A Deep Dive into Phoneme-Level Pronunciation Assessment

7 min read · Feb 29, 2024

Rudder Analytics Following

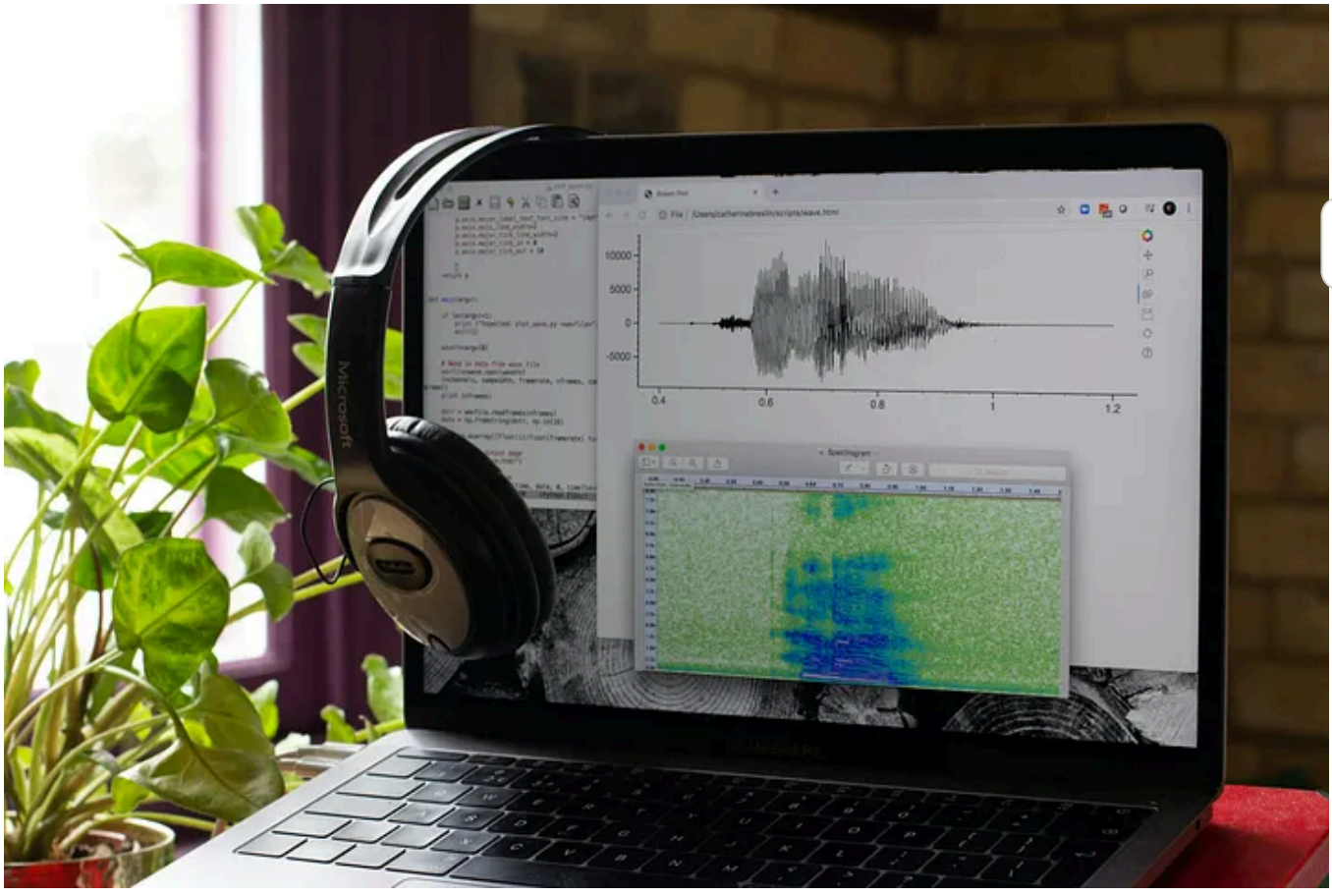


Photo by Catherine Breslin on Unsplash

In the rapidly evolving digital education domain, our team at Rudder Analytics embarked on a pioneering project. We aimed to enhance language learning through cutting-edge AI and machine learning technologies. Partnering with a premier language learning platform, we sought to address a significant challenge in the field: providing detailed and

actionable feedback on pronunciation at the phoneme level, a critical aspect of mastering any language. This case study delves into the sophisticated technical landscape we navigated to develop an advanced phoneme-level pronunciation assessment tool, showcasing our data analytics, engineering, and machine learning expertise.

