ReadVideo

Final Project

CS 5319, Natural Language Processing

Instructor: Dr. Nigel Ward

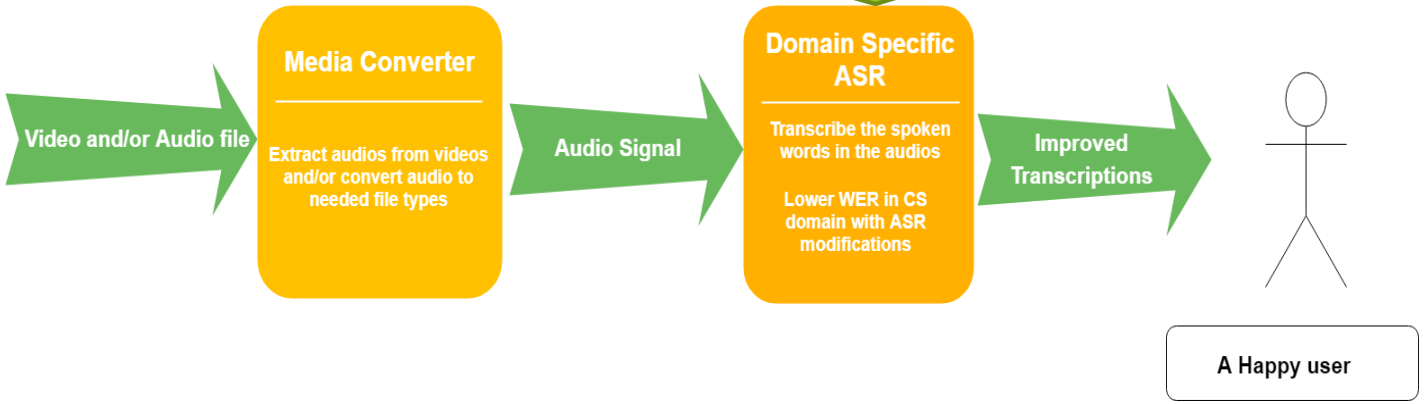
By:

Anindita Nath

Gerardo Cervantes

Motivation:

no such CS specific models known, difficulty in understanding accents, more help to non-native speakers



Project Overview

Media Converter

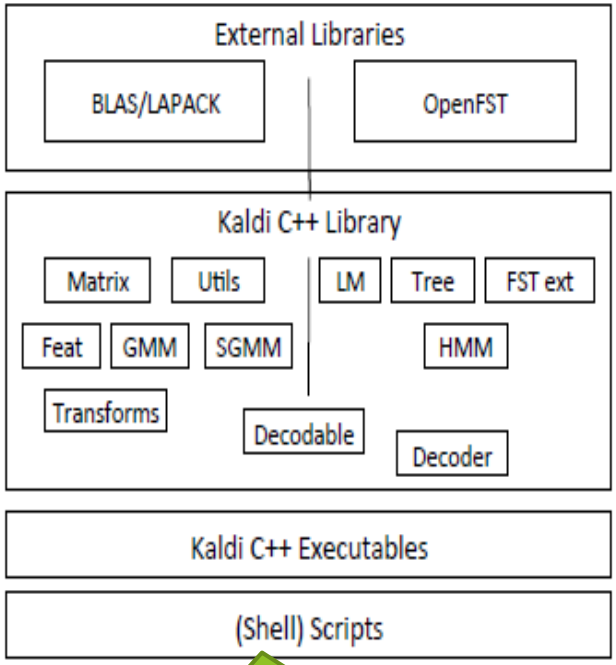
a bash script, uses Ffmpeg then Sox

input: path to the audio/video

converts most video/audio formats to the desired/required one

```
ffmpeg -i $1 output.flac
sox output.flac -b 16 output2.flac rate 16k
rm output.flac
```

KALDI – ASR Toolkit



top-level, customize them, write own

- Feature Extraction: standard MFCC and PLP features; cepstral mean and variance normalization, LDA, TC/MLLT, HLDA, etc.
- Acoustic Modeling: conventional models (i.e. diagonal GMMs) and subspace Gaussian mixture models (SGMMs), extensions to new kinds of models.
- Phonetic Decision Trees: HMM
- Language Modeling: FST-based; IRSTLM toolkit.
- Decoding Graphs: Weighted Finite State Transducers (WFSTs)
- Decoders: simple to highly optimized.

