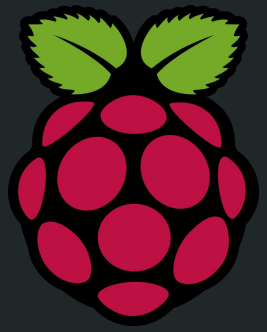
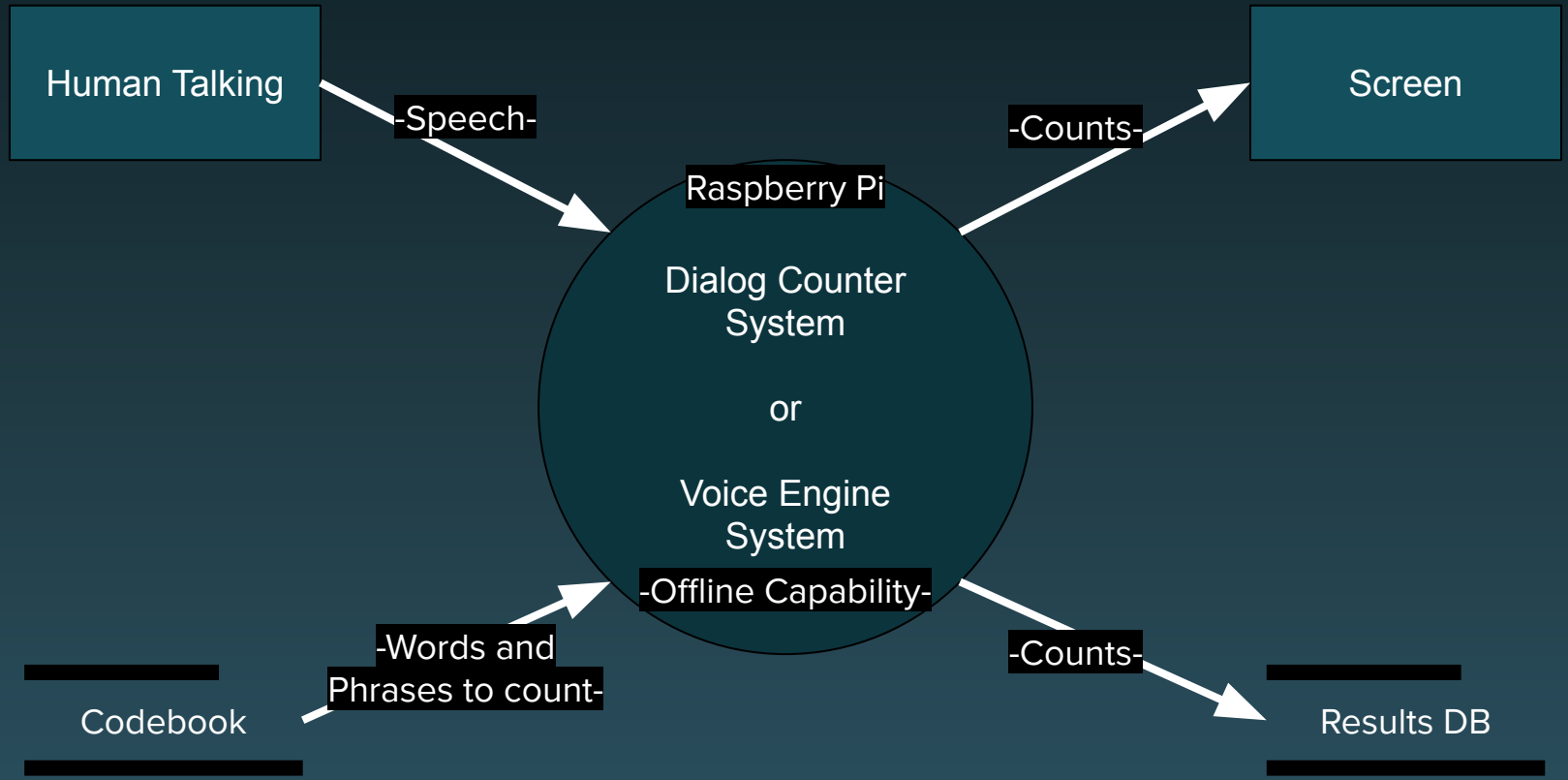
Raspberry Pi with Listener

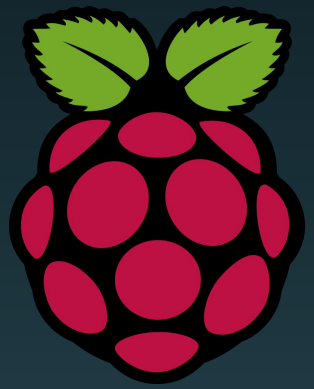


Jalapeno Hotties

Introduction Continue...



