KATbot

Team B0:
Ashika Koganti, Abha
Agrawal, Jade Traiger

Application Area

Storytelling robot that interacts with people to aid in language and reading comprehension

- Merging AI with educational tools
- Target Audience: early elementary school age children
- Child-friendly user experience

Hello!

Solution Approach

Speech Processing & Text to Speech: Convert speech to ML input and ML output to speech

1. Text to speech dialogue prompts user for input
2. User speech is processed and sent to the ML model
3. ML model returns the rest of dialogue

Robot: Custom-made robot inspired by Japanese lucky cats

1. Robot houses all electronics needed for project
2. 2x one degree of freedom robot arms
3. Text display to display current sentence
4. Eye displays

Solution Approach

Machine Learning: receive user's input word, output sentence by sentence to TTS

1. Start with manually configured template, keywords removed
2. Prompts user for part of speech
3. User input goes through error detection and grammar correction
4. Algorithm predicts dependent words to customize the story

System Diagram

Story Generation Model

User Input
- TTS for beginning of sentence
- Prompted with part of speech
- command word for help

Part of Speech detection
- identify PoS with nltk package
- obvious grammar mistakes ignored

Error
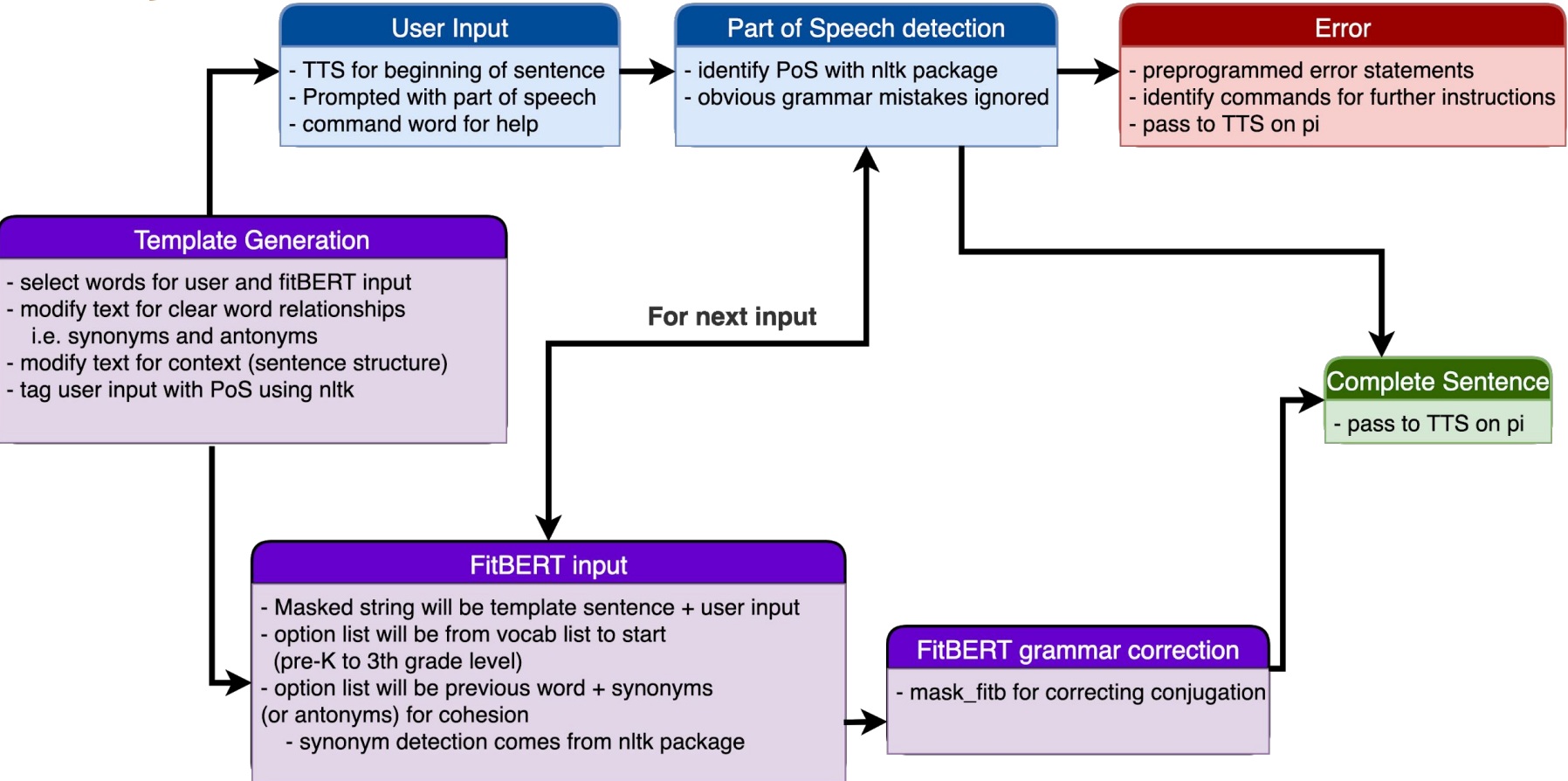- preprogrammed error statements
- identify commands for further instructions
- pass to TTS on pi