Why KALDI?

- GoogleSpeechCloudAPI : long audios to be stored and downloaded from cloud, needs account permission and sync, no offline videos, defeats our purpose
- best WER documented than PocketSphinx
- available scripts, various models
- available free corpora with the official toolkit
- besides, we wanted to learn how to work with it and adapt it to our needs.

Testing with Yes/No Corpus

Corpus	Algorithm	Audio	Word-Error Rate
Yes/No	Monophone	Yes/No audios	0.00%

- preliminary experiment
- distinguish between spoken word 'Yes' and 'No'
- dataset: individual recording of the word multiple times
- language: Hebrew
- acoustic model: monophone
- 60 audios. 31 for training, 29 test data.

Training with Librispeech Corpus

Corpus	Algorithm	Audio	Word-Error Rate
Librispeech	Monophone	Librispeech	57.64%
Librispeech	Tri1	Librispeech(test)	30.82%
Librispeech	Tri2b	Librispeech(test)	24.44%

- downloaded from OpenSLR
- audios of people reading English text (1000 hrs.)
- audios segmented into utterances: start, end, ID, corresponding transcriptions
- limited memory and RAM, mini-sized corpus used only
- train set: 5 hrs., dev set: 3 hrs., test set: 5 hrs.
- 3 models : monophone, Tri1, Tri2b
- Features: LDA

Testing with Computer Science Domain Data

- test data: YouTube channel, Computerphile, videos
  - lectures, talks, monologues on Computer Science topics
  - wrote script to extract audios and import transcriptions/subtitles directly from YouTube videos
  - 5 videos with transcriptions
- uniform performance metric, Word-error-rate, our customised script

Testing with Computer Science Domain Data

- trained on Librispeech Tril model
- treated 1 audio as 1 utterance

Corpus	Algorithm	Audio	Word-Error Rate
Librispeech	1ril	CPAudio1	143.85%

Transcription Prediction

WELL AS I ¡UNK¿ ROUN AN ROUN ¡UNK¿ ¡UNK¿ ¡UNK¿ OR MY ADDRESS ¡UNK¿ TO ¡UNK¿ THE ¡UNK¿ AND SON UNLESS IT IS ONE WHO KNEW NEITHER HIS IS THE ¡UNK¿ OF ITS OWN THAT ITS IMAGES IN QUESTION

Original Transcription

we had this idea of a machine having or a network card or a wifi card having a 'mac' address which I understand to be a unique address to that dev not necessarily that device but certainly to that network interface (that's probably the best word for it is it?) So the question is why do we need IP addresses if we've got mac addresses? It's an interesting question

Training with Tedlium Corpus

Why this?

- real bad results from previous model
- structure of TED-talks matched Computerphile
- another freely available with Kaldi
- reasonable sized relevant corpus
- options to extend acoustic model with deep neural networks

Training with Tedlium Corpus

Corpus	Algorithm	Audio	Word-Error Rate
TED-Lium	chain TDNN	TED-Lium(test)	15.0%

- TED talks with cleaned automatic transcripts:
- distributed under 'Creative Commons BY-NC-ND 3.0' license
- size: 32 GB
- audios segmented into 3 seconds utterances, first 10k only
- acoustic model : chain Time Delay Neural Networks model
- 4-gram language model
- train set: 212 hours, test set: 10 hours
- training for 4 epochs only

Testing with Computer Science Domain Data

Corpus	Algorithm	Audio	Word-Error Rate
TED-Lium	DNN-pretrained	CPAudio2	22.96%

Transcription Prediction

when i was a kid the disaster we worry about most was a nuclear war. that's why we had a bear like this down our basement filled with cans of food and water. nuclear attack came we were supposed to go downstairs hunker down and eat out of that barrel. today the greatest risk of global catastrophe. don't look like this instead it looks like this. if anything kills over ten million people in the next few decades it's most likely to be a highly infectious virus rather than a war. not missiles that microbes now part of the reason for this is that we have invested a huge amount in nuclear deterrence we've actually invested very little in a system to stop an epidemic. we're not ready for the next epidemic.