Introduction



Hardware Team

- Enrique Casillas
- Joel Ascher

- Duc Ngo

- Bekzod Tolipov
- Lawrence Catahay

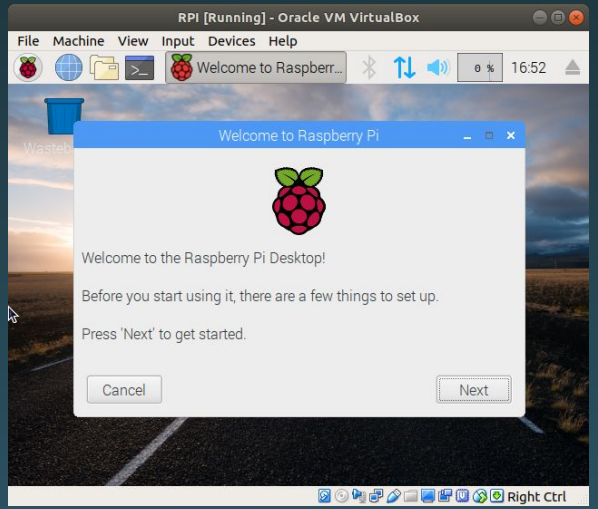
{software_team();}

What is Raspberry Pi?

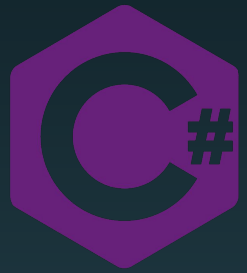
- Raspberry Pi 3 Model B
  - CPU: 1.4GHz 64-bit quad-core ARM Cortex-A53 CPU
  - RAM: 1GB LPDDR2 SDRAM
  - WIFI: Dual-band 802.11ac wireless LAN (2.4GHz and 5GHz) and Bluetooth 4.2
- Audio DAC HAT Sound Card (Audio + Speaker + MIC)
- Miscellaneous: Protective Case, 32GB Mini SD (storage)
- There are dozens of operating systems available on Raspberry Pi, and there is not a perfect distribution. Each distribution has its strengths and weaknesses and is more suited to a specific use.
  - Raspbian
  - Ubuntu Mate
  - Retropie
  - OSMC
  - Kali Linux
  - ...

Working with Pi

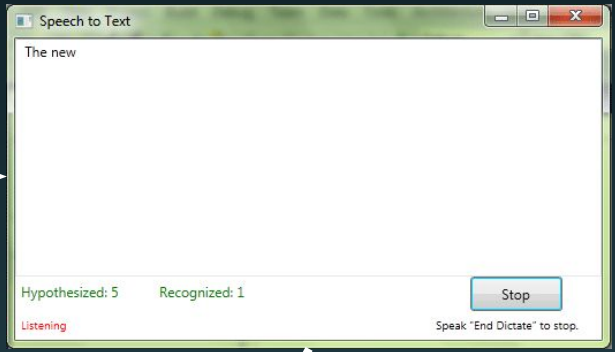
- Virtual Machine
  - Emulator
  - Oracle VM VirtualBox
    - Powerful Cross-platform (Windows, Linux, Mac OS X, and Solaris) Virtualization Software for x86-based systems.
  - Very convenient for testing purposes as there's no consequences for wrecking the virtual OS.

