AI FOR SOCIAL IMPACT

WADHWANI AI

AI Assisted Reading Tool

Shashank Kirtania

Project Aim

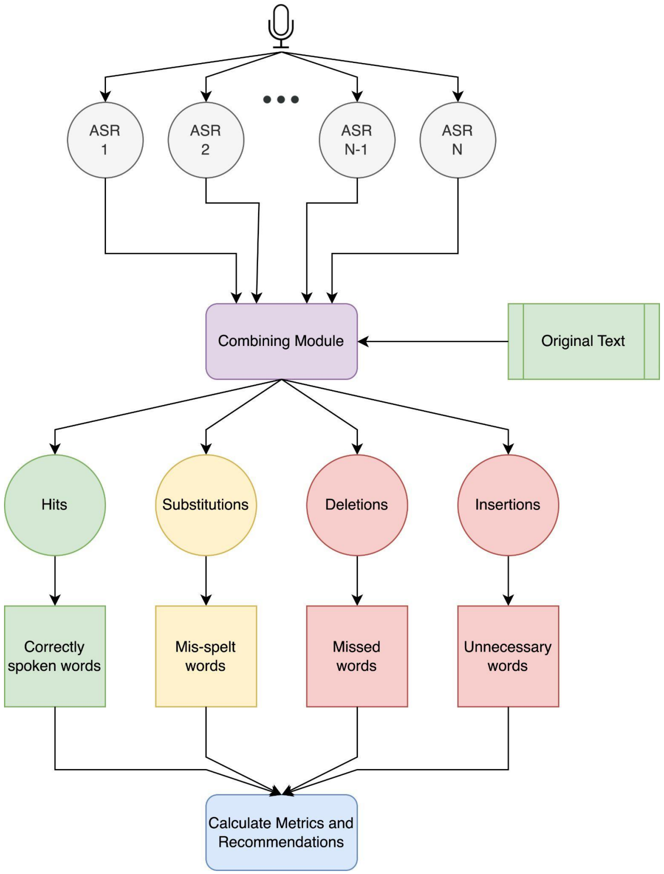
- Understanding complexity of reading assessments for elementary school students.
- Understanding the pauses between the words and leveraging that for finding fluency.
- Using standard ASR models to identify various metrics to analyse reading capability.
- Provide recommendations/exercise to improve fluency.
- Perform speaker verification to avoid cheating/malpractices.

Challenges:

- Since this is an assessment tool, it should be accent and dialect agnostic.
- Quality audio data is scarce for most indian languages.
- ASR error can be falsely counted as error by the student.
- System latency should not be more than a few seconds. (optional)

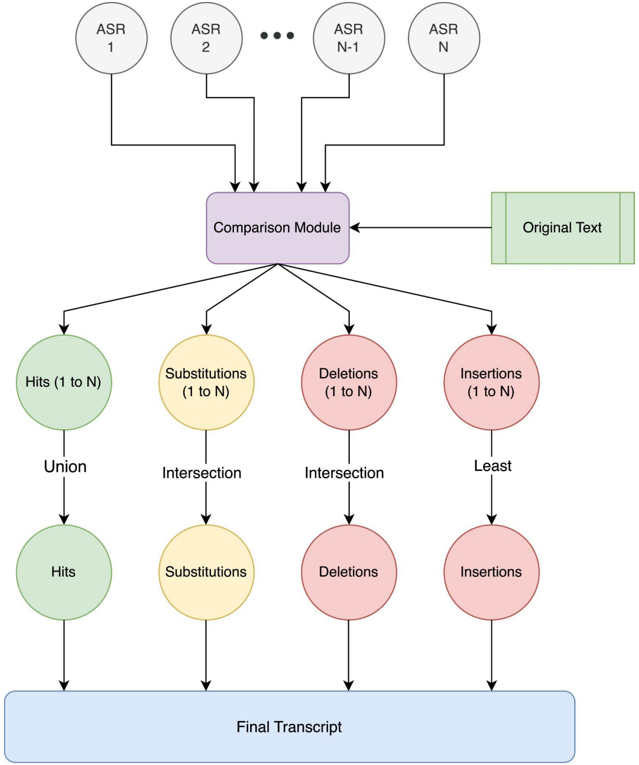
Overall Workflow (Current)

- Record speaker audio, pass it through multiple ASR models
- Combining Module combines multiple ASR outputs by taking into account the Original Text.
- Finally we get Hits, Substitutions, Deletions and Insertions.
- Now we calculate Metrics and give recommendations based on the above 4 outputs.