Navigating the Challenge: Beyond Conventional Solutions

The initial challenge was the limitations of out-of-the-box pronunciation scoring APIs provided by major cloud services like GCP, Azure, and AWS. These services, while robust, fell short of delivering the granular level of detail required for effective pronunciation assessment. To overcome this problem, the decision was made to construct a bespoke model that could meet the specific needs of the platform.

Our objective was clear: to architect a solution that transcends these limitations, enabling a more personalized and impactful learning experience.

Holistic Approach: Integrating Advanced Algorithms with Linguistic Insights

Our strategy was anchored in a holistic approach, merging advanced machine learning techniques with deep linguistic insights to achieve higher accuracy in pronunciation assessment.

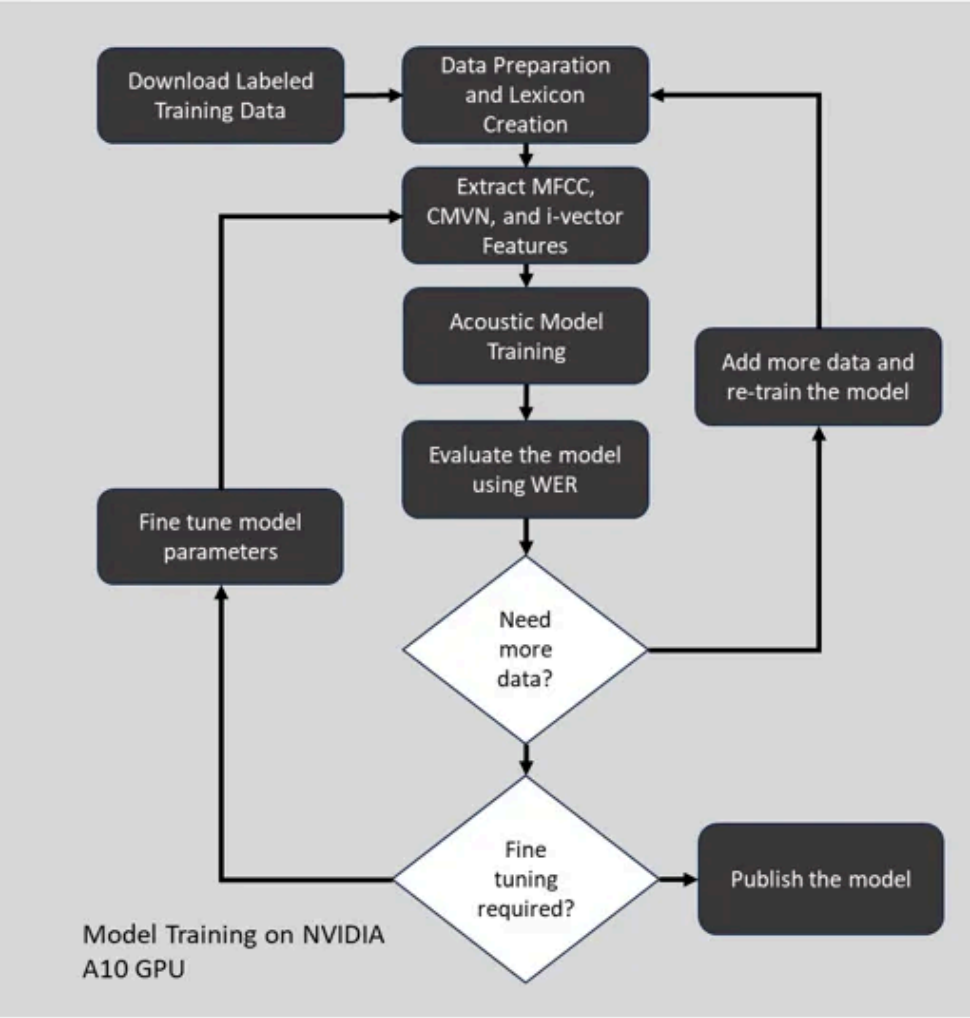
If you are interested in the codebase, check out our GitHub repository

Goodness of Pronunciation (GOP)

A cornerstone of our approach was the implementation of the Goodness of Pronunciation (GOP) metric. GOP, a posterior probability variant, is a quantitative measure of pronunciation accuracy at the phoneme level. It's an important tool for identifying mispronunciations, enabling targeted feedback for language learners. GOP is used to evaluate the system's performance in recognizing and scoring the pronunciation of a given utterance.