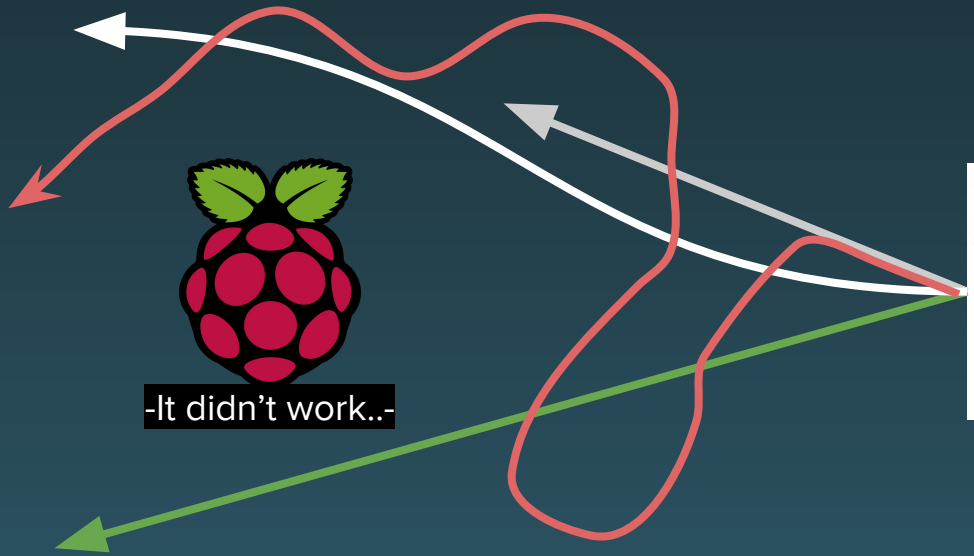


The Struggle...



- ✓ Create a `SpeechConfig` object from your subscription key and region.
- ✓ Create a `SpeechRecognizer` object using the `SpeechConfig` object from above.
- ✓ Using the `SpeechRecognizer` object, start the recognition process for a single utterance.
- ✓ Inspect the `SpeechRecognitionResult` returned.





Voice Engine - CMUSphinx

- CMUSphinx - Developed at Carnegie Mellon University
  - State of art speech recognition algorithms for efficient speech recognition. CMUSphinx tools are designed specifically for low-resource platforms.
  - Focus on practical application development and not on research
  - Support for several languages like US English, UK English, French, Mandarin, German, Dutch, Russian and ability to build a models for others
  - Commercial support.
  - Wide range of tools for many speech-recognition related purposes (keyword spotting, alignment, pronunciation evaluation).

Pocketsphinx

- Pocketsphinx is a part of the CMU Sphinx Open Source Toolkit For Speech Recognition.
  - Supports Windows, Mac OS X, and Linux platforms.
  - Provides a Python Interface to CMU Sphinxbase and Pocketsphinx Libraries created with SWIG and Setuptools.
  - Application Utilizes an Iterator Class for Continuous Recognition or Keyword Search from a Microphone.

Pocketsphinx & LiveSpeech

- Pocketsphinx is an Offline Speech Recognition Engine & must be Installed Locally.
- To Convert Speech to Text, LiveSpeech is Imported from the Pocketsphinx Python Library.

```
from pocketsphinx import LiveSpeech
for phrase in LiveSpeech(): print(phrase)
```

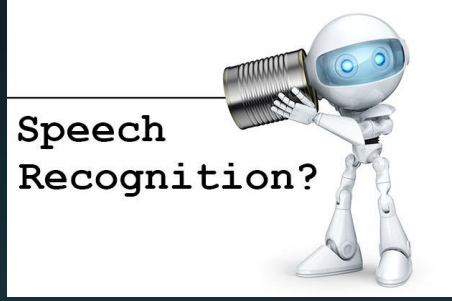
python™ | Raspberry Pi

There are a couple of caveats...

- Accuracy is poor... why?
  - The mismatch of the acoustic model.
  - The mismatch of the language model.
  - The mismatch in the dictionary and the pronunciation of the words.
- It takes time to train the model to improve accuracy.
  - Database of test samples
  - Speech pronouncement -> WAV files
  - Calculate the Word Error Rate (WER) using the word_align.pl tool from Sphinxtrain.
  - Larger size = more accuracy (bigger is better)
- Due to Limited Time & Resources
  - Unable to fully train the model.
  - Model cannot be taught every English word.
  - Possesses the potential to be trained to capture any desired input.

