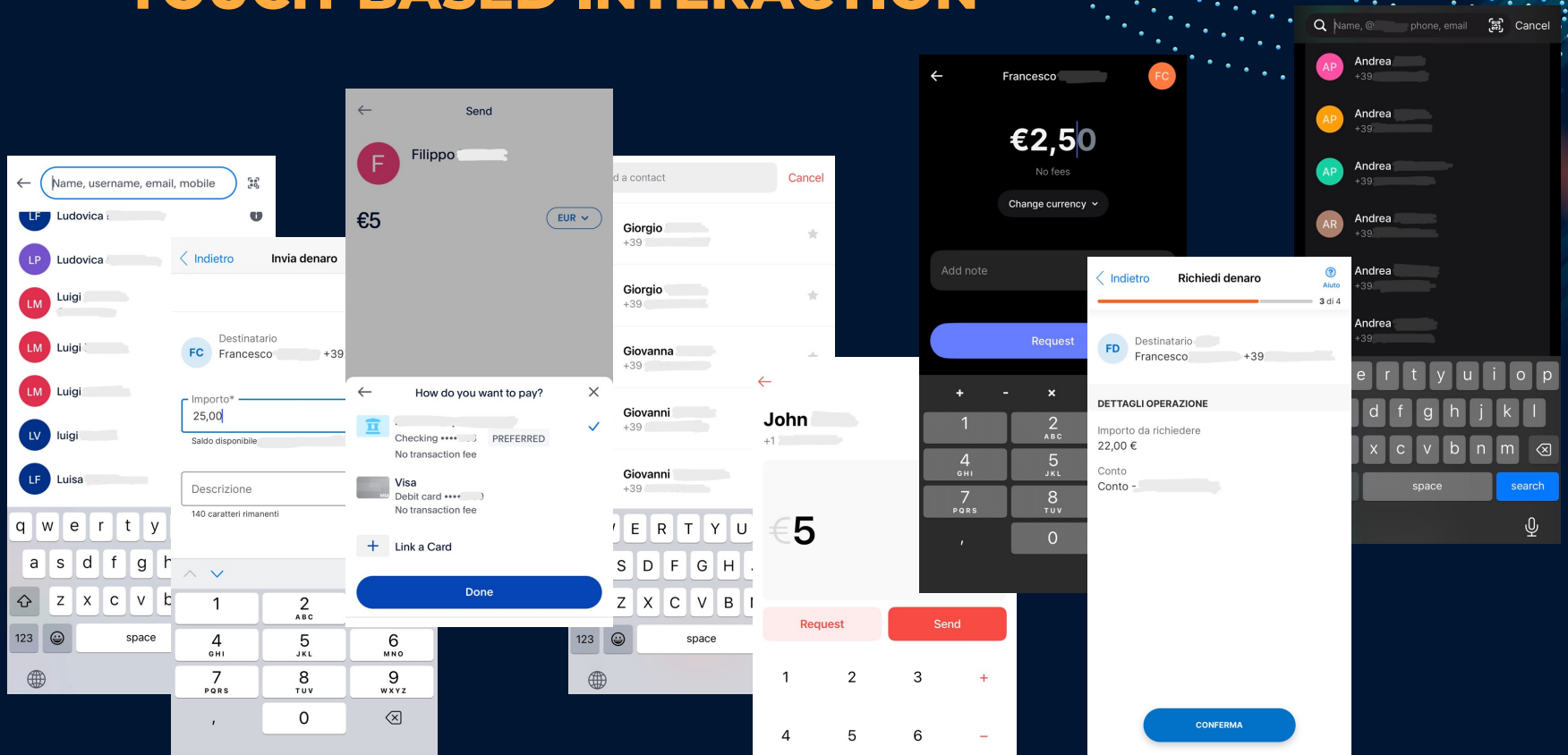








TRADITIONAL TOUCH-BASED INTERACTION



TRADITIONAL TOUCH-BASED INTERACTION

THESIS GOAL

Improve *User eXperience* and *accessibility* of mobile applications offering **financial services**, while maintaining the highest levels of *privacy* and *security*

THESIS GOAL

Improve *User eXperience* and *accessibility* of mobile applications offering **financial services**, while maintaining the highest levels of *privacy* and *security*

HOW?

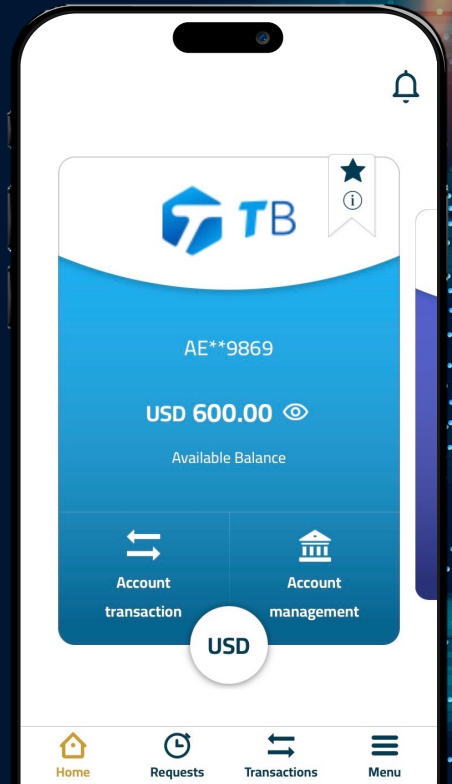
Integrating an **AI-Powered Voice Assistant** into *P2P* payment apps which runs exclusively **on-device**

WORKING CONTEXT

REPLY PAY

This Thesis project was born in collaboration with *Pay Reply*, IT Consultancy company specialized in **Digital Payments**

- Proposed solution for **iOS platform only**, due to my work background in the company
- Real *Peer-to-Peer (P2P)* Payment iOS app taken as **case study**, where to integrate the Voice Assistant



OUTLINE

BACKGROUND

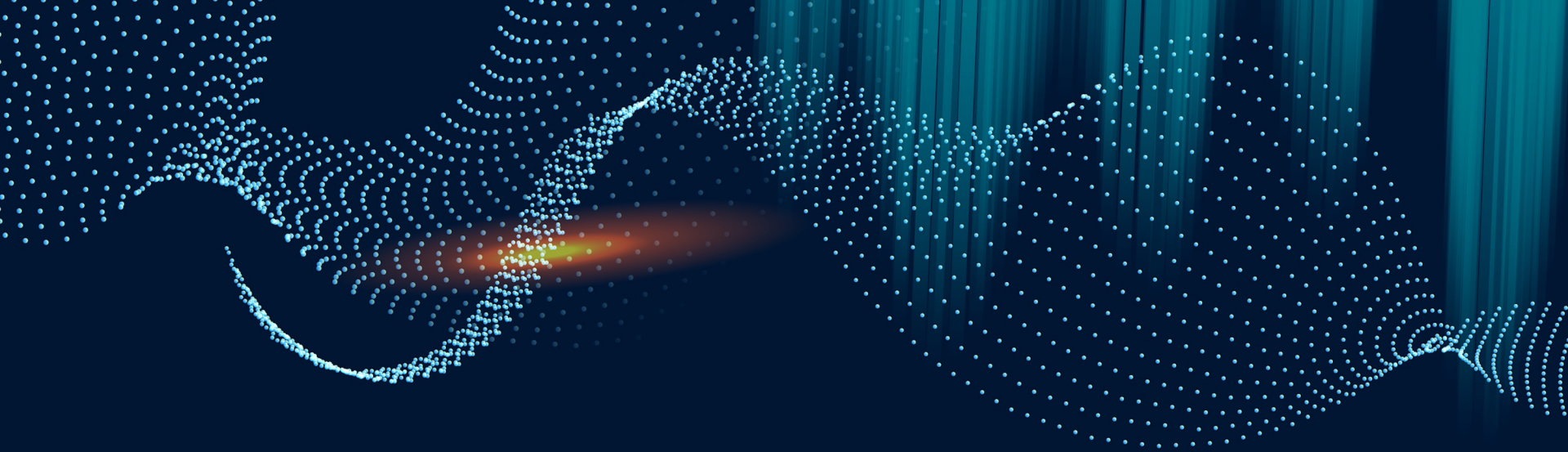
Relevant topics and used state-of-the-art technologies

SOLUTION

System requirements, design and implementation

EVALUATION

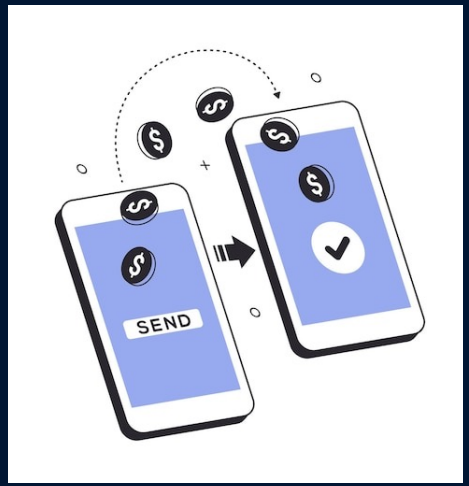
Test app deployment & user-based evaluation phase



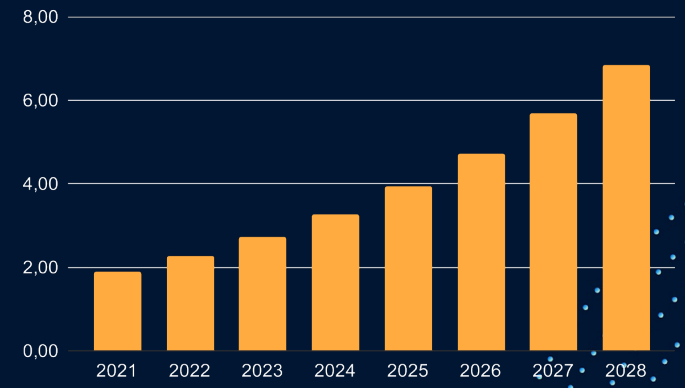
BACKGROUND

Relevant topics and used
state-of-the-art technologies

Peer-to-Peer (P2P) Payments



P2P PAYMENTS MARKET SIZE (USD Trillions)

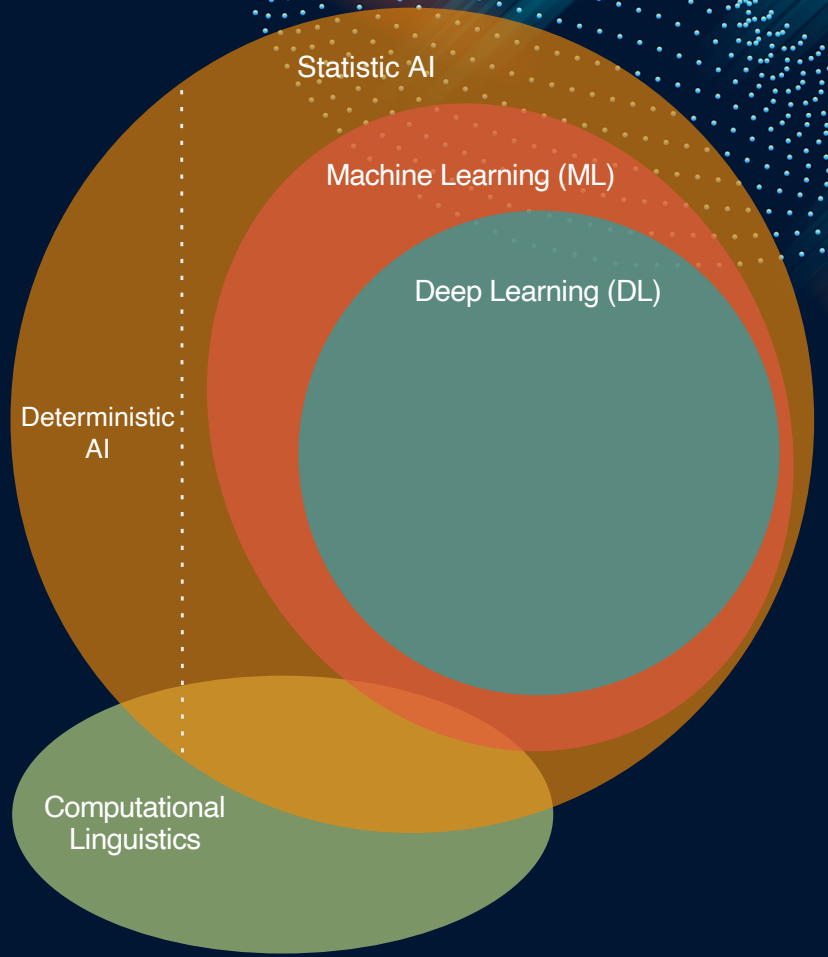


Source: AppInventiv

- **Direct financial transactions** between people, without intermediaries
- Digital platforms empowered by Mobile Applications