The Strategic Employment of Kaldi ASR Toolkit

Kaldi, an open-source ASR framework, stands at the core of our solution. Renowned for its flexibility and efficiency in handling speech recognition tasks, Kaldi offers a range of recipes for tailoring acoustic models to specific needs. Our choice to utilize Kaldi was driven by its comprehensive feature set and the ability to be customized for phoneme level detection, a critical requirement for our project.



Acoustic model training on NVIDIA A10 GPU, Image by Author.

Data Collection and Preparation

The foundation of our solution was a robust data infrastructure, engineered to handle vast datasets. We utilized the Librispeech dataset, a comprehensive collection of English language audio files. It contains over 1000 hours of speech, recorded by 2,484 speakers, and has been designed to be representative of the different accents and dialects of spoken English. These recordings were made using high-quality microphones and a sound-treated recording environment to ensure high-quality audio.

This dataset contains labeled audio data. We also collected the pronunciation lexicon which included words and their corresponding sequences of phonemes, essential for training our model to recognize and evaluate the smallest sound units in speech.

Major Components

In Kaldi, when computing Goodness of Pronunciation (GOP), the acoustic model, pronunciation lexicon, and language model play distinct roles in evaluating how well a speaker's utterance matches the expected pronunciation of words in a given language.

There are 3 main parts of the GOP *Speechocean* recipe from Kaldi.

Acoustic Model: The acoustic model is trained to recognize the various sounds (phonemes) that make up speech. It maps the raw audio features to phonetic units. In the context of GOP, the acoustic model evaluates how closely the sounds in the speaker's utterance match the expected phonemes of the correct pronunciation. The model's confidence in phoneme predictions plays a key role in calculating the GOP score.

Pronunciation Lexicon: The pronunciation lexicon provides the expected phonetic transcriptions of words. It is a reference for how words should be pronounced in terms of phonemes. When calculating GOP, the system uses the lexicon to determine the target pronunciation of words or phrases being evaluated. The comparison between this target pronunciation and the spoken pronunciation (as interpreted by the acoustic model) is fundamental to assessing pronunciation quality.

prepare_lang.sh script is used to prepare lexicon and language-specific data files. It includes creating a *lexicon.txt* file that contains word-to-phone mapping (eg. Hello -> HH EH L OW)

Language Model: While the language model is primarily used to predict the likelihood of word sequences in speech recognition, its role in GOP can be indirect but important. It can help disambiguate phonetically similar words or provide context that makes certain pronunciations more likely than others, thus influencing the assessment of pronunciation quality. The language model can also ensure that the phoneme sequences being evaluated are within plausible linguistic constructs, which can affect the interpretation of pronunciation accuracy.

Training Process

1. Preparation of Resources: We gathered a phonetically rich and transcribed speech corpus. Then, set up a pronunciation dictionary (lexicon), language models, and necessary configuration files.

2. Feature Extraction: Extracted acoustic features from the speech corpus. Commonly used features include MFCCs (Mel-Frequency Cepstral Coefficients) or FBANK (Filterbank Energies).

3. Training Acoustic Models: We then used the extracted features and transcriptions to train acoustic models. The models learn the relationship between the acoustic features and the phonetic units or words.

Training starts with building a simple model:

Monophone Models: These models recognize phonemes without considering context (neighboring phonemes). They are simpler and less accurate but provide a good starting point.

Kaldi's *train_mono.sh* script is used to perform the monophone training.

Triphone Models: These models consider the context of phonemes (typically the immediate previous and next phonemes). They are more complex and capture more details about speech patterns.

Kaldi's *train_deltas.sh* script is used to perform the triphone training.

Refinement: Once triphone models are trained, Kaldi's *train_SAT.sh* script is used to refine the model to handle different speakers. SAT stands for Speaker Adaptive Training.

4. Alignment: Performed forced alignment using the trained acoustic models to align the phonetic transcription with the acoustic features. This step is crucial for GOP as it determines how well the predicted phonemes match the phonemes spoken in the audio.

Kaldi provides *align_si.sh* script just for this purpose!

The script uses the transcriptions and a lexicon (which maps words to their phonetic representations) to compile training graphs. These graphs represent how words (and their phonetic components) can transition during speech according to the language model.

The script performs the alignment task using the training graphs, the existing acoustic model, and the normalized features. This involves determining the most likely sequence of states (which correspond to phonemes or groups of phonemes) that the acoustic model believes were spoken in each training utterance.

5. GOP Calculation: The Goodness of Pronunciation score is calculated based on the likelihoods produced by the acoustic model during alignment. GOP is a log-likelihood ratio for each phoneme, normalized by the phoneme duration. It indicates how well the phoneme matches the expected model of that phoneme.