Pocketsphinx Execution

- Python's Pocketspinx LiveSpeech Sample:




```
import sys
import pocketsphinx

if __name__ == "__main__":

    hmdir = "/usr/share/pocketsphinx/model/hmm/wsj1"
    lmdir = "/usr/share/pocketsphinx/model/lm/wsj/wlist5o.lm.DMP"
    dictd = "/usr/share/pocketsphinx/model/lm/wsj/wlist5o.dic"
    wavfile = sys.argv[1]

    speechRec = pocketsphinx.Decoder(hmm = hmdir, lm = lmdir, dict = dictd)
    wavFile = file(wavfile,'rb')
    speechRec.decode_raw(wavFile)
    result = speechRec.get_hyp()

    print result
```

Codebook Generator Intro

1. Error Messages
2. Allow user to change mistakes
3. Spell Checking
4. Formatted output file

Codebook Generator: Words

```
Please input up to 6 words separated by comma',' (word_1, word_2...):
sine       ,                cosine,        generic     ,    forhet
Is the given word spelled correctly?
Given input: (forhet)
Possible spelling: (forget)
Do you wish to change it:
1) Yes
2) No
1
Given input: (forhet)
Possible spelling: (forget)
please input corrected word:
forget
['sine', 'cosine', 'generic', 'forget']
```

Codebook Generator: Phrases

```
Please input up to 4 phrases separated by comma',' min:2 words and max:3 words (phrase_1, phrase_2...):
sine
1
ERROR in Phrases: Given data contains only a word, you need 2 or 3 words to create phrase
REMINDER: Please follow the following example:
Please input up to 4 phrases separated by comma',' min:2 words and max:3 words (phrase_1, phrase_2...):
```

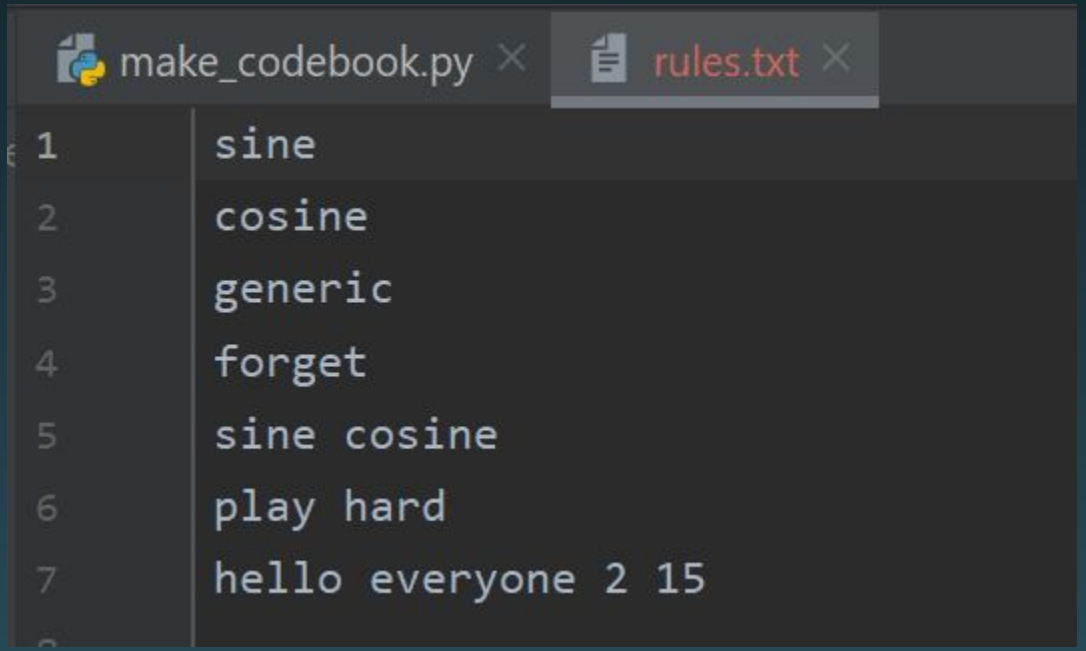
```
Please input up to 4 phrases separated by comma',' min:2 words and max:3 words (phrase_1, phrase_2...):
sine coasine, getttt hord
2
Is the given word spelled correctly?
Given input: (coasine)
Possible spelling: (cosine)
Do you wish to change it:
1) Yes
2) No
2

--CAUTION: Current word may not return results as expected
```