Original Transcription

When I was a kid, the disaster we worried about most was a nuclear war. That's why we had a barrel like this down in our basement, filled with cans of food and water. When the nuclear attack came, we were supposed to go downstairs, hunker down, and eat out of that barrel. Today the greatest risk of global catastrophe doesn't look like this. Instead, it looks like this. If anything kills over 10 million people in the next few decades, it's most likely to be a highly infectious virus rather than a war. Not missiles, but microbes. Now, part of the reason for this is that we've invested a huge amount in nuclear deterrents. But we've actually invested very little in a system to stop an epidemic. We're not ready for the next epidemic.

Computer Science Words

Hand-picked Computer Science words

['computer', 'hash', 'array', 'programming', 'number', 'float', 'double', 'integer', 'encrypt', 'password', 'user', 'machine', 'learning', 'network']

- WordNet:
  - get the synsets of every word
  - add to the list
- Word Embeddings:
  - get the top 5 closest words in the vector space
  - GloVe 50 dimensions
  - Wikipedia 2014 + Gigaword 5.

Computer Science Words

Generated Computer Science words

{'nonzero', 'ice-cream_soda', 'non-negative', 'passwords', 'software', 'authentication', 'cryptographic', 'internet', 'skills', 'application', 'password', 'learning', 'car', 'technology', 'numeral', 'login', 'channel', 'straight', 'bivalent', 'integers', 'decrypt', 'least', 'user', 'determine', 'single', 'phone_number', 'upside', 'electronic', 'device', 'memorize', 'scheduling', 'encrypts', 'combining', 'act', 'count', 'floated', 'gun', 'sail', 'encrypted', 'drug_user', 'using', 'username', 'users', 'double', 'practical', 'server', 'formula_1', 'floating', 'sha-1', 'tying', 'integer', 'eruditeness', 'sophisticated', 'non-zero', 'ranging', 'types', 'net', 'creating', 'align', 'hash', 'array', 'duplicate', 'calculator', 'enables', 'interactive', 'annotate', 'code', 'channels', 'teach', 'experience', 'learn', 'third', 'float', 'air_bladder', 'numbers', 'only', 'encrypting', 'ten', 'network', 'networks', 'double_over', 'multiplication', 'computers', 'floats', 'format', 'issue', 'range', 'number', 'interface', 'teaching', 'doubling', 'hashish', 'machine', 'cable', 'triple', 'other', 'total', 'doubly', 'programming', 'program', 'knowledge', 'exploiter', 'used', 'computer', 'machines', 'hashes', 'md5'}

- integrated into DNN-based Tedlium model
- awaiting results (1 epoch, w-e-r: 25.6%)

Issues Faced

- Kaldi failed in Windows OS, delayed start
- limited RAM and memory in virtual Ubuntu
- prolonged running time and hence, debugging time
- could not use entire corpus
- could not use parallel architecture
- most free OpenSLR data not structured in required format (that of test corpus)
- different models vary in formats, metrics; difficult tweaking

Future Work

- transfer project on GPUs or AWS
- train with entire Tedlium Corpus
- use theano-based LSTM-RNNs
- improve Computer Science list

We learned how to:

- make KALDI work with different types of corpora.
- build different parts of an ASR
- modify a lexicon, making it more domain-specific
- run, edit and write own shell/bash scripts.
- work with Ubuntu in Virtual box environment
- directly download audios form YouTube videos, extract audios from videos and convert to various audio formats; import subtitles text from YouTube videos

References

- [1] Tanel Alum ae. Full-duplex Speech-to-text System for Estonian. Kaunas, Lithuania, 2014.
- [2] Daniel Povey et al. “The kaldi speech recognition toolkit”. In: In IEEE 2011 workshop. 2011.
- http://kaldi-asr.org