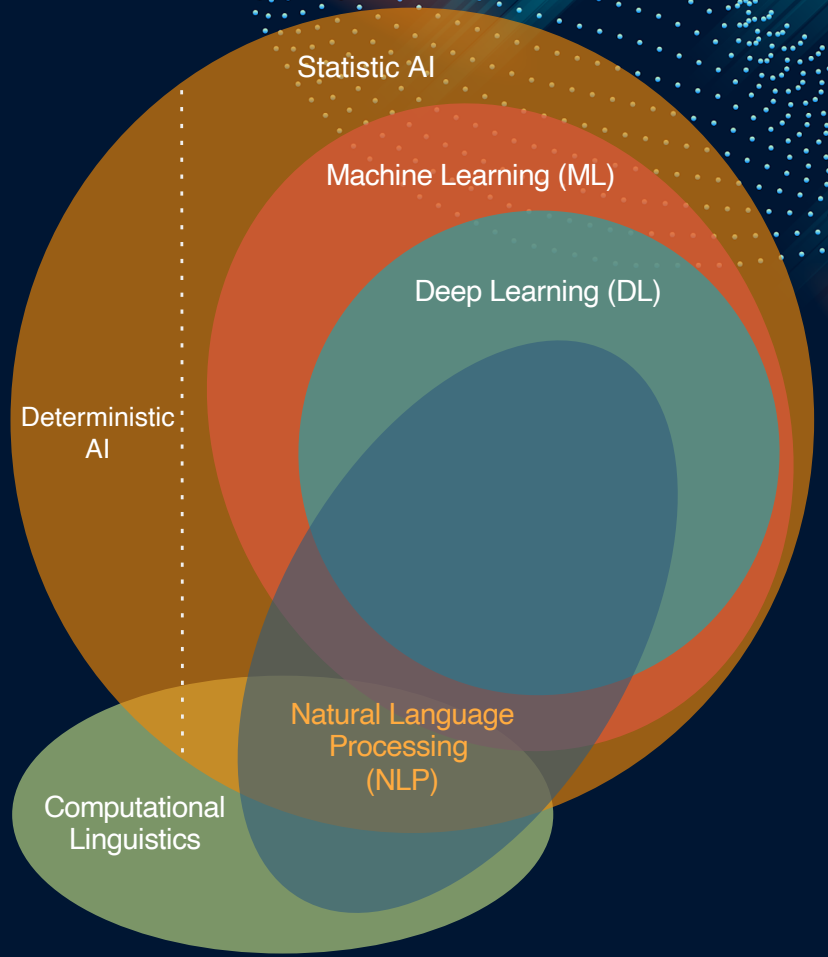
ARTIFICIAL INTELLIGENCE?

- Natural Language Processing (NLP)
- Natural Language Understanding (NLU)



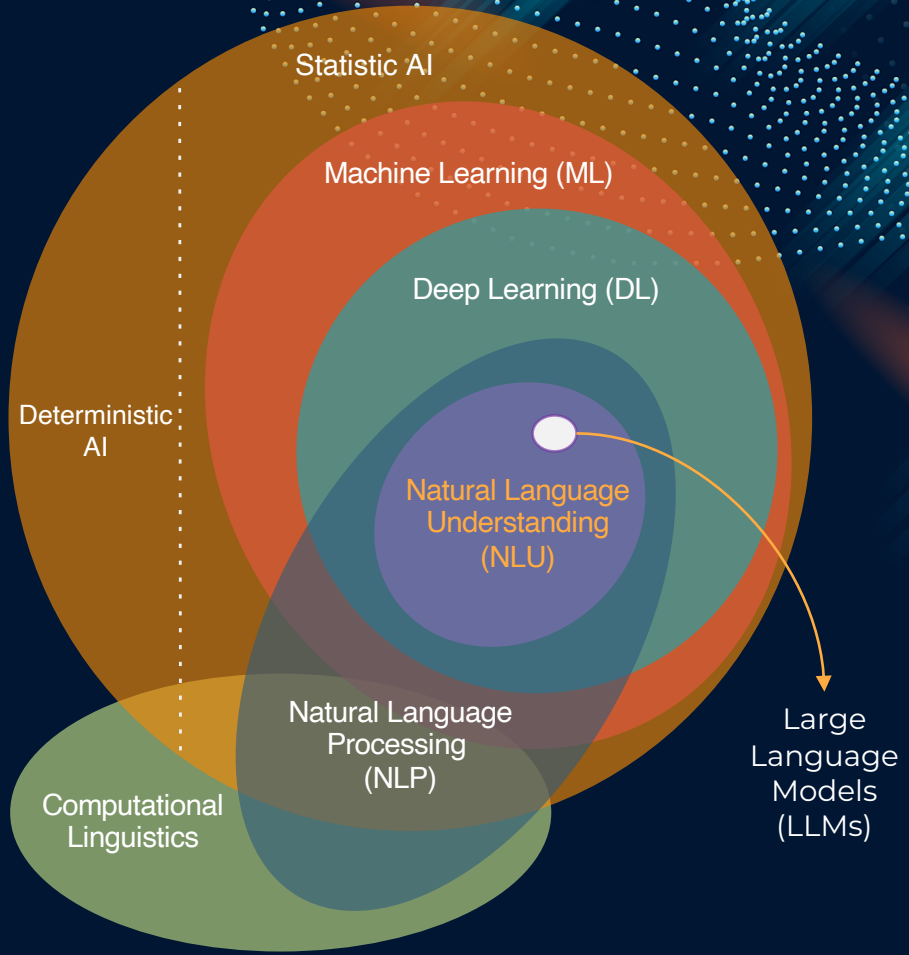
ARTIFICIAL INTELLIGENCE?

- Natural Language Processing (NLP)
- Natural Language Understanding (NLU)



ARTIFICIAL INTELLIGENCE?

- Natural Language Processing (NLP)
- Natural Language Understanding (NLU)



INVOLVED NLP TASKS

SPEECH RECOGNITION

Transform user speech into meaningful sentences (*Speech-to-Text*)

TEXT CLASSIFICATION

Detect user intent **(intent recognition)** and extract relevant information **(entity extraction)**

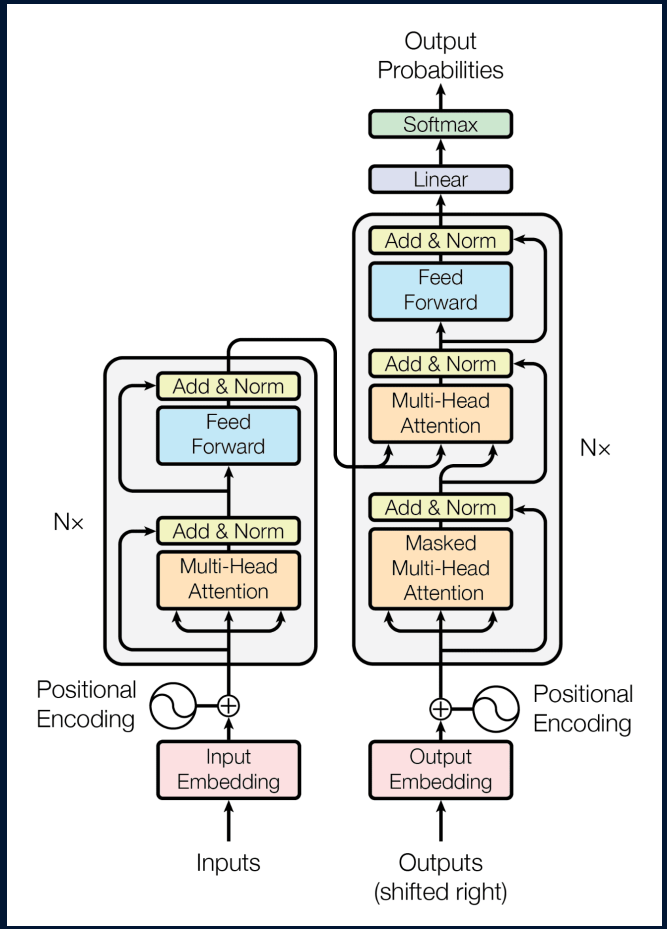
SPEECH SYNTHESIS

Transform textual responses into audible speech *(Text-to-Speech)*

TRANSFORMER ARCHITECTURE

At the foundation of LLMs

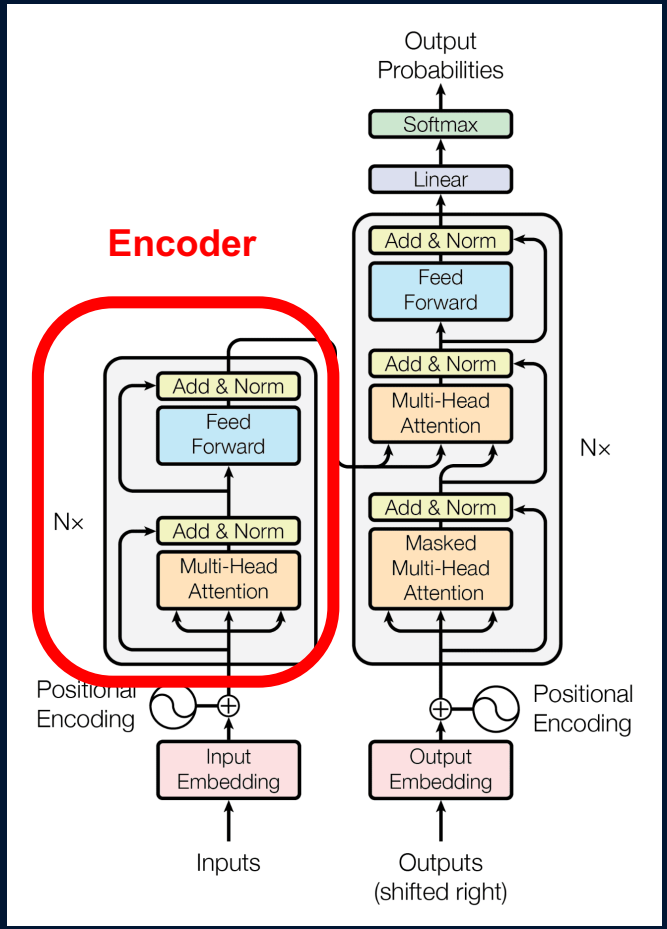
- *"Attention is All You Need"* – Google (2017)
- Born for a *Machine Translation* task
- (Deep) *encoder-decoder* architecture:
  - **Encoder**: produces context-aware numerical representations
    ⇒ the basis of **BERT**
  - **Decoder**: generates the next words (one at a time)
    ⇒ the basis of **GPT**