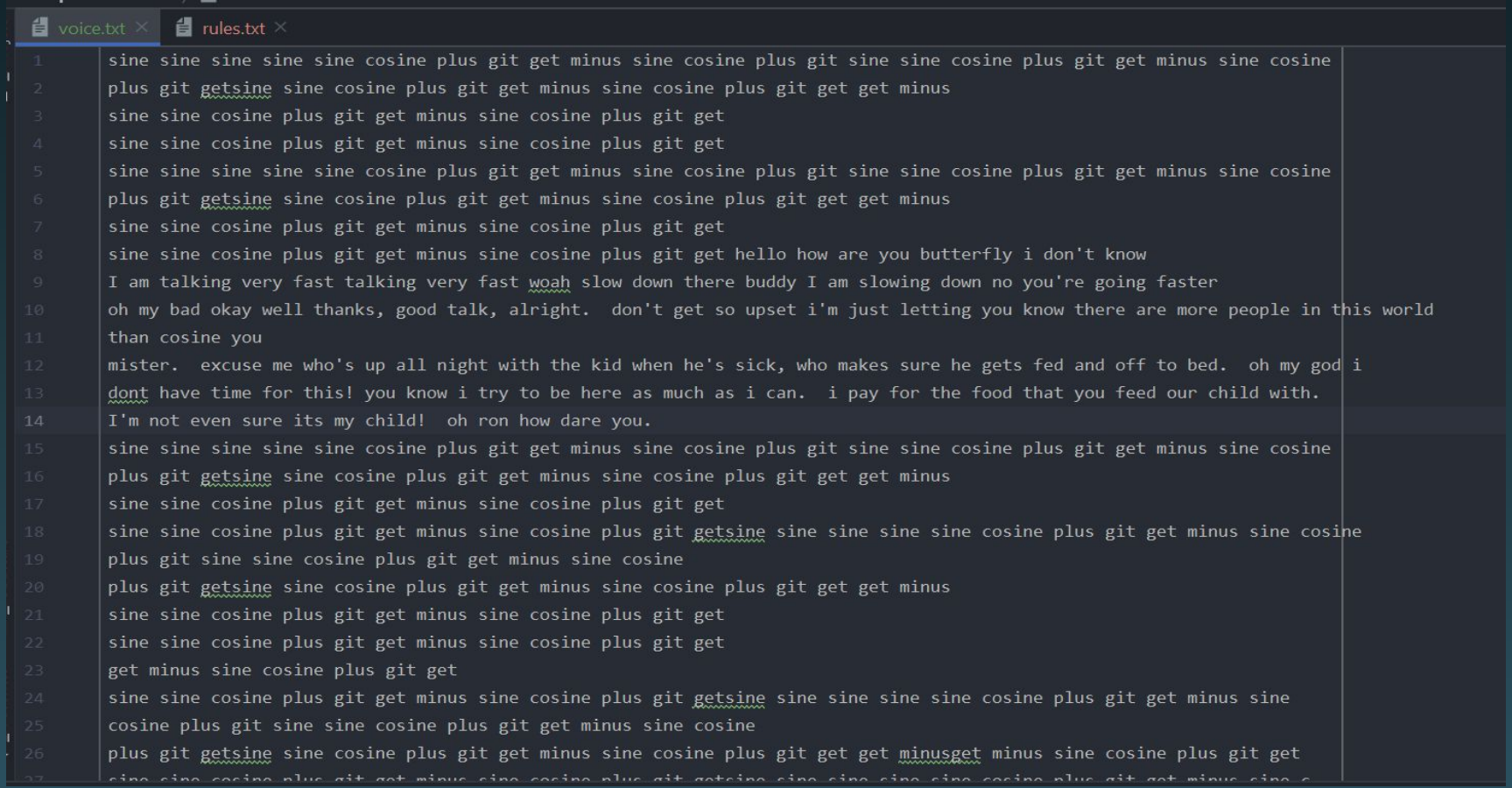
Codebook Generator: Patterns

```
Please input up to 2 pattern followed by lower-bound=l, upper-bound=u separated by comma',' (word_1 word_2 l u, word_1 word_2 l u...
hello everyone
ERROR in Pattern: Less than 4 elements given, missing data
REMINDER: Please follow the following example:
Please input up to 2 pattern followed by lower-bound=l, upper-bound=u separated by comma',' (word_1 word_2 l u, word_1 word_2 l u...
hello everyoione 2 15
Is the given word spelled correctly?
Given input: (everyoione)
Possible spelling: (everyone)
Do you wish to change it:
1) Yes
2) No
1
Given input: (everyoione)
Possible spelling: (everyone)
please input corrected word:
everyone
[['hello', 'everyone', '2', '15']]
```