Template Generation
- select words for user and fitBERT input
- modify text for clear word relationships
    i.e. synonyms and antonyms
- modify text for context (sentence structure)
- tag user input with PoS using nltk

For next input

Complete Sentence
- pass to TTS on pi

FitBERT input
- Masked string will be template sentence + user input
- option list will be from vocab list to start
    (pre-K to 3th grade level)
- option list will be previous word + synonyms
(or antonyms) for cohesion
    - synonym detection comes from nltk package

FitBERT grammar correction
- mask_fitb for correcting conjugation

Implementation Plan

Machine Learning Storytelling
- Templates from Aesop's Fables (177 stories)
- NLTK - natural language processing speech package
  - Part of speech tagging
  - Synonym generation and recall
- FitBERT - 'Fill in the blanks' BERT
(Bidirectional Encoder Representations from Transformers)
  - Sentence prediction
  - Grammar correction
- Laptop for MVP, aim to put it on Nvidia Jetson Nano

Speech Processing & Text to Speech
- Conference Microphone / USB Speakers
- Python Speech Processing Package with PocketSphinx
- Python gTTS
- Create a friendly voice by pitch shifting with PSOLA
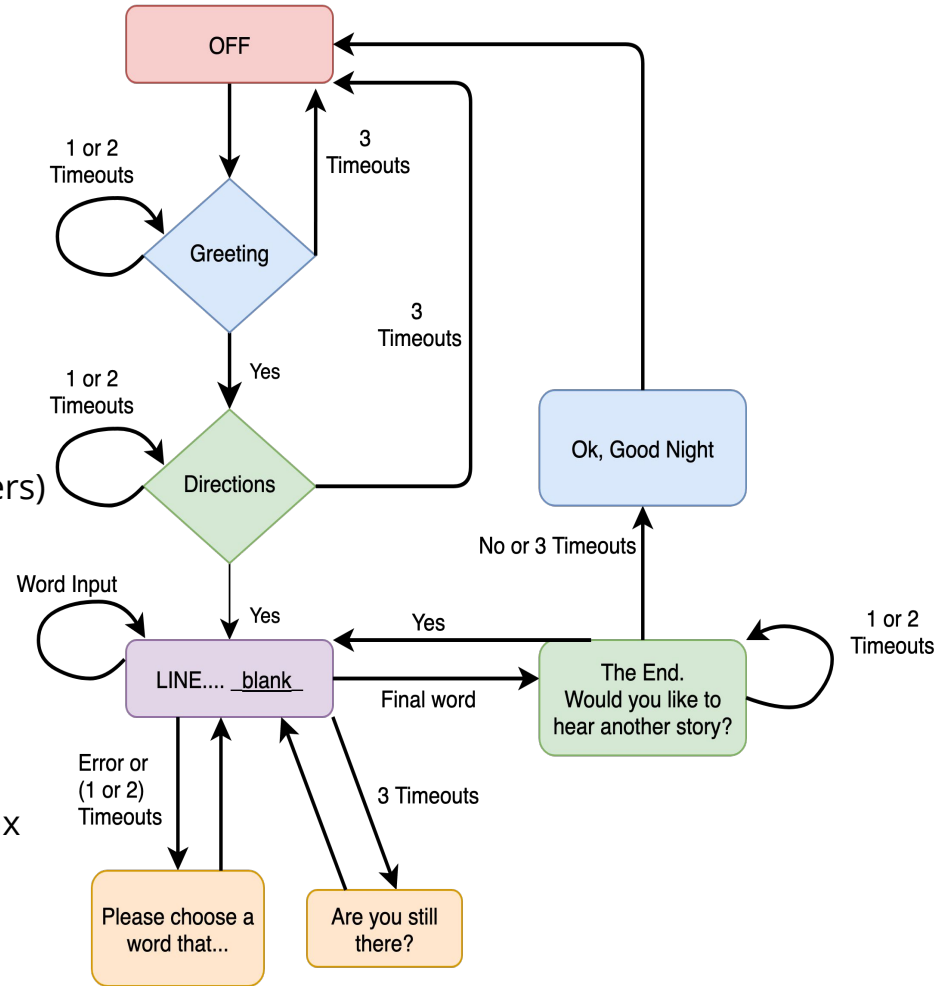
Implementation Plan