TRANSFORMER ARCHITECTURE

At the foundation of LLMs

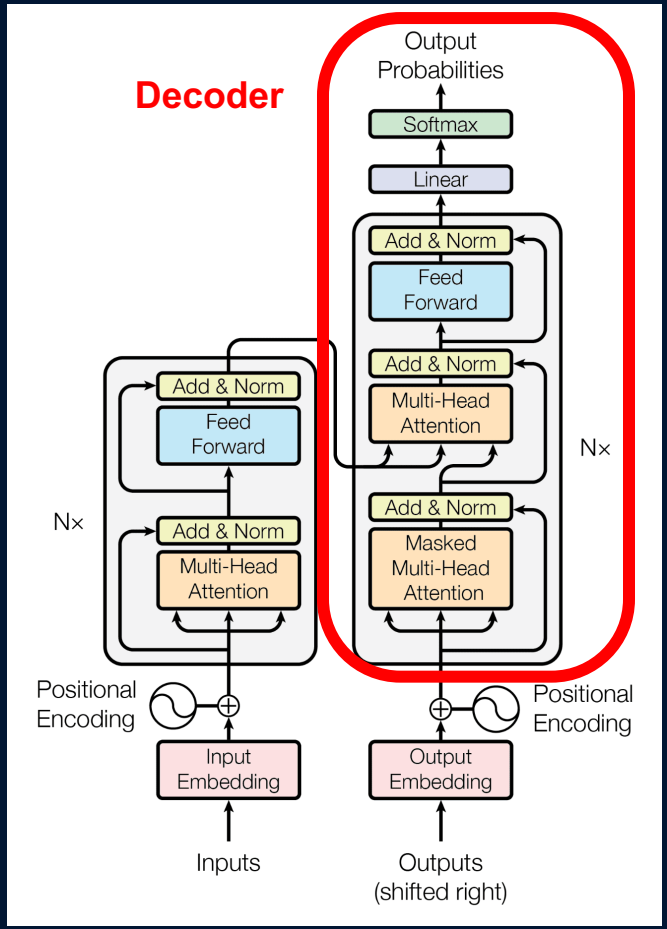
- *"Attention is All You Need"* – Google (2017)
- Born for a *Machine Translation* task
- (Deep) *encoder-decoder* architecture:
  - **Encoder**: produces context-aware numerical representations
    ⇒ the basis of **BERT**
  - **Decoder**: generates the next words (one at a time)
    ⇒ the basis of **GPT**



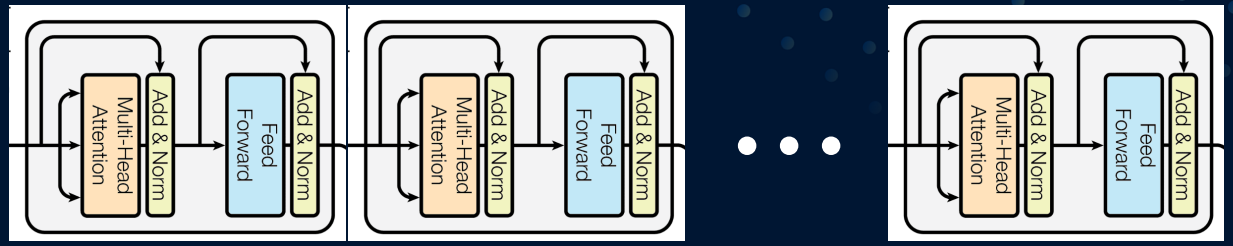
TRANSFORMER ARCHITECTURE

At the foundation of LLMs

- *"Attention is All You Need"* – Google (2017)
- Born for a *Machine Translation* task
- (Deep) *encoder-decoder* architecture:
  - **Encoder**: produces context-aware numerical representations
    ⇒ the basis of **BERT**
  - **Decoder**: generates the next words (one at a time)
    ⇒ the basis of **GPT**

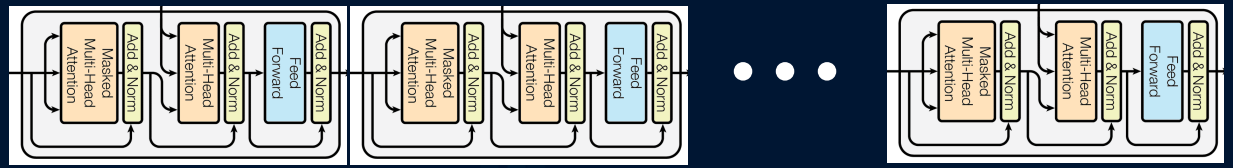


Bidirectional Encoder Representation from Transformers



BERT

Generative Pretrained Transformer



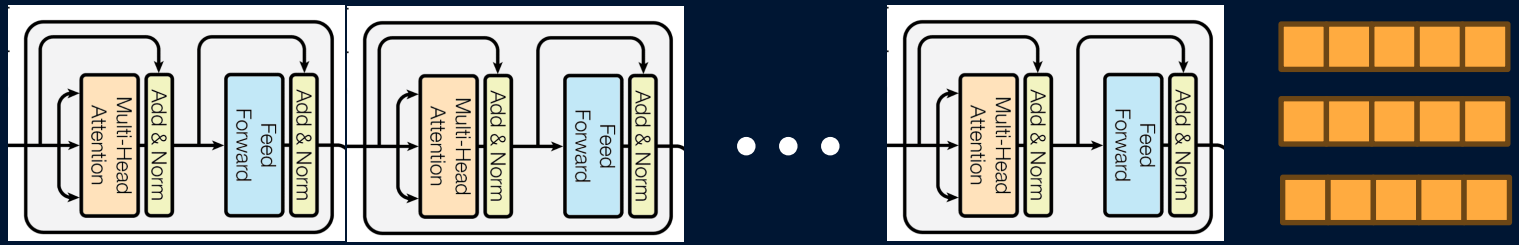
GPT

BERT

Language Model

Numerical representations (*embeddings*)

This is amazing



Produces very context-aware words **embeddings**

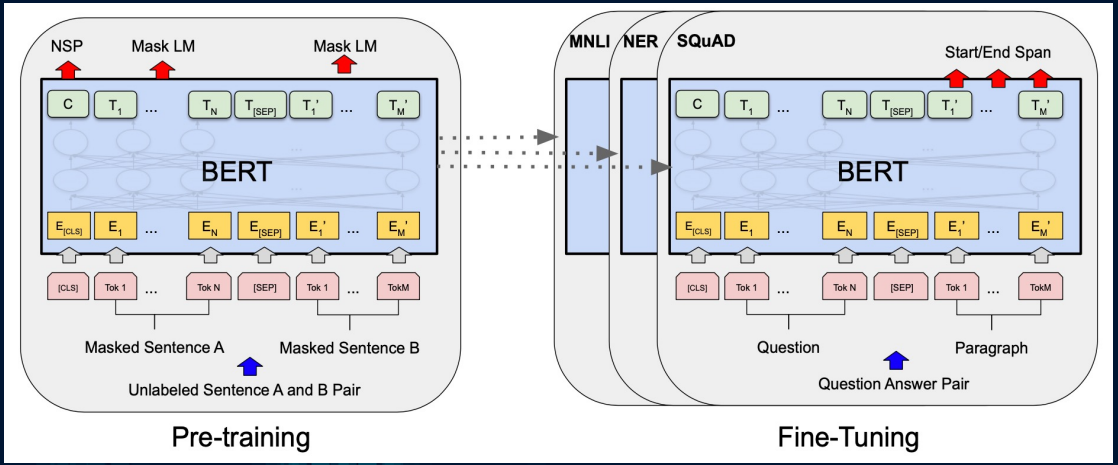
Capture **bidirectional context** within the sentence

BERT

Model training happens in 2 distinct phases

Pre-training:
NSP & MLM over large corpora
(time and resource consuming)

Fine-tuning:
downstream task
(**easy** and **fast**)



iOS Development

UIKit vs SwiftUI

- Programming languages: Objective-C ⇒ **Swift**

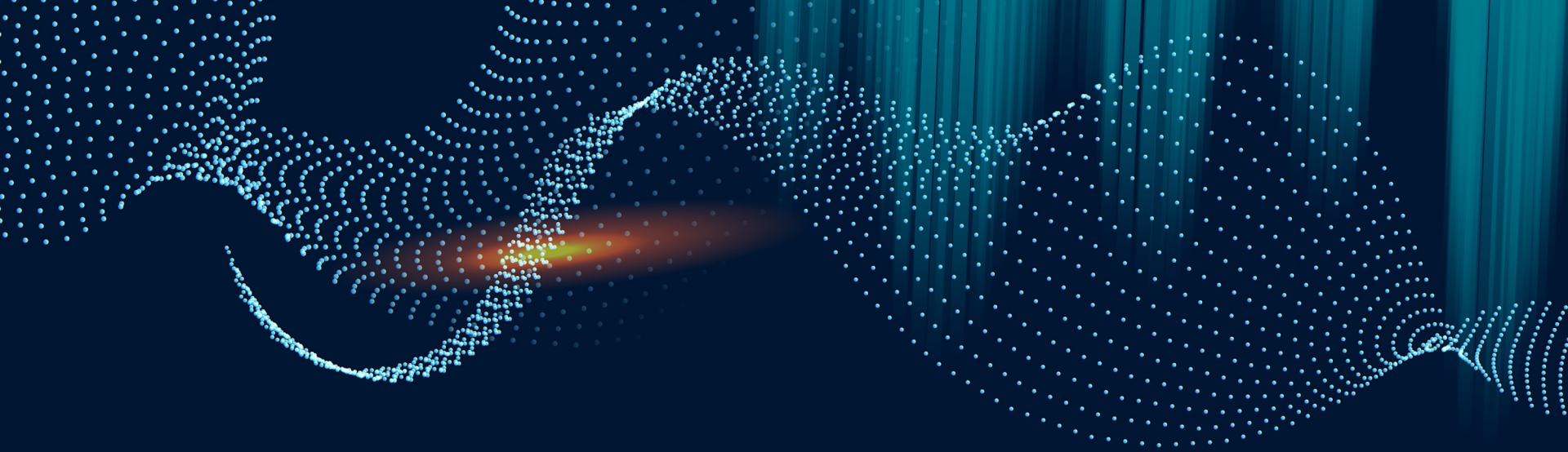
- iOS architectural frameworks:

UIKit

Imperative paradigm, based on a Model-View-Controller (MVC) architecture

SwiftUI

Declarative paradigm: describe *what* you see, rather than *how* to implement it



SOLUTION

System requirements,
design and implementation

SYSTEM REQUIREMENTS

Voice Assistant‘s main requirements

FUNCTIONAL

- **Financial operation** assistance through **voice commands**
- **Conversational interaction** for information gathering
- **Feedback** and **confirmation** mechanisms

SYSTEM REQUIREMENTS

Voice Assistant's main requirements

FUNCTIONAL

- **Financial operation** assistance through **voice commands**
- **Conversational interaction** for information gathering
- **Feedback** and **confirmation** mechanisms

NON-FUNCTIONAL

- ***Usability*** and *Accessibility*
- Low Latency
- *Privacy* and *Security*
- **On-device processing**
- Modularity

SYSTEM REQUIREMENTS

Voice Assistant's main requirements

FUNCTIONAL

- **Financial operation** assistance through **voice commands**
- **Conversational interaction** for information gathering
- **Feedback** and **confirmation** mechanisms

NON-FUNCTIONAL

- ***Usability*** and *Accessibility*
- Low Latency
- *Privacy* and *Security*
- **On-device processing**
- Modularity

EVERYTHING RUNS ON-DEVICE!

