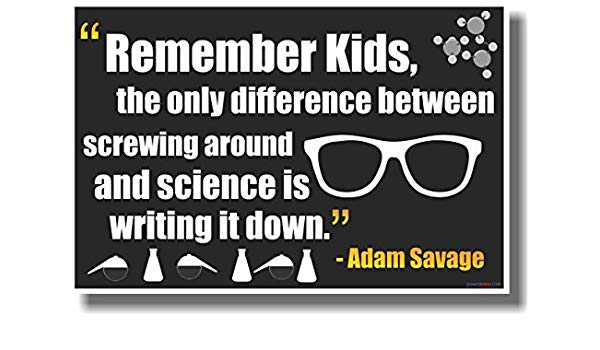
Keuzedeel   
Special Input/Output

Onderzoek



Team #

Ruben Steenbrink

|  |
| --- |
| *Keuzedeel Special Input - Output*  *D1-K1-W1: Onderzoekt nieuwe vormen van input en/of output* |

Datum: 2019-05-09

Template v1.1

# Inleiding

Dit Document bevat alle data dat te maken heeft met het speech controller project. Hier in word de uitvoering bestande uit: research, bronnen, etc. En resultaat het uiteindelijke uitkom van de uitvoering.

# 

# Uitvoering

Het originelen idee was dat het project gebruik ging maken van een tekst naar spraak programma, zoals kaldi en sphinx dit was helaas niet mogelijk, dit is omdat beide een AI zijn die getraind moet worden en meerderen resources nodig hebben die te veel tijd nodig zouden hebben.

Het is daarom besloten om het proces eenvoudiger te maken, eerst was dat om een online browser based programma bij de naam speechnotes te gebruiken, hier ben ik nooit aan gekomen. Ik ben daarna meer gaan kijken naar hoe ik de tekst die uit het tekst naar spraak zou kunnen laten registreren als computer input. Ik heb, met de hulp van mijn leraar Berend, op de node-key-sender libary gekomen. Ik daarna de node chat-example master genomen en extra code ervoor geschreven, deze code maakt het zo dat wanneer er bijvoorbeeld ‘up’ wordt ingetypt de up key wordt geregistreerd en verstuurd naar de servers OS(besturingssysteem (In dit geval windows. Door de server te activeren met powershell, kunnen mensen met het Ip adres van de computer, die het programma rent, in de chat komen en de voor gedefinieerde commands gebruiken om commands naar de server computer te geven. De input wordt door een switch gecheckt, deze switch gebruikt regular exspressions, zoals /start/i en de test functie om true of false door te geven en dan de passende code uit te voeren.

Ook om te kunnen weten of de persoon hun string bedoelen voor input gebruikt het script een if loop voor de switch de bestaat uit een indexof(“/”) hiermee kunt de server tussen scheid maken tussen input en normalen spraak.

Herkansing:

Voor de herkansing van dit project heb 1 extra week gekregen, het plan van deze week is te zorgen dat het input gedeelte van het project werkt. Dit betekent dat ik moet gaan kijken naar een manier om de input die ik nu heb naar de gefocuste window te sturen.

Het is tot mijn attentie gekomen dat de node-key-sender libary een wrapper voor een jar file is en een complete java environment nodig heeft om te werken, ik heb hier zelf niets van gemerkt, maar ik ben wel door deze persoon naar een alternatieve methoden gewezen, deze methoden Robotjs dit kan net zoals node-key-sender input versturen. Ik zou het moeten testen om te kunnen weten of dit op games werkt. UPDATE: Robotjs kan niet geïnstalleerd worden door dat het geen gyp binding kan laden, ik heb dit geprobeerd optelossen. Een van deze pogingen was om de afhankelijkheden apart te downloaden, dit gaf aan dat een van mijn visual studio files corrupt was, ik heb deze gerepareerd en alles weer opnieuw geprobeerd alles ging goed totdat ik Robotjs wou instaleren. Ik heb hier de hulp van mijn leraar Richard voor gevraagd maar zelfs met de extra hulp kwamen we nergens, ik heb uitgevonden dat dit al een probleem is sinds 2015. Dankzij de hulp van mijn leraar Berend weet ik wat het probleem is, robotjs gebruikt een programma dat niet in de nieuwste versie van node zat, om dit optelossen heb ik node 10 gedownload en hiermee heb ik Robotjs kunnen installeren.

Met de hulp van mijn leraar Berend ben ik er ook achter gekomen waarom games de input niet konden lezen, dit kwam omdat ze naar keyDown events luisteren, maar node-key-sender verstuurd een keyUp event naar de computers OS. Ook heb gevonden dat de emulator die ik wou gebruiken voor de test alleen luistert naar een geregistreerde controller(voorbeeld: Xbox, PlayStation en keyboard) dat betekent dat geen van deze programma’s input geregistreerd wordt omdat het niet van deze geregistreerd programma’s kwam.

# Resultaat

Het resultaat is een chat systeem, die mensen kunnen joinen, die reageert op bepaalde woorden zoals up, /down, /left en /right.

# 

# Bronnenlijst

<https://kaldi-asr.org/doc/dependencies.html>

<https://www.npmjs.com/package/node-key-sender>

<https://cmusphinx.github.io/wiki/tutorialsphinx4/>

<https://www.w3schools.com/js/js_regexp.asp>

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular\_Expressions

<https://stackoverflow.com>

<http://robotjs.io/docs/syntax#keytapkey-modifier>

https://github.com/octalmage/robotjs/issues/115