Custom-made Robot

- Laser-cut acrylic frame for support with 3D-printed shell for aesthetics
- Body dimensions: 8" x 8" x 10"
  Head dimensions: 6" x 6" x 9"
  - Houses Raspi, batteries, displays, cables, etc
- 2x 1-DoF Robot Arms
  - Dimensions: 1.5" x 1.5" x 6"
  - Servo motors provide enough torque to move weight of acrylic/PLA arm

Robot Design and Dimensions
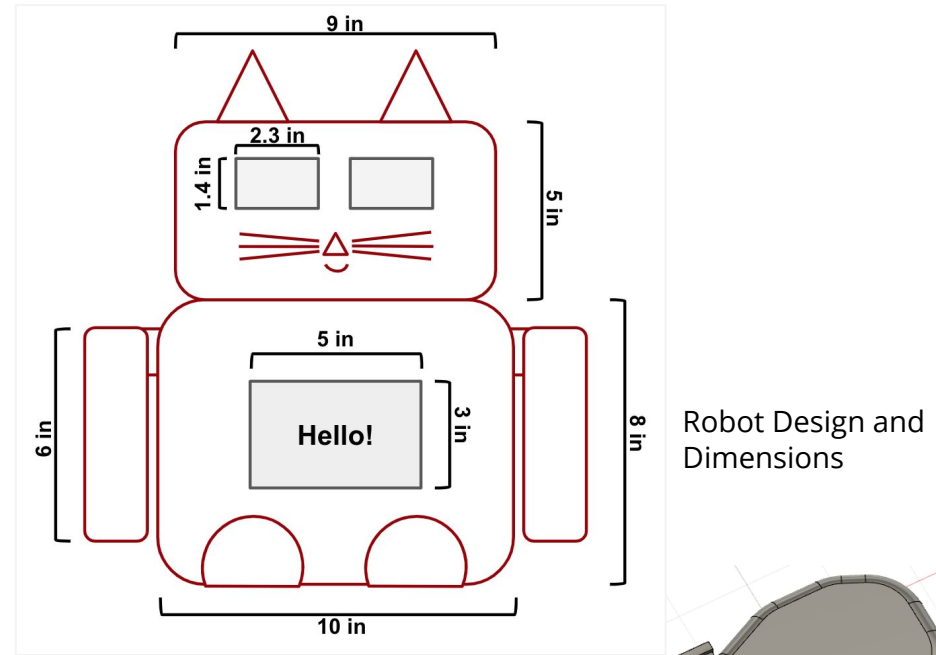
CAD of Robot Arm Frame
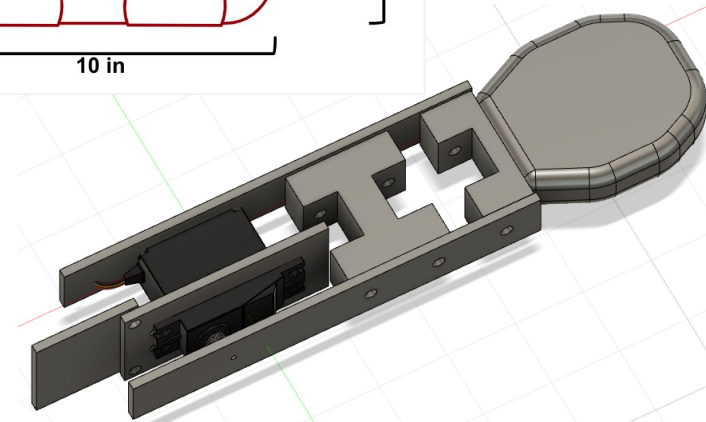
Metrics and Validation