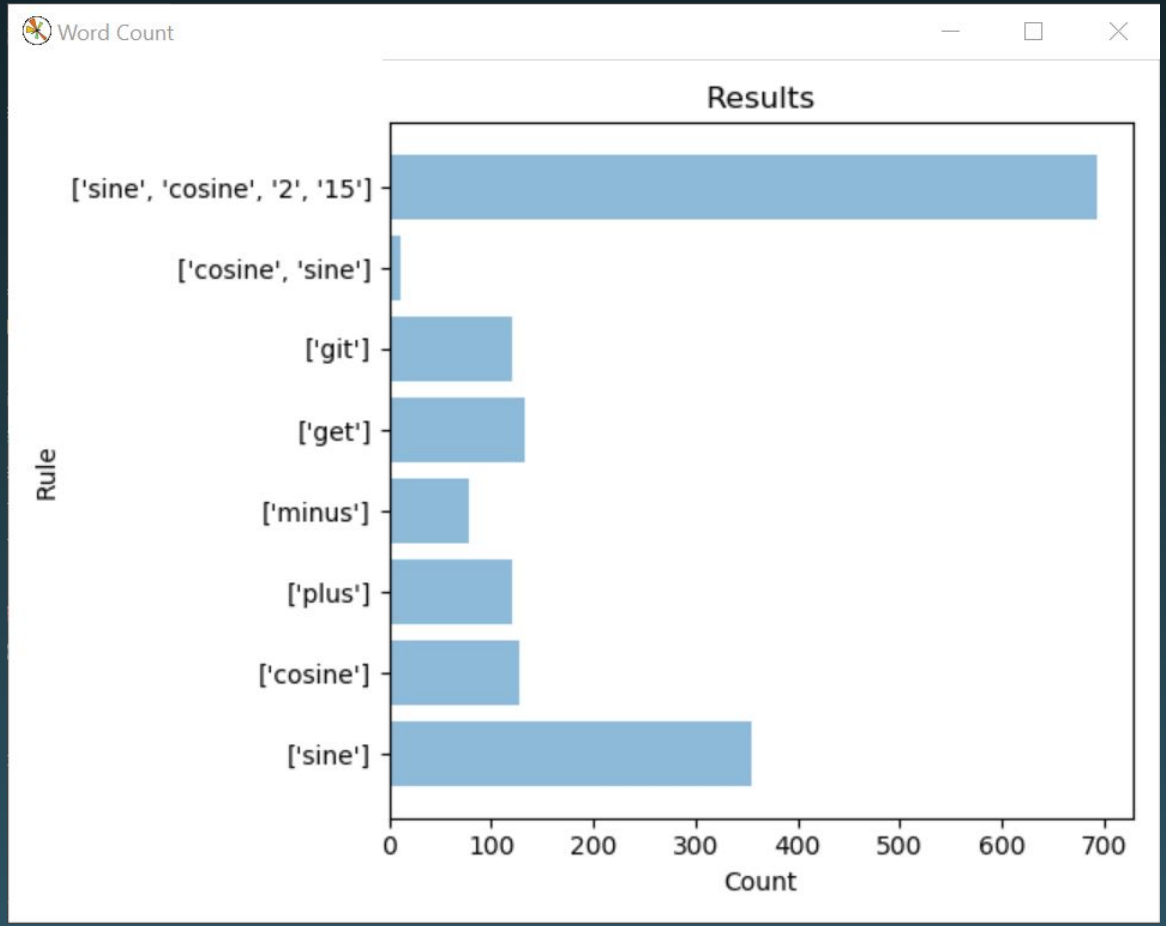
Codebook Generator: Output file



Word Counter with Graph



Word Counter with Graph (2)



Q & A