GOP is calculated using the *compute-gop* script of Kaldi. The steps include:

Compute Posteriors: It first computes the posterior probabilities of different phoneme sequences given the acoustic model and the observed features.

Calculate Log-Likelihoods: The script computes the log-likelihoods for each phoneme occurring at each time frame.

Evaluate Pronunciation: GOP is calculated by comparing the log-likelihood of the most likely phoneme sequence (as per the hypothesis) to alternative phoneme sequences. To avoid bias toward longer phonemes, the score is normalized by the duration of the phoneme.

6. Pronunciation Profiling: The GOP scores can be used to profile the speaker's pronunciation. Low scores indicate areas where the speaker's pronunciation deviates from the expected model.

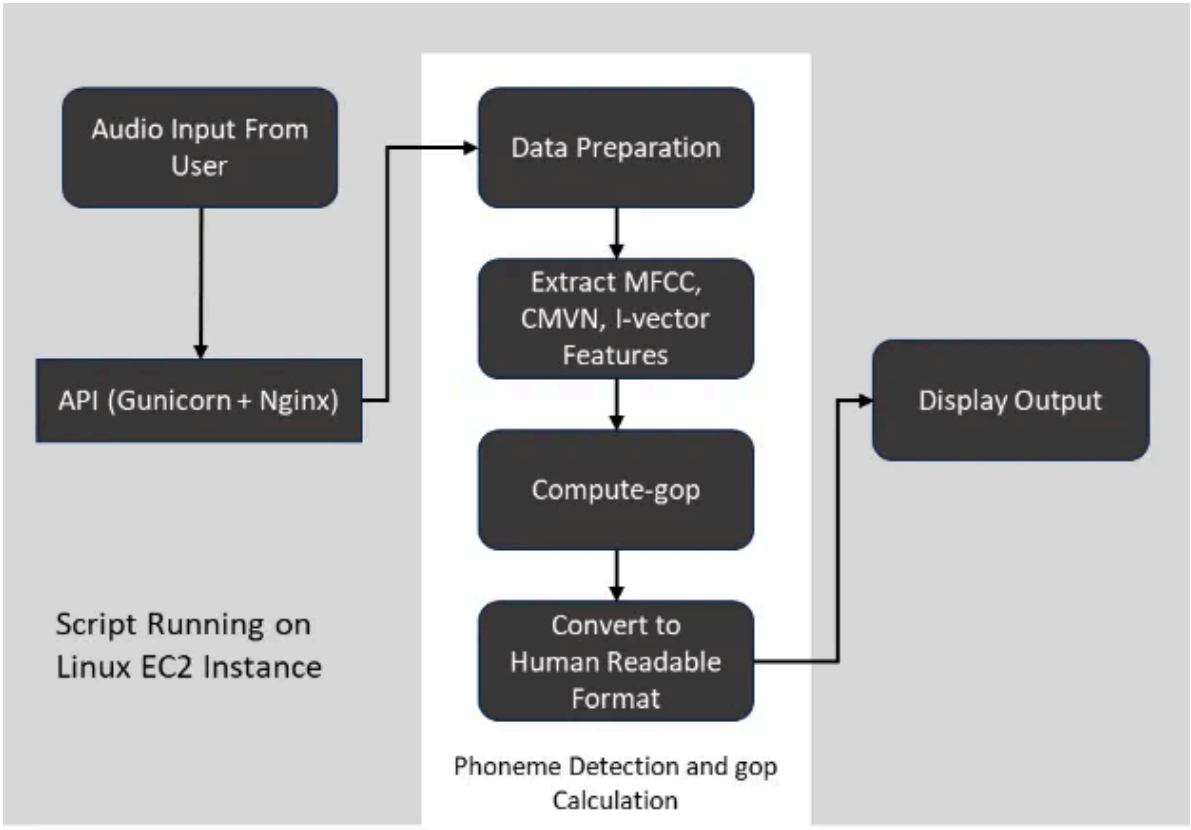
7. Model Refinement: Based on the GOP scores, we identified the need for additional training data in areas where the pronunciation model is weak. Additional training and refinement of models may occur iteratively.

8. Application of GOP: Once the system is well-calibrated, GOP scores can be applied in various ways, such as in language learning applications to provide feedback on pronunciation, in speech recognition systems to improve robustness, or in speaker assessment and training tools.

If you are interested in the codebase, check out our GitHub repository

Model Evaluation

A critical phase of our implementation process was rigorous system testing and evaluation. We assessed the model's performance using Word Error Rate (WER), a common metric in speech recognition that helped us understand how often the model incorrectly predicted phonemes. WER is a critical evaluation metric of Automatic Speech Recognition (ASR) systems, serving as a quantifiable measure of transcription accuracy. It is calculated by comparing the ASR system's output against a reference transcription, considering the number of substitutions, deletions, and insertions needed to match the system's output to the reference.