SYSTEM REQUIREMENTS

Main functional requirements

SEND MONEY

Send some money to a registered contact using a bank account

REQUEST MONEY

Request some money from a registered contact using a bank account

CHECK BALANCE

Check the current balance of a bank account

CHECK TRANSACTIONS

Check your last transactions

SYSTEM DESIGN

Voice Assistant's software components

UI/UX

Interacts with the user

SPEECH RECOGNIZER

Performs *Speech-to-Text*

TEXT CLASSIFIER

Intent detection and *Entity extraction*

DIALOGUE STATE TRACKER

Manages conversation state

RESPONSE GENERATOR

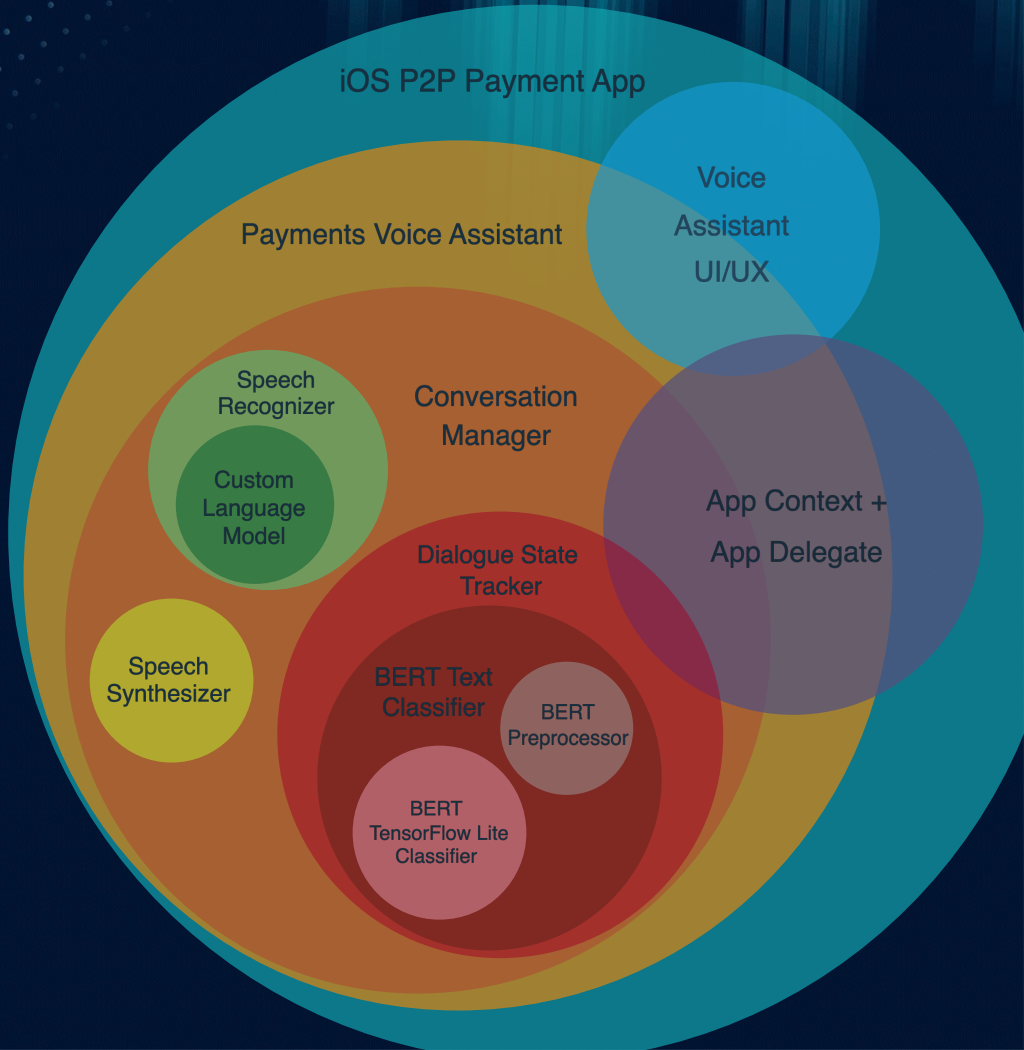
Produces textual response

SPEECH SYNTHESIZER

Performs *Text-to-Speech*

SYSTEM IMPLEMENTATION

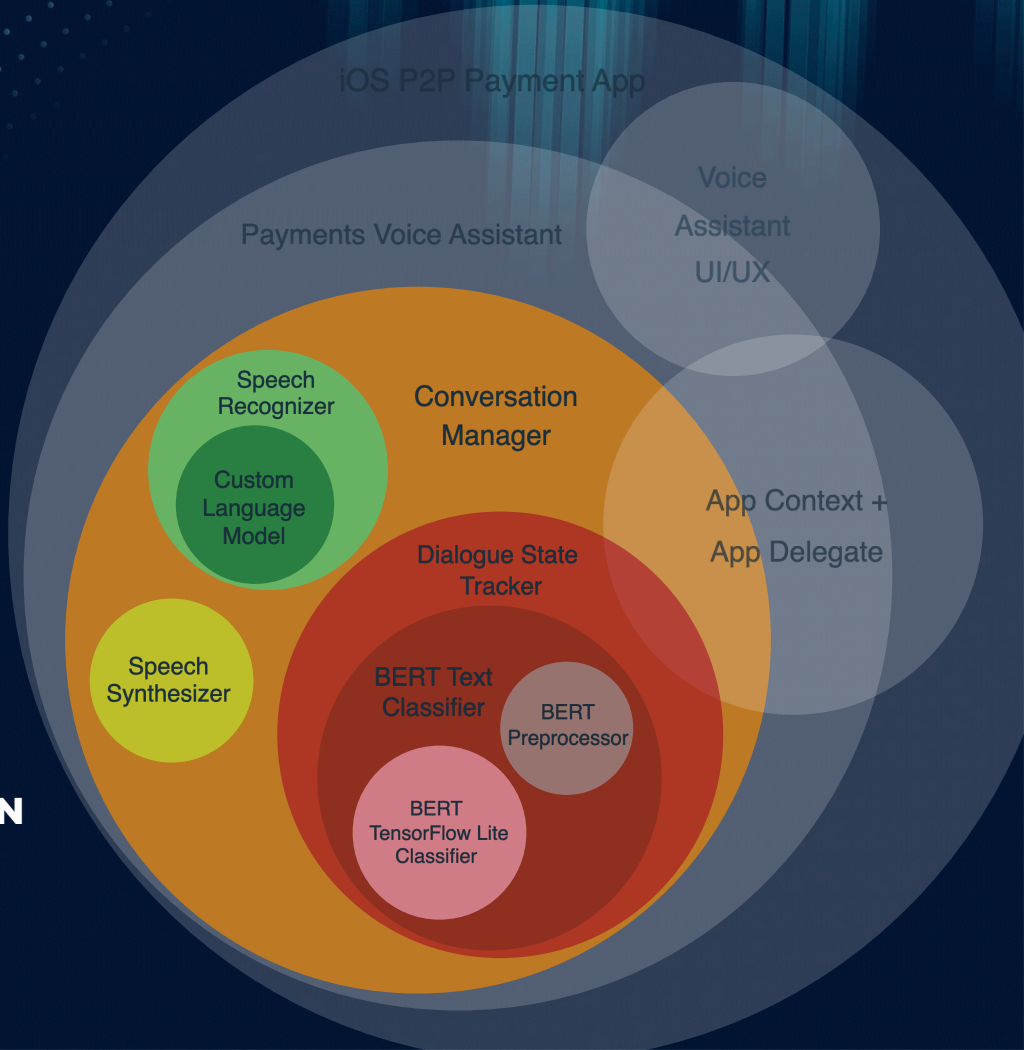
Independent software module *(iOS framework)* seamlessly integrable into any payment application



SYSTEM IMPLEMENTATION

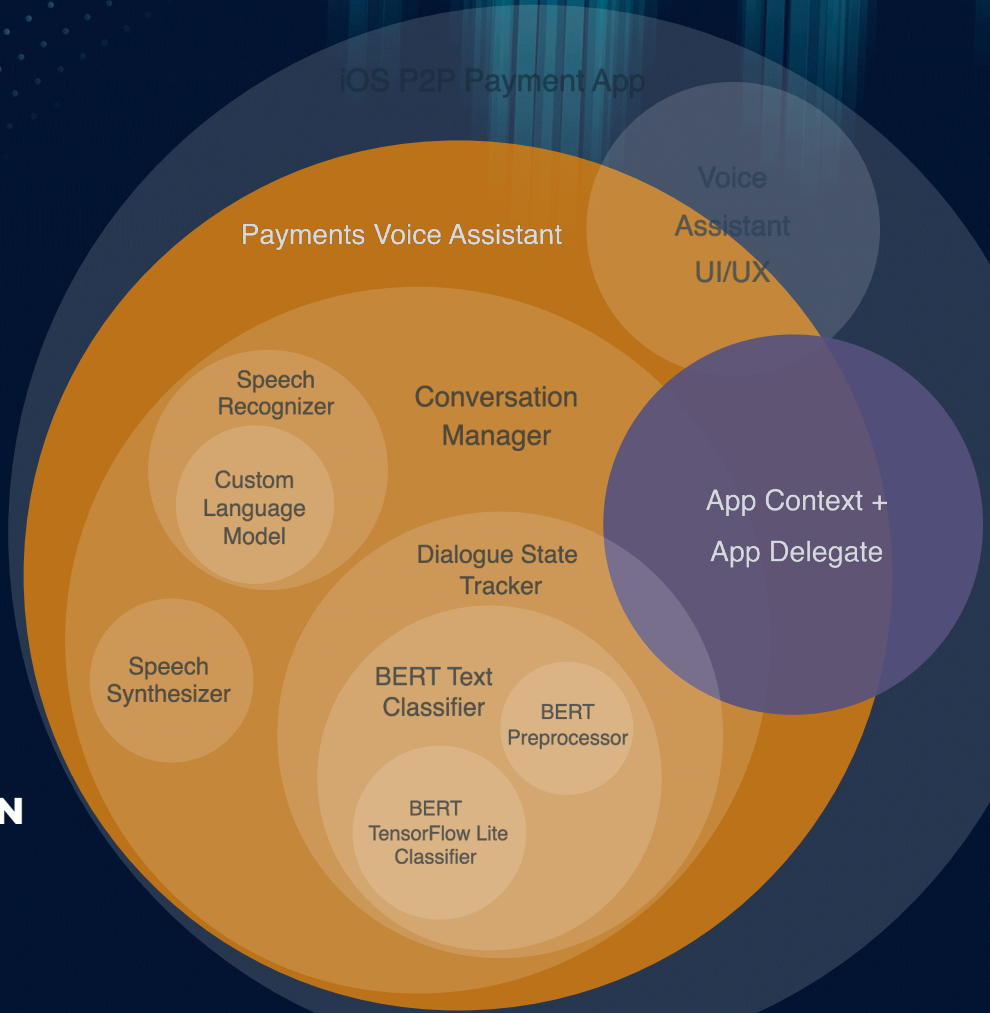
Independent software module *(iOS framework)* seamlessly integrable into any payment application

DEPENDENCY INJECTION AT LOW LEVEL AND HIGH-LEVEL



SYSTEM IMPLEMENTATION

Independent software module *(iOS framework)* seamlessly integrable into any payment application