Description	Goal	Verification Method
Part of Speech Error Detection	90% accuracy	SW Testing - Test Dataset
Synonym Recall	85% accuracy	SW Testing - Test Dataset
Speech Processing Accuracy	15% Word Error Rate	Measure decoding errors
System Latency	4 - 6 sec	Time user i/p to speech o/p
Power	30 - 45 min	User testing

Metrics and Validation

Description	Goal	Verification Method
Story Cohesion	Cohesion level falls between original stories and random stories	User survey - grade three types of stories based on 5 variables: *Logical Sense, Themes, Genre, Narrator, Style*
User Satisfaction	- Liked the stories (87.5%) - Wanted to play again (100%) - Robot was friendly (87.5%) - Robot's stories were interesting (87.5%) - Robot's stories were understandable (100%)	User Survey

Risk Management

Component	Risk Factor	Backup Plan
Story Creation	Poor cohesion, Poor fill in the blank choices	Reduce number of user/FitBERT inputs in story templates
Speech Recognition / TTS	Both rely on internet connection	Have local speech recognition and TTS capable packages

Project Management

TASK TITLE	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13
Dates	2/3/2020	2/10/2020	2/17/2020	2/24/2020	3/2/2020	3/9/2020	3/16/2020	3/23/2020	3/30/2020	4/6/2020	4/13/2020	4/20/2020	4/27/2020
Important deadlines	Project Proposal			Design Presentation		Spring Break			Design Demo			Design Demo	Final Presentation
Team Self-Education													
Define detailed metrics													
Research												Abha	
--> Audio input/output												Ashika	
--> Speech processing package												Jade	
--> Story generation pipeline + algorithm												Abha+Ashika	
--> Robot design												Abha+Jade	
--> Communication protocols												Ashika+Jade	
Evaluate solution approaches												Everyone	
Design & Build MVP Parts													
System Architecture Diagram													
System interaction design													
Hardware Benchmarks and Metrics													
Software Benchmarks and Metrics													
Write Design Document													
Prepare for Design Presentation													
Bill of Materials/Order Parts													
Analog input/output schematics													
Audio input implementation													
Speech Processing on RPi													
Text to speech on RPi													
Communication Protocols													
Face Display													
Robot arms													
Path Planning for arms													
Robot Shell and Aesthetics													
ML dataset gathering													
Part of speech tagging													
--> Error detection for user input													
--> Synonym detection													
--> Template making													
Fill in the blank													
ML story generation unit testing													
Integration and Testing													
Audio input to ML algorithm													
Robot displays text from ML algorithm													
Robot text to speech													
Gesture evaluation													
Design Evaluation & Improvement													
Whole system testing													
Design evaluation													
Improvements													
Final system testing													
Prepare for Final Presentation													