Inference script running on Linux EC2 instance for Goodness of Pronunciation calculation, Image by Author.

Measurable Impact: Enhancing User Experience and Engagement

The deployment of this phoneme-level pronunciation assessment tool has had a profound impact on the platform's user engagement metrics. We observed a 12% increase in user engagement, a testament to the enriched learning experience provided by our solution. Furthermore, the platform saw an 8% rise in user retention, indicating that users found the tool engaging and effective in improving their skills. Perhaps most telling was the 10% increase in user referrals and testimonials, a clear indicator of the tool's impact on users' language learning journeys and its contribution to positive word-of-mouth for the platform.

Conclusion

Our comprehensive approach to enhancing phoneme detection in language learning platforms has set a new standard in pronunciation training. We have crafted a system that improves pronunciation accuracy and enriches the language learning experience by utilizing advanced technological solutions like the Kaldi ASR toolkit. This project exemplifies our commitment to harnessing advanced technology in addressing educational challenges, contributing significantly to the advancement of language learning methodologies.

Elevate your projects with our expertise in cutting-edge technology and innovation. Whether it's advancing language learning tools or pioneering in new tech frontiers, our team is ready to collaborate and drive success. Join us in shaping the future — explore

our services, and let's create something remarkable together. Connect with us today and take the first step towards transforming your ideas into reality.

Drop by and say hello! Website LinkedIn Facebook Instagram X

Machine Learning | Data Science | Language Learning | Speech Recognition | Pronunciation Scoring

Following

Written by Rudder Analytics

67 followers · 3 following

End-to-end data analytics consulting firm, providing predictive and exploratory analytics solutions and AI & ML services.

Responses (1)

Rick Liao

What are your thoughts?

Jorgecardete
Mar 1, 2024

Great article!

1 reply Reply

More from Rudder Analytics

Rudder Analytics

AI Agent for SQL Queries and Visualization using Multi-agent Framework

Solution to Data Querying Challenges

Apr 11 24

Rudder Analytics

Voice-Based Security: Implementing a Robust Speaker Verification System

Implementing a robust voice authentication system. Overcoming challenges, achieving high accuracy, and ensuring seamless se

Jun 20, 2024 34

Rudder Analytics

AI Agent for Loan and Mortgage Approval: A Smarter Path to Compliance and Decision-Making

In today's highly competitive financial landscape, financial institutions must balance speed, accuracy, and compliance when reviewing loan...

May 23 8

Rudder Analytics

Enhancing Podcast Audio Clarity with Advanced Speech Separation Techniques

Advanced speech separation boosts podcast efficiency by 17%, enhances audio clarity, and increases listener engagement.