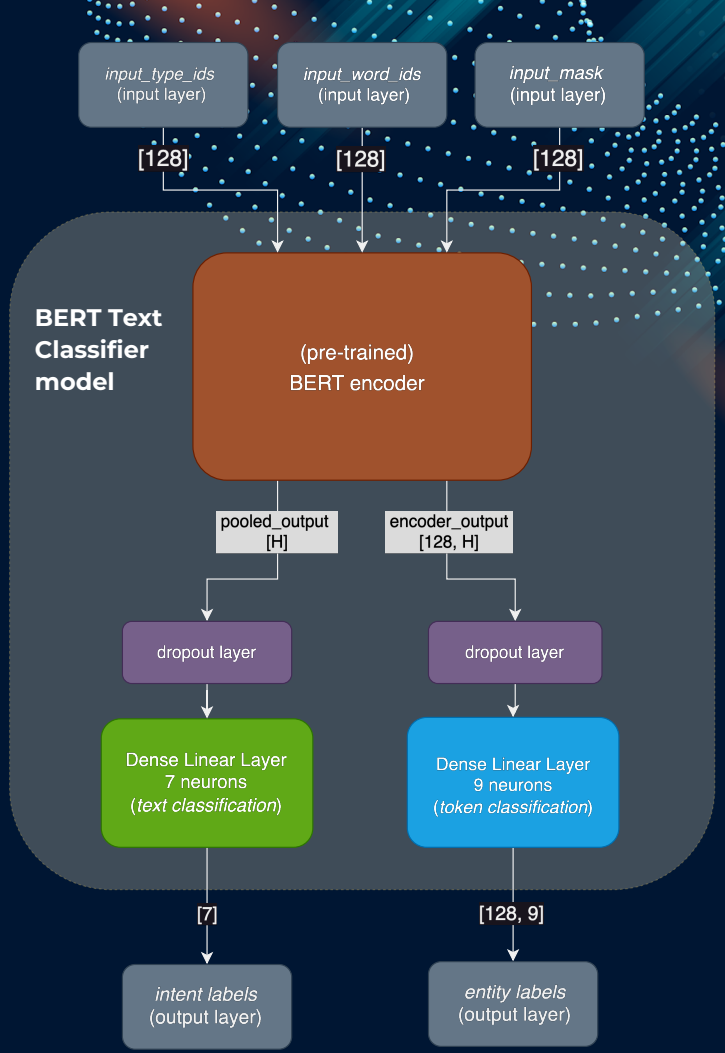
DEPENDENCY INJECTION AT LOW LEVEL AND HIGH-LEVEL

BERT Text Classifier

Core Machine Learning model

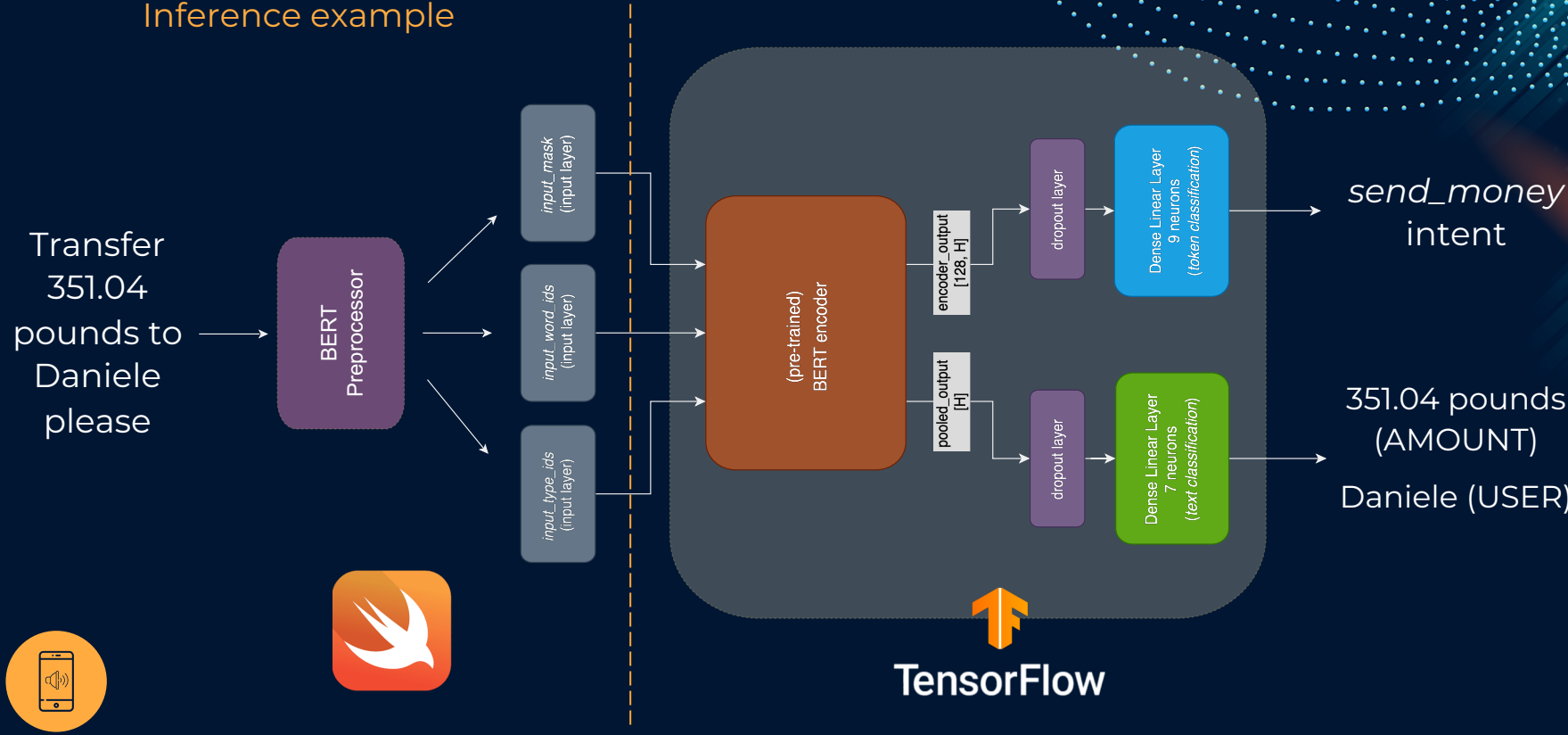
- Based on a **TensorFlow** Keras model built and **fine-tuned** in Python on *Google Colab*
- Converted in **TensorFlow Lite** for mobile deployment
- Performs both:
  - **Intent classification**
  - **Entity extraction**

TensorFlow



BERT Text Classifier

Inference example



ARTIFICIAL DATASET

Designed intents and entities

- Used to **fine-tune** the BERT Text Classifier
- Artificially produced with *templates* and *entities* generated with *ChatGPT4* ⇒ **~30k** sentences
- Designed **intents** and **entities**:

INTENT
none
check balance
check transactions
send money
request money
yes
no

BIO ENTITY LABEL
O
B-AMOUNT
I-AMOUNT
B-BANK
I-BANK
B-CURRENCY
I-CURRENCY
B-USER
I-USER

BERT Text Classifier

TensorFlow Lite conversion and quantization

- Converted and *quantized* models into **TensorFlow Lite** format
- Selected **Small BERT** configuration

BERT encoder	quantized size	on-device inference time
mini	~11 MB	~140 ms
small	**~29 MB**	**~320 ms**
medium	~42 MB	~600 ms

BERT Text Classifier

TensorFlow Lite conversion and quantization

- Converted and *quantized* models into **TensorFlow Lite** format
- Selected **Small BERT** configuration

BERT encoder	quantized size	on-device inference time
mini	~11 MB	~140 ms
small	**~29 MB**	**~320 ms**
medium	~42 MB	~600 ms

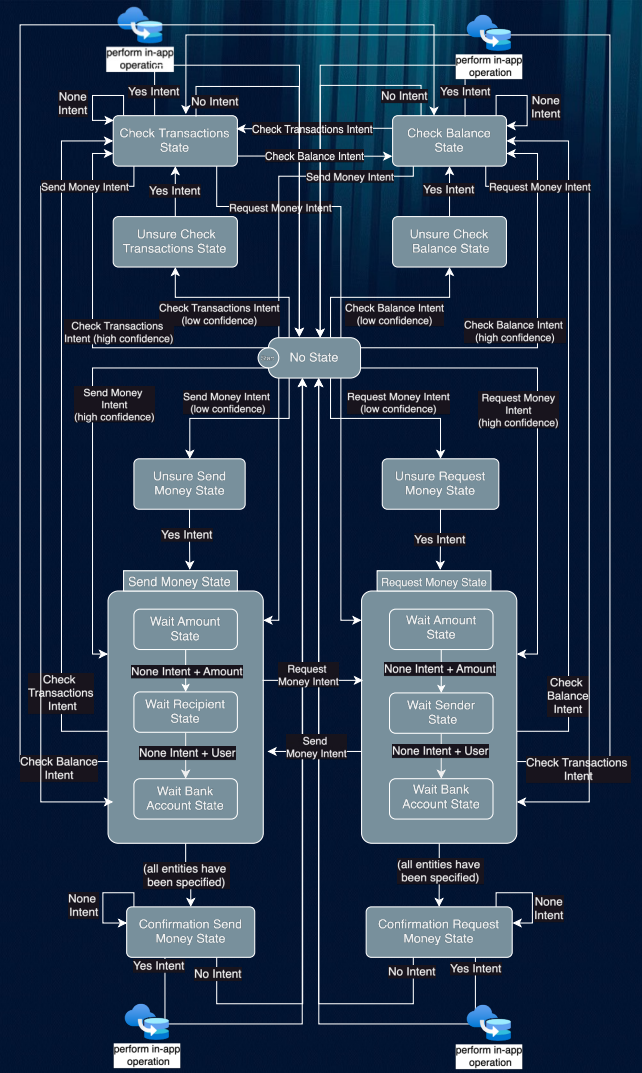
100%
Intent recognition test accuracy

99.99%
Entity extraction test accuracy

EVALUATION RESULTS

DIALOGUE STATE TRACKER

- Keeps track of the **conversation state**
- Implemented as a **Finite-State Machine**
- *State Pattern* adopted
- **Matches** detected *entities* with app specific information



SPEECH RECOGNIZER

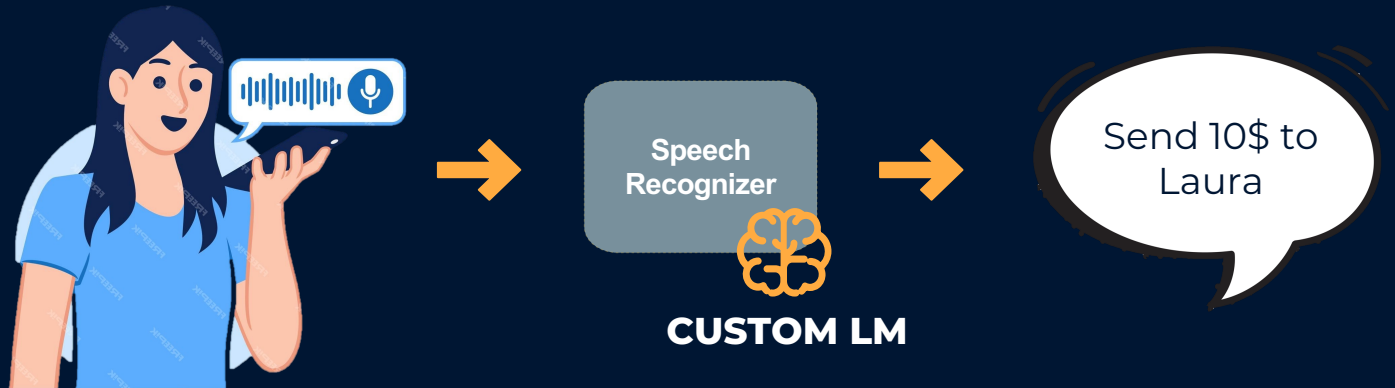
Enhanced with a Custom Language Model

SPEECH-TO-TEXT ON-DEVICE

Using iOS *Speech* framework

CUSTOM LANGUAGE MODEL

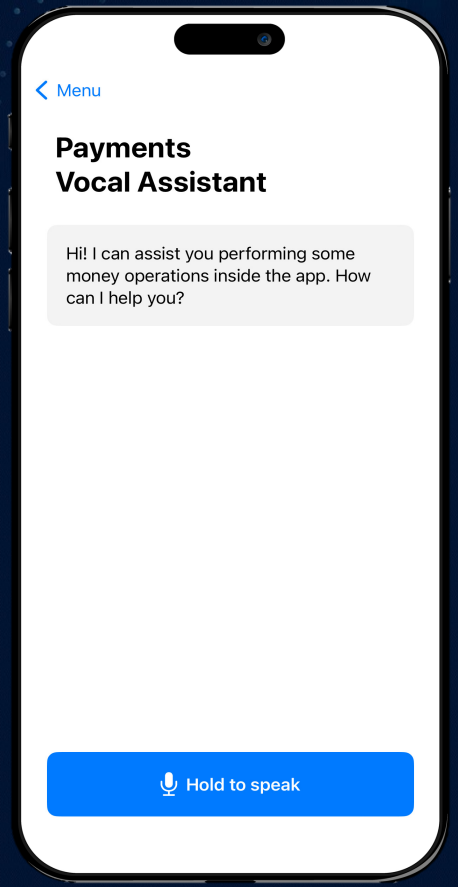
Leveraged new iOS 17 APIs to enhance SR with application domain data