Combining Module

- This module's task is to combine the outputs of multiple ASR models and provide a final transcript.
- Through this module, we want to minimize the ASR errors in favour of the speaker.
- Each ASR output is first compared with the Original Text and broken down into 4 parts: Hits, Substitutions, Deletions and Insertions.
- Final Hits become the union of the individual Hits.
  Consequently,
  Final substitution becomes the intersection of individual substitutions.
  Final Deletion becomes the deletion of individual deletion.
- For insertion, we consider it to be the least insertion of all outputs.
- Combine the final Hits, Substitution, Deletions and Insertions to create the final transcript.



Combining Module

Example

Input = “i am a good boy. i like to swim in the dark”

ASR outputs =
[
“umm i am ah good boy. i like to swim in dark",
"i am ah good boy. ilike to swim in dark",
"umm i am ah good boy. x i like to swim in dark"
]

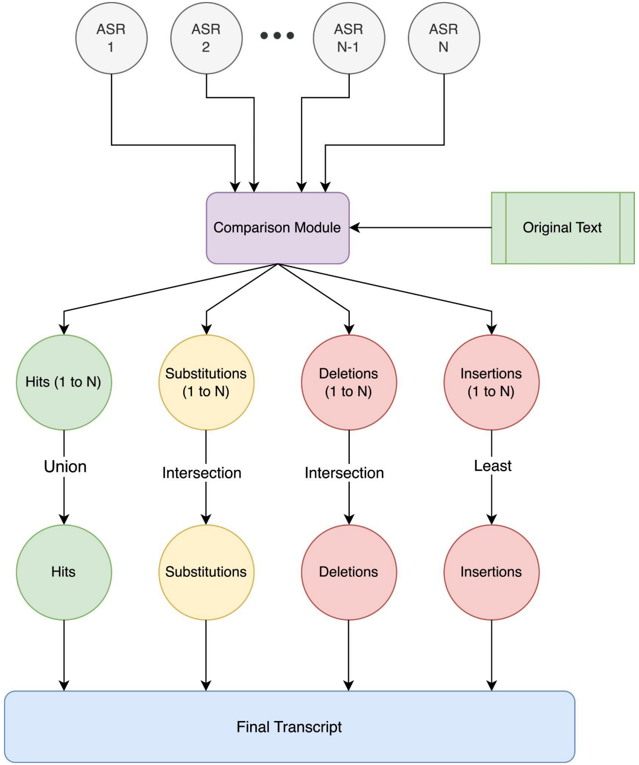
Hits: [“i”, “am”, “good”, “boy”, “i”, “like”, “to”, “swim”,”in”, “dark”]

Substitution: [“a”: “ah”]

Deletions: [“the”]

Insertions: []

