Snowboy to działający w czasie rzeczywistym, zawsze nasłuchujący oraz działający w trybie offline wysoce konfigurowalny silnik wykrywania słów kluczowych , który działa na systemach Linux, OS X oraz Raspberry Pi.

Zalety systemu Snowboy:

* wysoce konfigurowalny definiować własnych słów kluczowych
* program nie korzysta z Internetu i nigdzie nie przesyła głosu
* lekki (działa na Raspberry Pi i zużywa mniej niż 10% procesora)

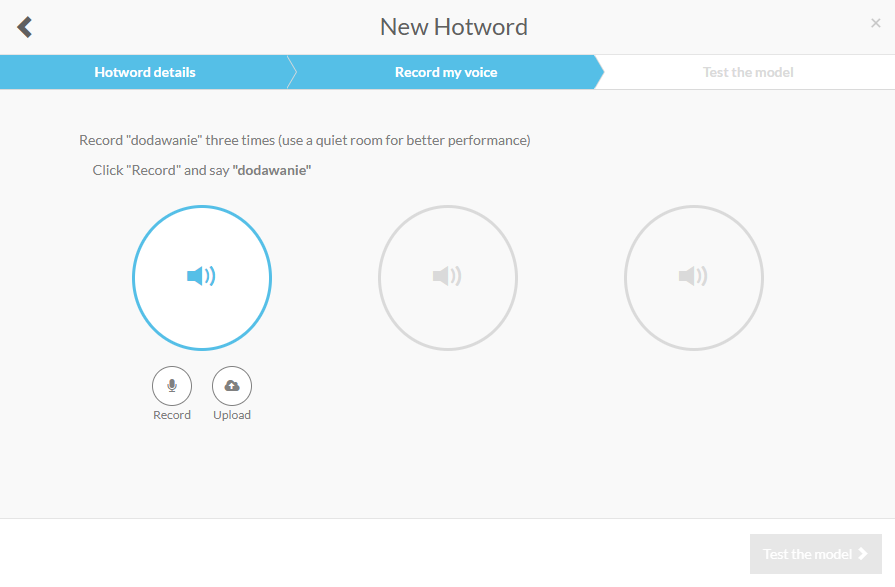
Snowboy jest to szybka alternatywa dla programu służącego do nasłuchiwania oraz wyłapywania i przetwarzania słów. Z punktu programistycznego ma to oczywiście spore minusy gdyż może się okazać, że nie mamy zbyt wielkiego pola manewru jeśli chodzi o algorytmy i funkcje działające nad wychwytywaniem słów kluczowych. Nie mamy również bezpośredniego wpływu na ingerencję w odbierany sygnał. Dzięki temu jednak, system ten jest bardzo szybki i prosty w implementacji.

Jednakże, projekt Snowboy był powierzchownie przez nas zbadany. Możliwe że istnieją możliwości na ustawienie poszczególnych parametrów dotyczących nasłuchiwania, sygnału, szumu, ciszy itp. Na pewno jest to rzecz warta uwagi oraz poświęcenia chwili czasu na zapoznanie się z tym tematem.

Przygotowywanie próbek dźwięku

Przygotowanie próbek odbywanie się poprzez nagranie głosu lub dźwięku poprzez stronę internetową która po zebraniu trzech próbek generuje model w postaci pliku z rozszerzeniem .pmdl

Gotowy model możemy pobrać z witryny oraz bezpośrednio podczepić pod nasz projekt importując go w kodzie do funkcji nasłuchującej.



Możemu również skorzystać z gotowych próbek głosu innych użytkowników dostępnych na portalu.

Link do strony: <https://snowboy.kitt.ai/dashboard>

Projekt snowboy dostarcza również kilka prostych przykładowych projektów demo, dzieki którym możemy zrozumieć jak to działa oraz pracować na projektach rozbudowywując demo.

Mamy także wiele projektów do wglądu i tutoriali odnośnie połączenia systemu sterowania głosem z systemami wbudowanymi. Dużą role w tym zakresie odgrywa również to że system snowboy możemy zainstalować na komputerach typu Rasberry Pi.

Wracając do samego kodowania, oto jeden z podstawowych kodów udostępnionych przez developera:

|  |
| --- |
|  |
|  | import snowboydecoder  import sys |
|  | import signal |
|  |
|  |
|  |
|  | interrupted = False |
|  | def signal\_handler(signal, frame): |
|  | global interrupted |
|  | interrupted = True |
|  | def interrupt\_callback(): |
|  | global interrupted |
|  | return interrupted |
|  |  |
|  | if len(sys.argv) != 3: |
|  | print("Error: need to specify 2 model names") |
|  | print("Usage: python demo.py 1st.model 2nd.model") |
|  | sys.exit(-1) |
|  |  |
|  | models = sys.argv[1:] |
|  |  |
|  |  |
|  | signal.signal(signal.SIGINT, signal\_handler) |
|  |  |
|  | sensitivity = [0.5]\*len(models) |
|  | detector = snowboydecoder.HotwordDetector(models, sensitivity=sensitivity) |
|  | callbacks = [lambda: snowboydecoder.play\_audio\_file(snowboydecoder.DETECT\_DING), |
|  | lambda: snowboydecoder.play\_audio\_file(snowboydecoder.DETECT\_DONG)] |
|  | print('Listening... Press Ctrl+C to exit') |
|  |  |
|  | detector.start(detected\_callback=callbacks, |
|  | interrupt\_check=interrupt\_callback, |
|  | sleep\_time=0.03) |
|  |  |
|  | detector.terminate() |

Wykrywanie 10 liczb (od 0 do 10):

import snowboydecoder

import sys

import signal

import pyttsx3

interrupted = False

def signal\_handler(signal, frame):

global interrupted

interrupted = True

def interrupt\_callback():

global interrupted

return interrupted

models = ["baza/jeden.pmdl","baza/dwa.pmdl", "baza/trzy.pmdl", "baza/cztery.pmdl", "baza/piec.pmdl", "baza/szesc.pmdl", "baza/siedem.pmdl", "baza/osiem.pmdl", "baza/dziewiec.pmdl", "baza/dziesiec.pmdl"]

signal.signal(signal.SIGINT, signal\_handler)

sensitivity = [0.5]\*len(models)

detector = snowboydecoder.HotwordDetector(models, sensitivity=sensitivity)

callbacks = [lambda: liczba(1), lambda: liczba(2), lambda: liczba(3), lambda: liczba(4), lambda: liczba(5), lambda: liczba(6), lambda: liczba(7), lambda: liczba(8), lambda: liczba(9), lambda: liczba(10)]

detector.start(detected\_callback=callbacks, interrupt\_check=interrupt\_callback, sleep\_time=0.03)

detector.terminate()

Powyżej przedstawiono najszybszą i najprostszą metodę do wykrywania wypowiadanych liczb bez ingerencji w pobierany/nasłuchiwany sygnał oraz bez ustawiania parametrów do wykrywania dzięków.

Linki do projektu snowboy:

<https://snowboy.kitt.ai/>

<https://github.com/Kitt-AI/snowboy>