May 9, 2024 45

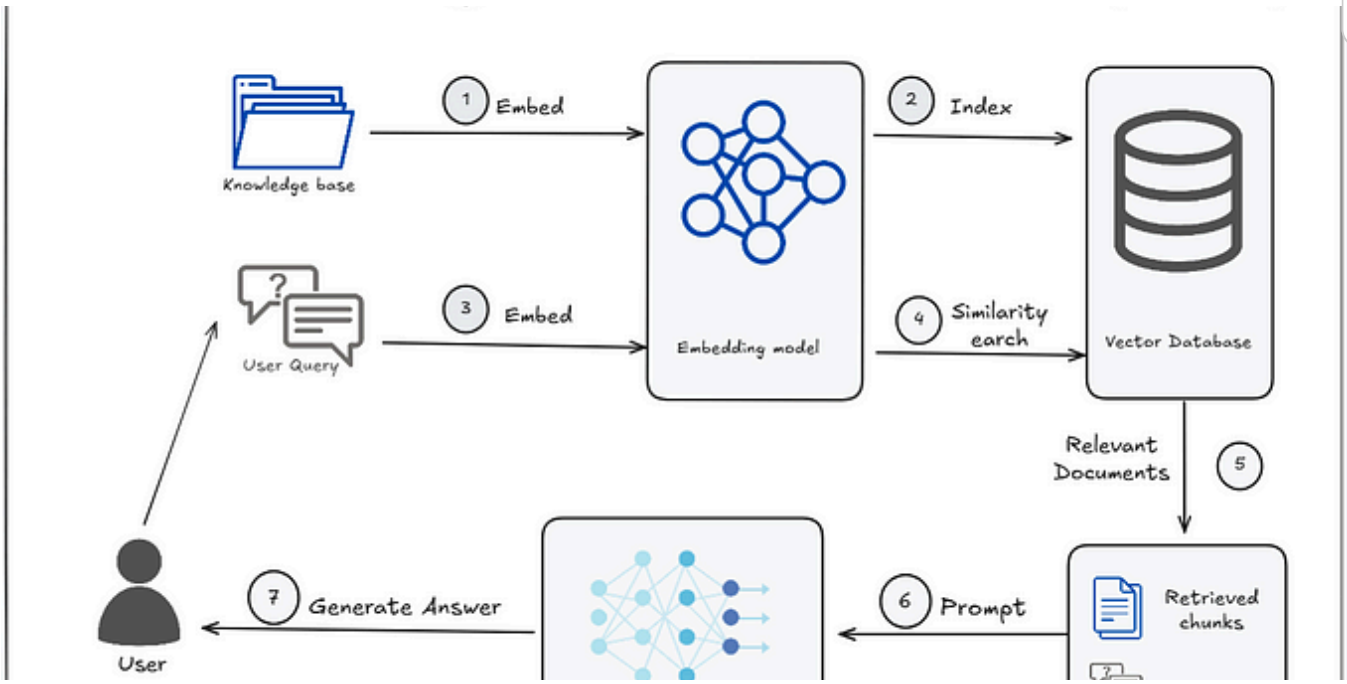
See all from Rudder Analytics

In AIGuys by Vishal Rajput

Leave Agentic AI Frameworks And Build Agents From Scratch

I'll be honest with you, I hate most agent-based AI workflows; they are simply unusable in the real world at scale. Despite the...