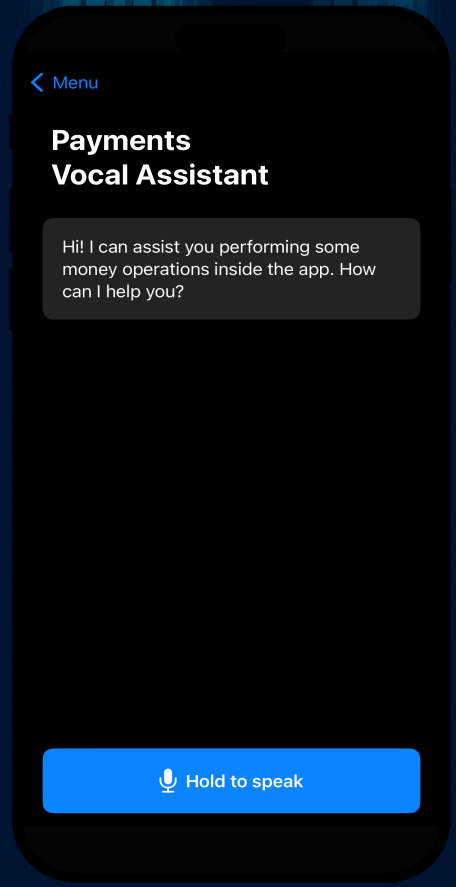


VOICE ASSISTANT UI/UX

SwiftUI View

- **Simple** and **effective**
- Both *white mode* and *dark mode*

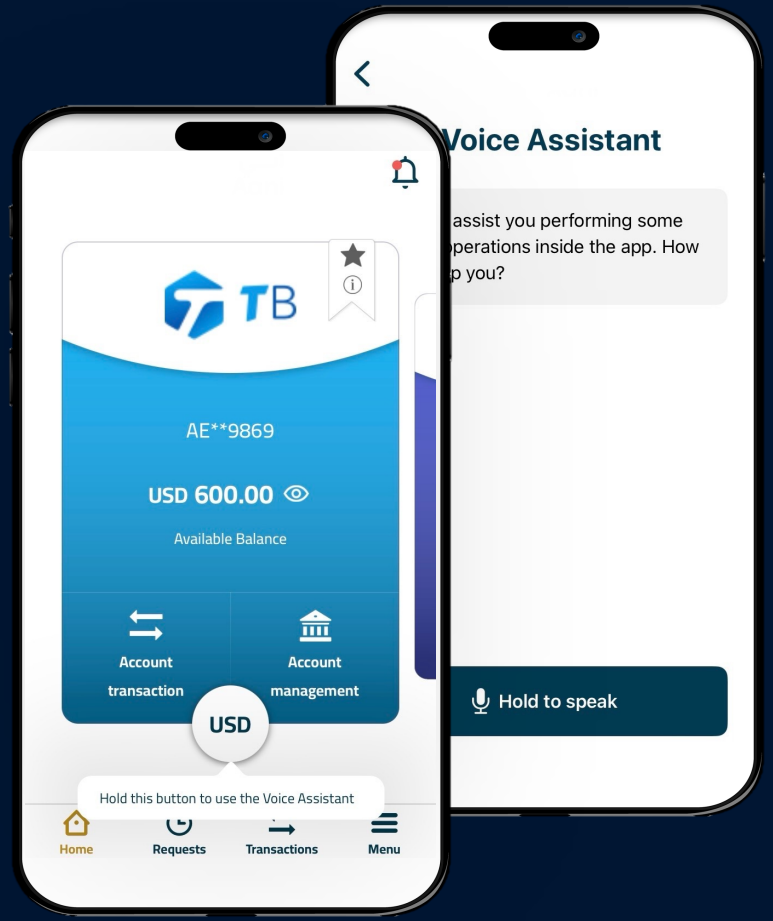


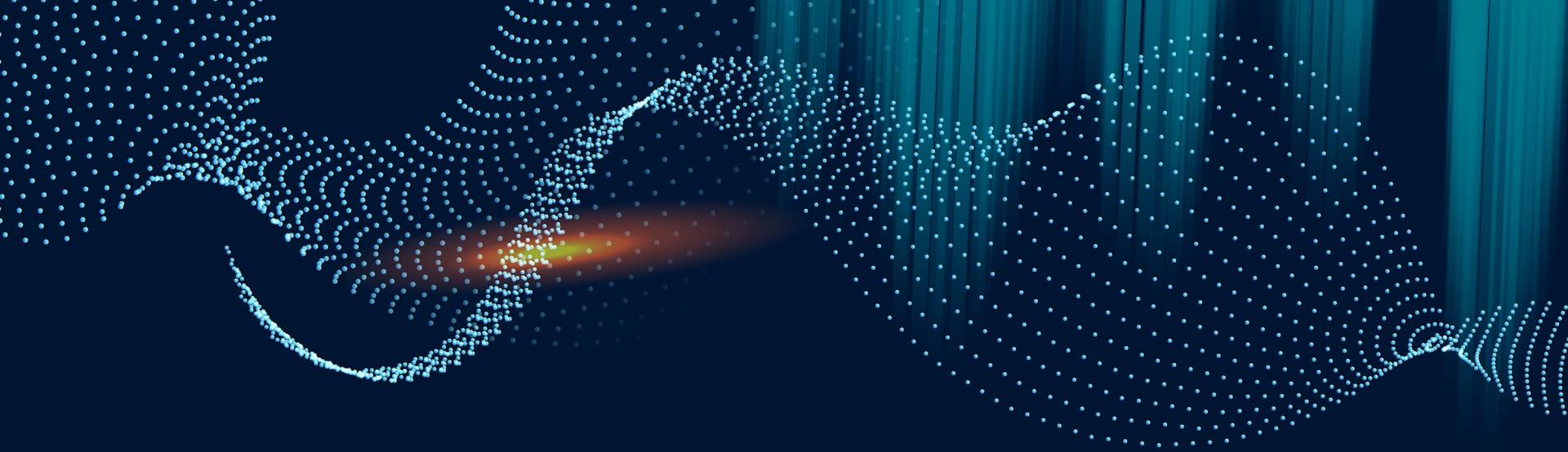


VOICE ASSISTANT INTEGRATION

Into ***case study*** P2P Payment app

- Voice Assistant **SwiftUI View** integrated into *UIKit* based app
- Injected:
  - **App Context**: user's *contacts* and *bank accounts* info
  - **App Delegate**: to perform in-app operations





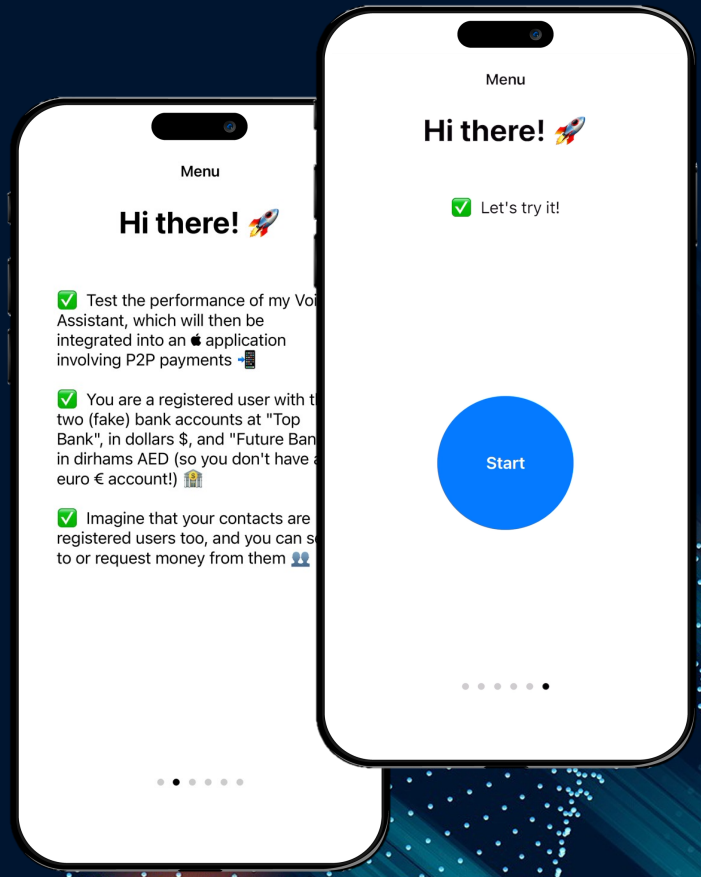
EVALUATION

Test app deployment &
user-based evaluation phase

USER EVALUATION

TestFlight app deployment

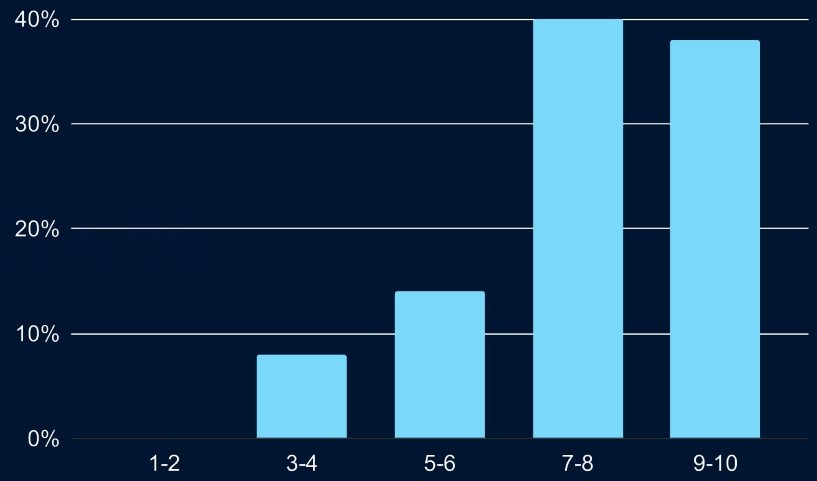
- Test iOS app released on ***TestFlight***
  - *Case study app* could **not** be used for extensive testing
- **50 user testers** tried the Voice Assistant
- User feedback gathered by means of an anonymous **Google Form**, focused on:
  - *Usability*
  - *Effectiveness*
  - *Sense of Security*



GOOGLE FORM RESULTS

User overall satisfaction

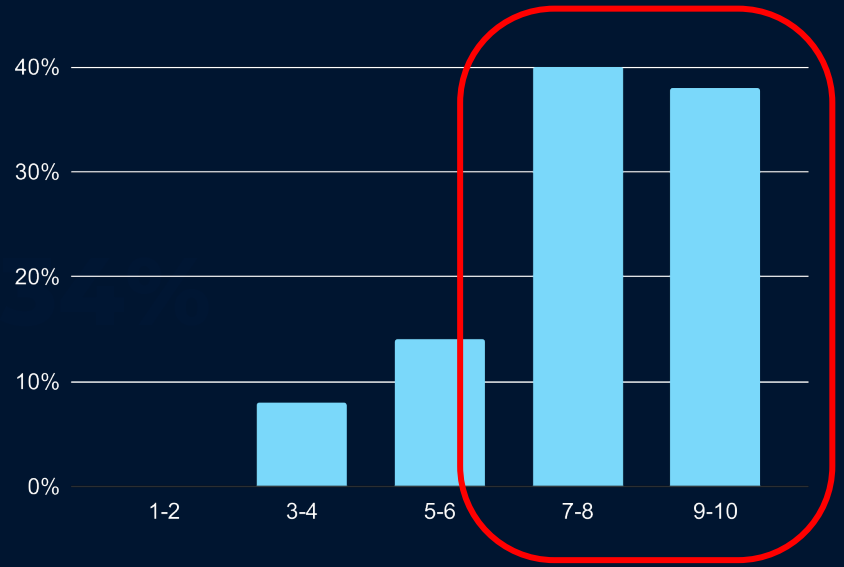
*How **satisfied** are you with the **overall experience** of using the Voice Assistant?*



