Speech recognition PBL proposal

Recognition Helpers+

Topic

Speech & emotion Recognition - Development of voice and emotion recognition applications for the hearing-impaired users.

Related works and literature survey

- Naver의 CLOVA Note : Speech recognition is performed on the uploaded speech file, and if there are several speakers, text is divided for each speaker and a transcript is prepared.
- Google Document App, 음성인식으로 메모 App : You can write with Speech Recognition.
- Writing function with speech recognition mounted on smartphone keypad.
- SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition (data augmentation methods) - https://arxiv.org/abs/1904.08779
- CNN-LSTM Emotional recognition model - https://koreascience.kr/article/CFKO202133648871932.pdf
- Lee, D., Lim, M., Park, H., & Kim, J.-H. (2017, February 28). LSTM RNN-based Korean Speech Recognition System Using CTC. Journal of Digital Contents Society. Digital Contents Society. https://doi.org/10.9728/dcs.2017.18.1.93
- 서민지 and 김명호. (2019). 음성의 감정 인식을 위한 감정 분류기 앙상블 기법. 한국IT정책경영학회 논문지, 11(2), 1187-1193. https://www.kci.go.kr/kciportal/ci/sereArticleSearch/ciSereArtiView.kci?sereArticleSearchBean.artiId=ART002460260
- 임명진, 이명호, 신주현. (2021). 딥러닝 기반 감정인식 성능향상 방법. 한국차세대컴퓨팅학회 논문지, 17(5), 88-95. https://m.earticle.net/Article/A402856

potential use case/scenario

Develop an application that starts voice recognition by pressing a button and converts the speaker's words and emotions (atmosphere) into text and displays them on the screen.

- **Conversations in 1:1 situations**
  In 1:1 conversation, press the button to start voice recognition. When the other person speaks, the other person's words are converted into text in real-time through speech recognition, and the speaker's emotional state at the moment of speaking is analyzed and displayed in text together. By using a speech recognition application that predicts emotional conditions in addition to reading emotions only by looking at the face, hearing-impaired users can better understand conversation situations.
- **Use in a lecture or meeting**
  Start speech recognition by pressing a button in a lecture or meeting. When the presenter speaks, it recognizes the speaker's words and emotions and displays them in text. This allows hearing-impaired users to better understand what words and atmosphere the presenter tried to convey.

plans

- **model and application**
  Provide users with results from the integration of the two models in the same format as the script instructions.
  Example sentence: "(Angrily) You said it first."
  1. Speech Recognition Model - Learning by extracting speech features such as Mel-frequency cepstral coefficients (MFCC) using librosa library, and then developing speech recognition models as RNN models using PyTorch library.
  2. Emotion recognition model - use the librosa library to extract features such as Mel-frequency cepstral coeffects (MFCC), chroma (pitch-related information), spectral

contrast, and RMS. Then, using the PyTorch library, we develop an emotion recognition model as a CNN model and train it.

3. In the case of applications, the learned model is uploaded to AWS and the learned model and application are produced to recognize voice through web communication. If the modeling takes a long time, developing applications may be omitted, so the parts related to the developing application in the timeline have been tagged as "optional."

- **responsibility**
    - project management : 손찬혁
    - documentation & model evaluate : 손열혼
    - speech recognition model : 황세현, 손윤석, 매튜 융만
    - emotion recognition model : 손찬혁

- **timeline**
  Repeat the weekly sprint to proceed with the project
    - sprint 1:
      Define project requirements.
      Set up development environment (librosa, pytorch, cuda…)

    - sprint 2~6
      Develop model, Train model, Evaluate model performance
      Evaluate model performance (using accuracy metrics such as word error rate, sentence error rate, recall, F1 score, and cosine similarity of entire sentences to evaluate speech recognition performance
      Evaluate emotion recognition by sentence-level emotion recognition accuracy).

    - sprint 7~8
      Integrate results from the model to create a component that displays integrated speech and emotion recognition results.
      Optional: Develop an Android app using Java or Kotlin.
      Optional: Deploy the integrated model on AWS and communicate with the app via web communication.

    - sprint 9~10
      Test the system and fix bugs

      Document the project (design document, user guide, developer document, etc.) as development progresses.

- **tool/library/stack/repository**
    - Tools to be used
        - librosa(https://librosa.org/doc/latest/index.html) : Pre-processing and feature extraction of audio data.
        - esp-net(https://github.com/espnet/espnet) : End-to-End Speech Processing Toolkit
        - pytorch(https://pytorch.org/) : Used for neural network training.

    - Optional tools
        - soundfile(https://pypi.org/project/soundfile/) : Used to import file formats not supported by librosa.
        - hugging face:

        - example1: https://huggingface.co/keras-io/ctc_asr
        - esp-net: https://huggingface.co/espnet/kan-bayashi_ljspeech_vits

    - Speech recognition datasets
        - OpenSLR :https://www.openslr.org/resources.php
        - Free Conversation Speech (General Men and Women) : https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=109
        - OpenSLR (Korean speech data) : https://www.openslr.org/40/
        - Public Data : https://www.data.go.kr/data/15073486/fileData.do

    - Emotion recognition datasets
        - emotion_dataset : https://aihub.or.kr/aihubdata/data/view.do?currMenu=&topMenu=&dataSetSn=263&aihubDataSe=extrldata

    - repository - github : https://github.com/SonChanhyuk/RecognitionHelpers

    - PM tool - Notion

- **apparatus**
    - cluster server: Used for data training.
    - Optional: AWS for web communication in the case of developing an application

- **system evaluation plan**
    - Sentence similarity calculation (accuracy, word error rate, sentence error rate, recall, F1 score).
    - Evaluating by vectorizing two sentences with sklearn's TF-IDF vectorizer and comparing cosine similarity.

- **documentation plan**
    - User manual
    - Model documentation: Model structure and functions used
    - How to use the training dataset and source information.
    - Model evaluate report

- **related links**:
    - rnn-t(google blog): https://ai.googleblog.com/2019/03/an-all-neural-on-device-speech.html
    - rnn-t(pytorch): https://github.com/sooftware/RNN-Transducer
    - real time speech recognition info: https://blog.naver.com/PostView.naver?blogId=nuguai&logNo=222425372642&parentCategoryNo=&categoryNo=9&viewDate=&isShowPopularPosts=true&from=search