4d ago 632 23

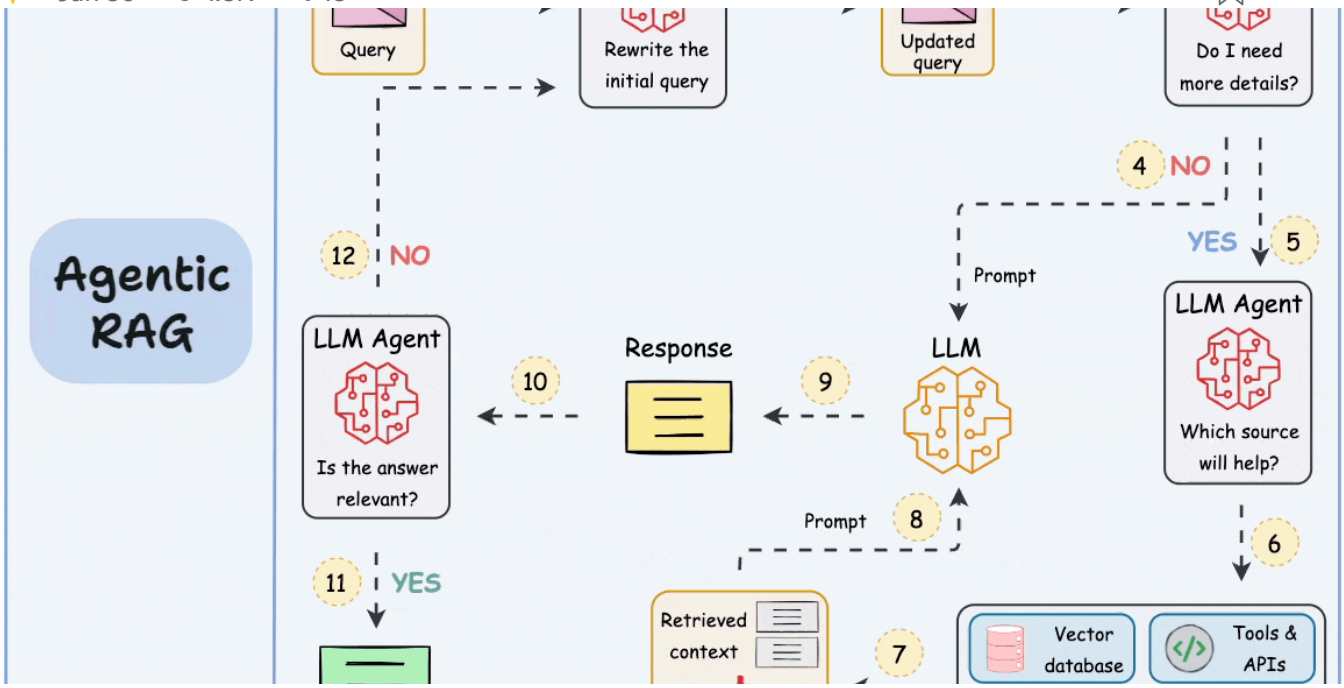


In AI Advances by Anjolaoluwa Ajayi

21 Chunking Strategies for RAG

And how to choose the right one for your next LLM application

Jun 30 1.3K 13

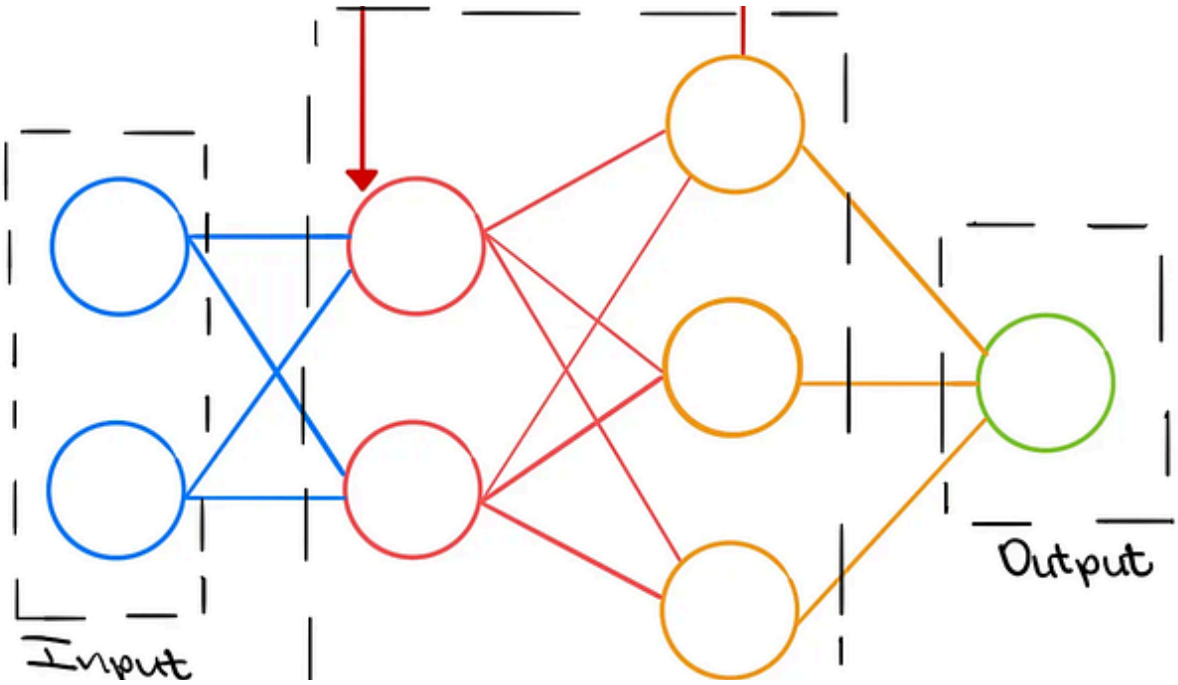


In Artificial Intelligence in Plain English by Piyush Agnihotri

Building Agentic RAG with LangGraph: Mastering Adaptive RAG for Production

Build intelligent RAG systems that know when to retrieve documents, search the web, or generate responses directly