GOOGLE FORM RESULTS

User overall satisfaction

*How **satisfied** are you with the **overall experience** of using the Voice Assistant?*



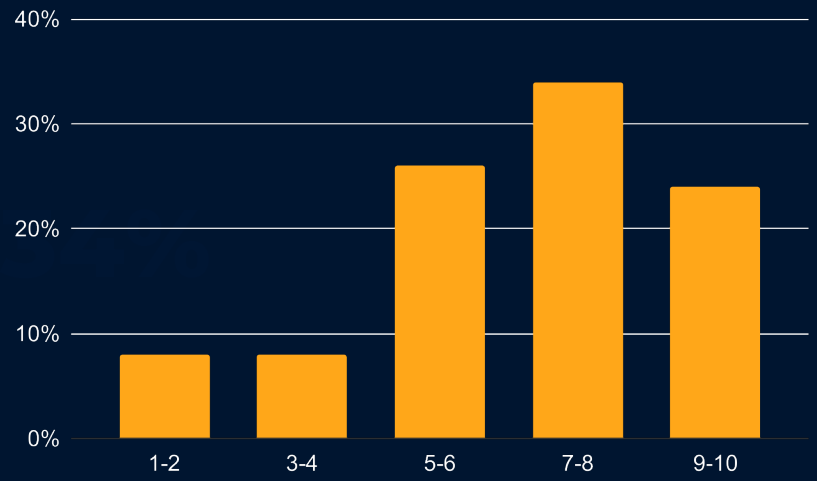
+78%

user overall satisfaction

GOOGLE FORM RESULTS

Eventual future usage

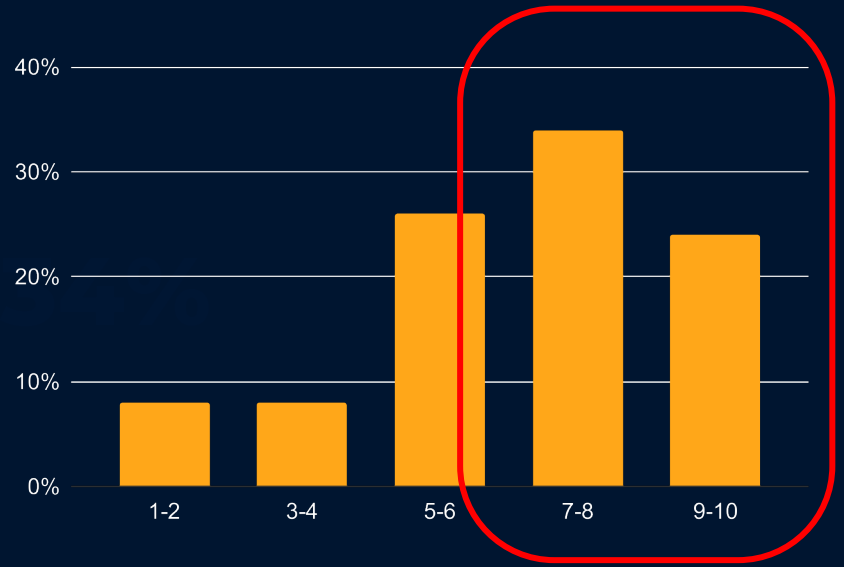
*How likely would you be to **use** the Voice Assistant for future money operations in a **real application**?*



GOOGLE FORM RESULTS

Eventual future usage

*How likely would you be to **use** the Voice Assistant for future money operations in a **real application**?*



+58%

would use it in a real app

POSSIBLE IMPROVEMENTS

Major issues

VOICE NATURALNESS

Enhance the quality of the Voice Assistant's voice

MULTI-LANGUAGE

Support more than English language

BETTER ENTITY EXTRACTION

Improve the Named Entity Recognition

MORE CAPABILITIES

Increase the operations supported by the assistant

CONCLUSIONS

Key findings

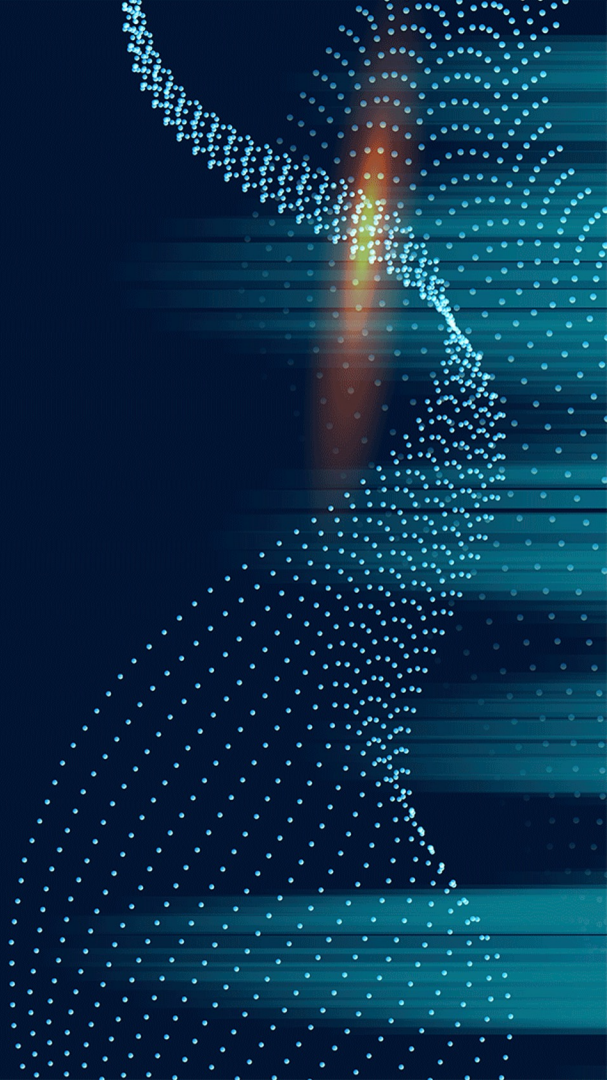
- **Voice-enabled features** enhance *interaction* and *naturalness* of financial services
- **AI technologies** can have a great impact on **mobile apps** *User eXperience (UX)*, maintaining *security* and *privacy* with **on-device processing**
- Still room for improvements in terms of *naturalness* and *effectiveness*

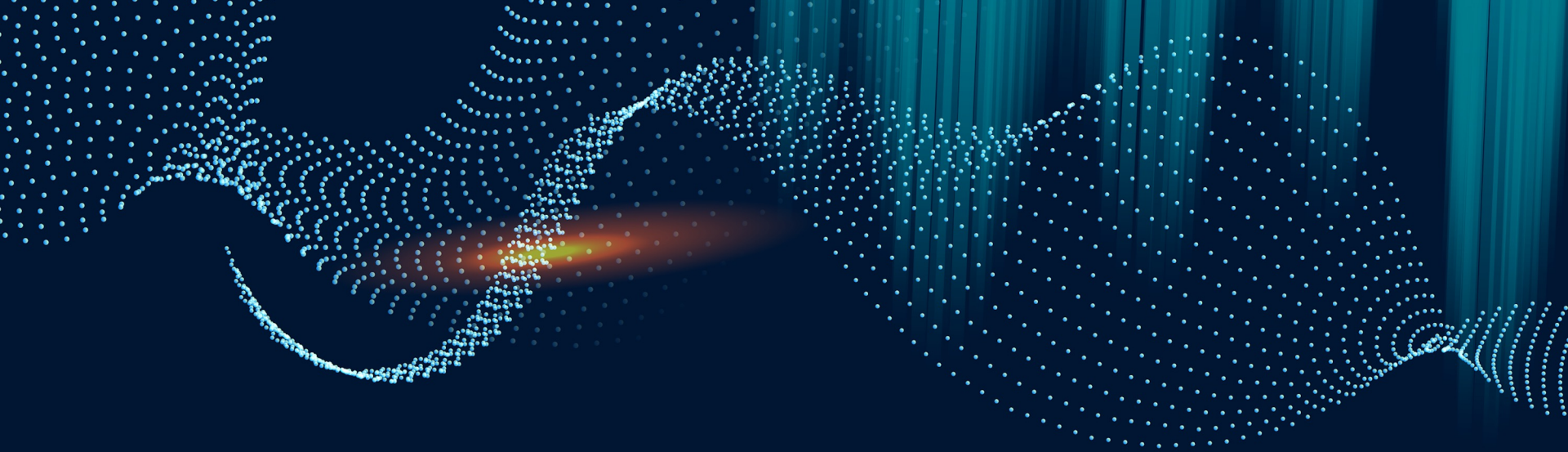
THANKS!

Do you have any questions?

Mario Mastrandrea

CREDITS: This presentation template was created by Slidesgo, including icons by Flaticon, and infographics & images by Freepik.