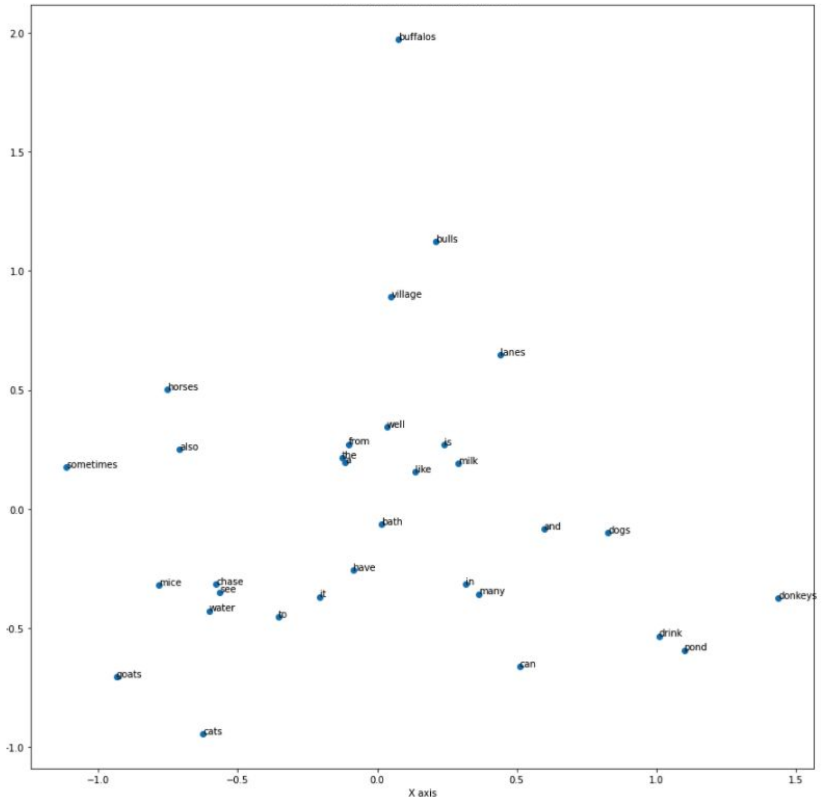
Final Transcript: “i am ah good boy. i like to swim in dark”



WADHWANI AI

Textual analysis

- The number of phonemes in a word can be an indicator of its complexity.
- Longer words tend to have more phonemes, which can make them more difficult to pronounce correctly.
- The number and type of syllables in a word can also impact its complexity.
- Words with complex phoneme combinations or irregularities in pronunciation can be more difficult to master.
- Analyzing the stress patterns in multi-syllabic words can also provide insight into the complexity of pronunciation.



Plot of similarity of words on basis of Phonemes

Unique user identification

- Mel frequency spectrograms are used to extract features from an audio signal for identifying unique speaker characteristics.
- Feature vectors are created by analyzing the spectrogram to extract pitch, formants, and acoustic characteristics.
- Machine learning algorithms can be used to group or differentiate between speakers based on their feature vectors.
- Neural networks and deep learning improve accuracy by allowing for more complex feature extraction and modeling.

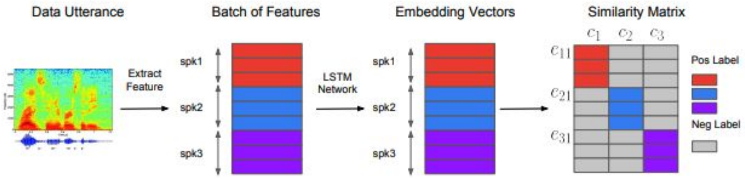
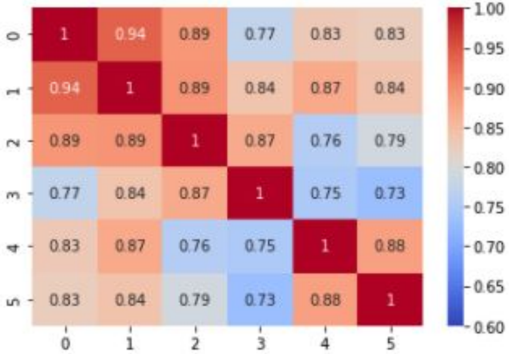


Fig. 1. System overview. Different colors indicate utterances/embeddings from different speakers.

Source: Generalized End-to-End Loss for Speaker Verification (Li Wan et. al)
https://doi.org/10.48550/arXiv.1710.10467



Cosine similarity between samples of 3 different people reading 2 different paragraphs

Thank you!

WADHWANI AI | www.wadhwaniai.org