Jul 21 1.6K 22

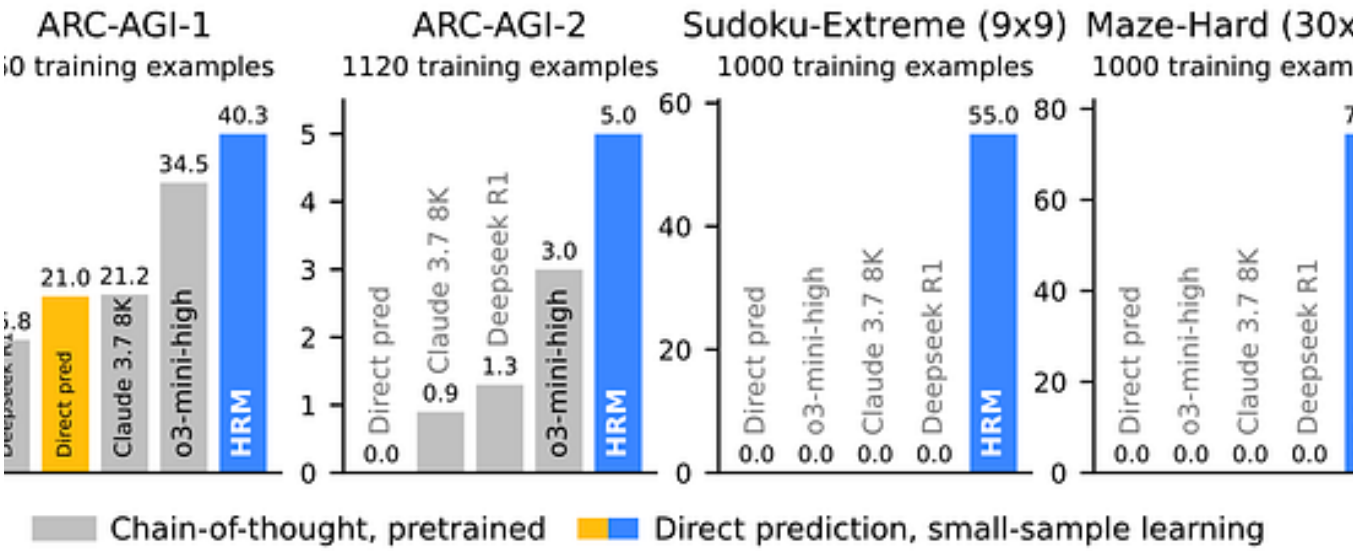


Dilip Kumar

Transformers Neural network architecture

The Transformer architecture is arguably the most significant development in machine learning in the last decade, revolutionizing Natural...

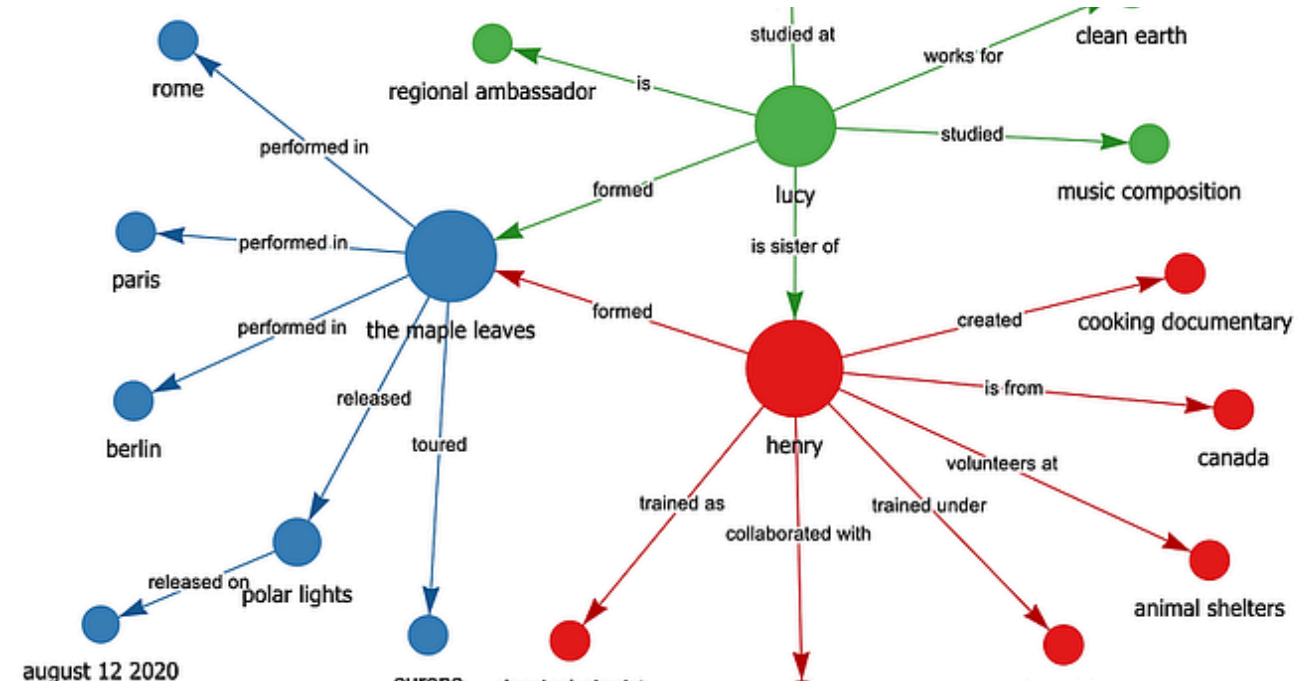
Jul 26 3



Allen Liang

Killing LMMs with a tiny 27M model: Hierarchical Reasoning Model Explained (HRM) Briefly Explained

6d ago 11



GANTEDA RAJABABU

Transforming Unstructured Text into Interactive Knowledge Graphs with Large Language Models

Introduction

Jul 24 172 3