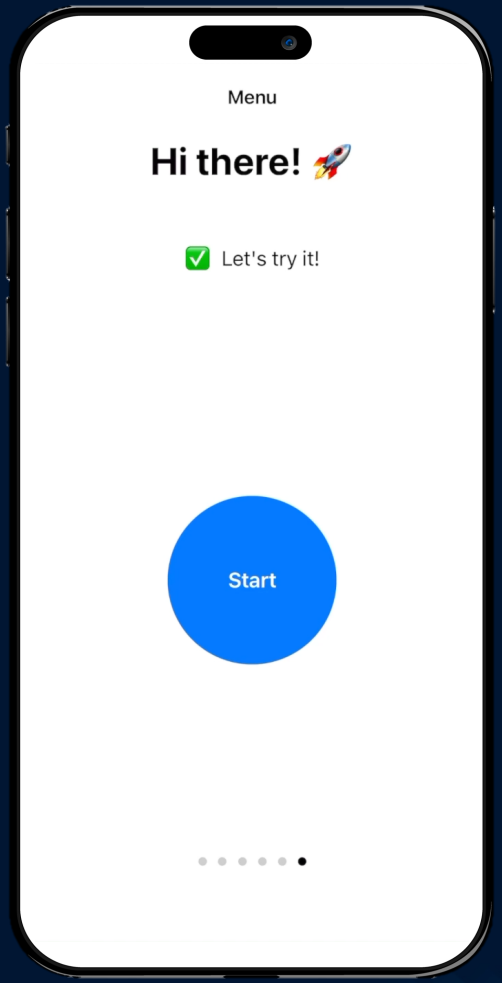




EXTRA SLIDES

For Q&A

LIVE DEMO

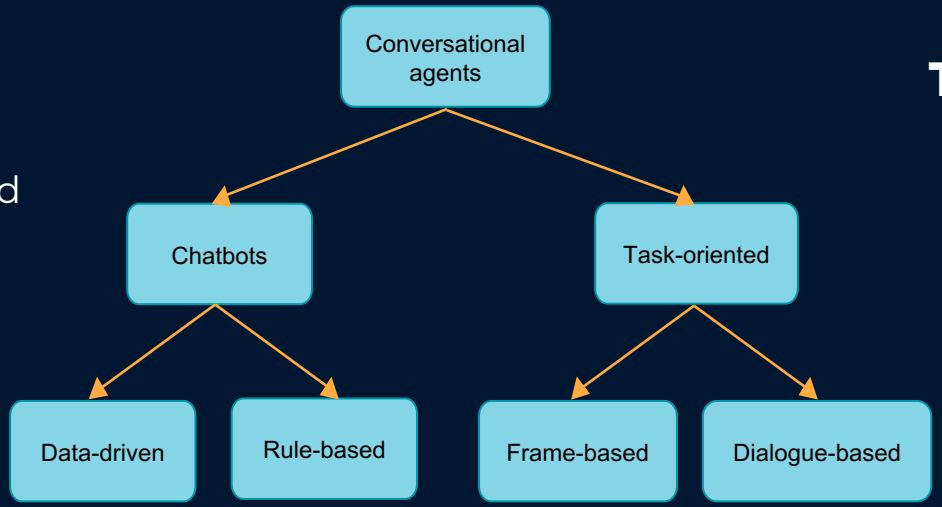


CONVERSATIONAL AGENTS

Chatbots vs Task-oriented dialogue systems

Chatbots

Mimic unstructured human-human conversation



Task-oriented

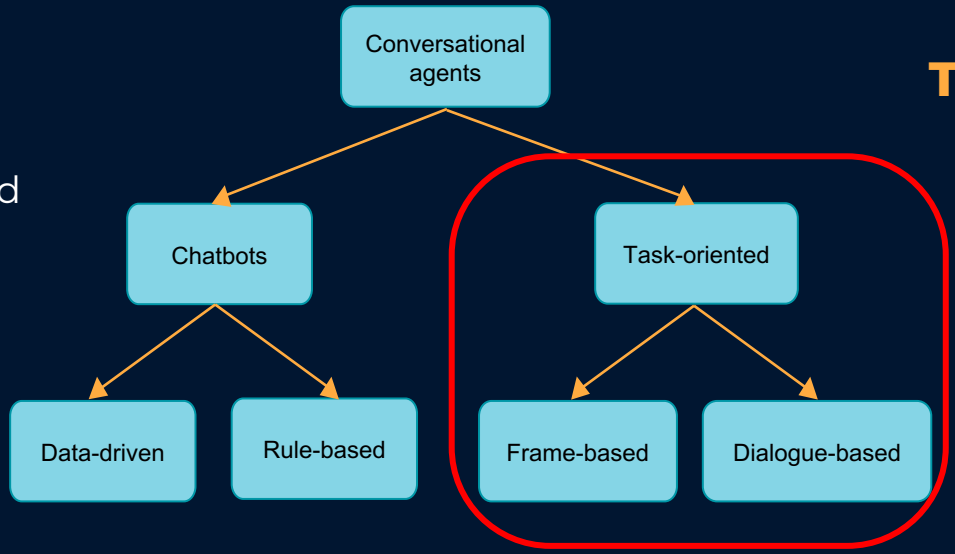
Goal-based agents used to solve tasks

CONVERSATIONAL AGENTS

Chatbots vs Task-oriented dialogue systems

Chatbots

Mimic unstructured human-human conversation



Task-oriented

Goal-based agents used to solve tasks

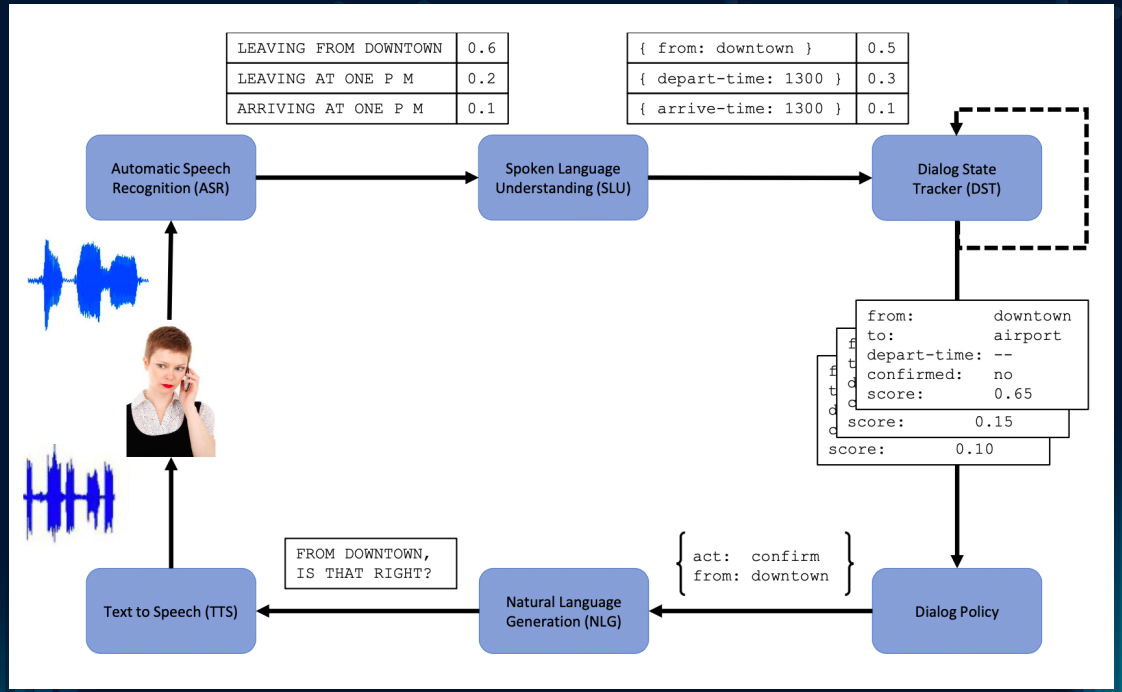
DIALOGUE STATE MANAGEMENT

Finite-State Machine

SYSTEM DESIGN

Voice Assistant's overall system

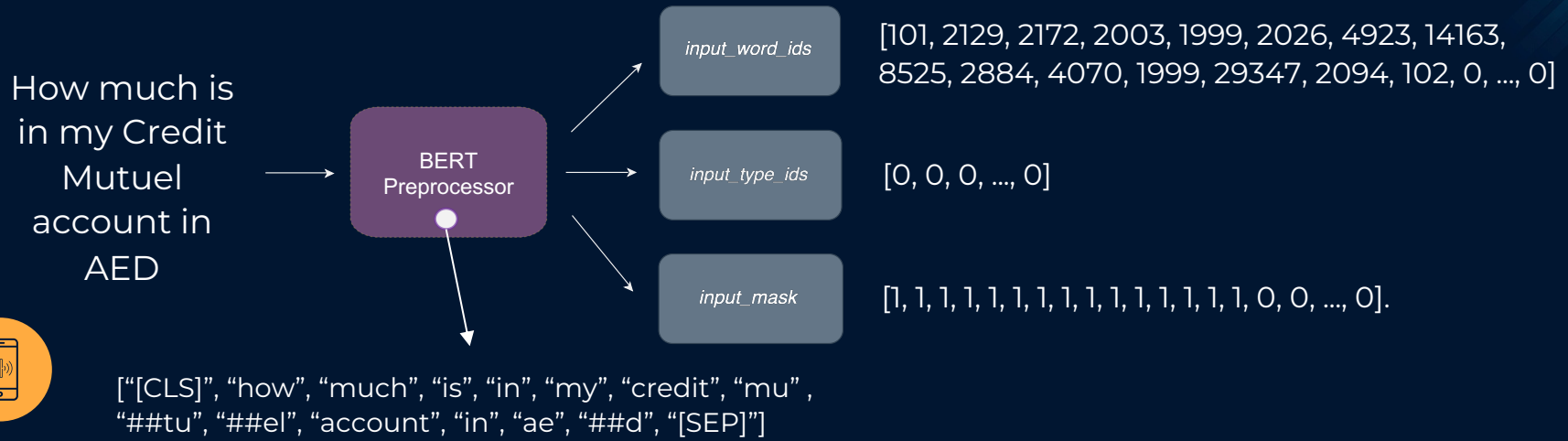
In-between ***frame-based*** and ***dialogue-based*** conversational agent



BERT PREPROCESSOR

Implemented in Swift

- Transforms an input sentence into suitable BERT input arrays:
  - *input_word_ids*: numerical tokens
  - *input_type_ids:* indicate tokens' sentence (0/1)
  - *input_mask:* indicate if token is padding or not (0/1)



BERT Text Classifier

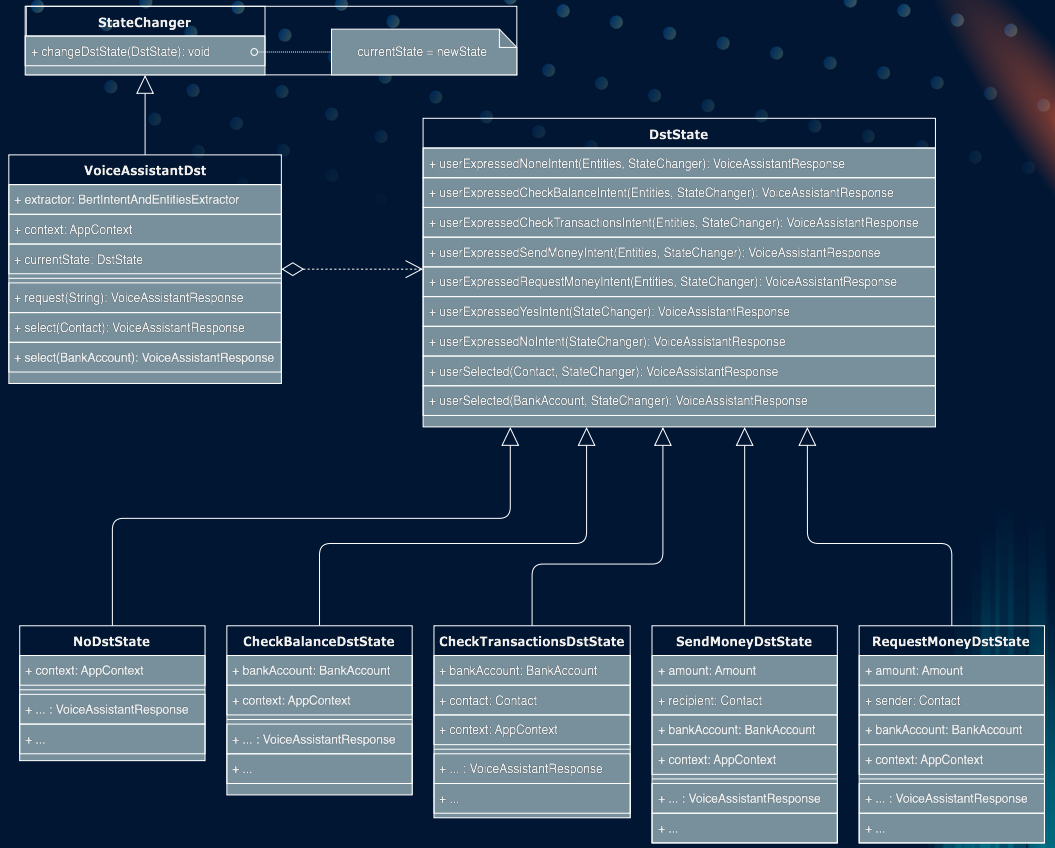
Fine-tuning and validation

- 75%/25% train-test split
- **Validation** (20% split) of different model configurations ⇒ *greedy approach*
- Best models' configurations results:

BERT encoder	Train time	learning rate	batch size	dropout	val. loss	intent accuracy	entity accuracy
mini	~15 min	5e-5	16	0.3	1.414e-3	0.99956	0.99987
small	~35 min	5e-5	16	0.2	8.494e-5	1.00000	0.99999
medium	~60 min	5e-5	16	0.2	3.949e-5	1.00000	1.00000

STATE PATTERN

- As introduced by the **Gang of Four** in *”Design Patterns: Elements of Reusable Object-Oriented Software”*
- Applied to implement the **Dialogue State Tracker**



CUSTOM LANGUAGE MODEL

New iOS 17 APIs

- **Fine-tuned** the iOS underlying *Speech Recognition* model with application domain sentences
- Used **templates** from the artificial dataset and customized at *runtime* with user's specific information (*bank accounts*, *contacts*) ⇒ ~90k sentences

TEMPLATE EXAMPLE
What are the latest transactions with <name> <surname>
Display my <bank> account balance
Could you assist in receiving 334 dollars and 32 cents from <name>