**1페이지**

안녕하세요 발표를 맡은 ㅇㅇㅇ입니다.

**2페이지**

저희는 기존 TTS의 문제점과 개선 방안/

데이터 전처리/

데이터 통계/

데이터 처리 결과 및 개선점/

향후 진행 계획 순으로 설명 드리겠습니다.

**3페이지**

먼저 TTS란 ＂Text-to-Speech＂의 약어로 텍스트를 음성으로 변환하는 기술을 의미하고

오디오북, 네비게이션, 시각장애인 보조 도구, 교육 어플리케이션 등 다양한 분야에서 활용됩니다.

저희는 대중적으로 사용되는 TTS의 외래어 발음 정확도를 테스트 문장을 통해 실험해보았습니다.

먼저 네이버의 파파고를 이용하여 실험 해본 결과, 보이는 것과 같이 WCDMA를 ‘크드마＇로

마이크로소프트의 bing을 이용하여 실험 해본 결과 UCLA를 ‘유시알에이’로

잘못된 발음으로 변환하는 것을 확인할 수 있었습니다.

**4페이지**

앞선 슬라이드에서 보신 것과 같이 상용 TTS들이 외래어를 정확한 한국어 발음으로 변환하지 못하는 이유는

외래어 부분에 있어 G2P 모델의 정확성이 상대적으로 떨어지기 때문인데

저희는 우리말샘 사전을 기반으로 한 발음열 기분석 사전을 이용하여 이를 개선할 예정입니다.

**5페이지**

먼저 우리말샘 사전에 존재하는 전체 단어 데이터를 다운 받은 뒤 필요한 형태로 전처리를 하였습니다.

Sense number가 001인 경우 즉, 사전에서 단어를 찾았을 때, 첫번째로 나오는 뜻만 저장하고,

북한어나 방언을 필터링하여 일반어만 저장하도록 하였습니다.

**6페이지**

앞에서 저장한 데이터를 고유어, 한자어, 외래어, 혼종어 4가지로 나누어 각각 DB로 저장하였습니다.

여기서 구분되어진 어에 대해 간단히 설명해보면

고유어는 순수한 한글로만 이루어진 단어이고

한자어는 하나 또는 둘 이상의 한자가 결합되어 사용되는 단어이고

외래어는 한자어를 제외한 외국어로 이루어진 단어를 의미합니다.

마지막으로 혼종어는 서로 다른 기원을 지닌 형태소가 합쳐져서 만들어진 단어를 뜻하는데,

예를 들어 페트병이나 종이컵 등이 있습니다.

**7페이지**

그리고 전체 DB에서 단어의 활용형들의 표기와 발음만을 따로 추출하여 새로운 활용어 DB 또한 생성하였습니다.

**8페이지**

이 표는 외래어 DB 속 저장된 데이터의 형식을 보여주는 표입니다.

Pronun\_list 는 단어가 실제로 어떻게 발음되는지 소리나는 대로 표기한 것이며

origin\_lang 의 경우 해당 단어가 한국어로 변환되기 전, 원래 형태를 보여주는 것이고

orgin\_lang\_type 은 해당 단어의 어원을 나타냅니다

이번 과제에서는 외래어 중에 어원이 영어인 단어들만 사용하기로 하였습니다.

**9페이지**

우리말샘 사전에 존재하는 전체 단어의 개수는 총 1,164,962개 였으나, 이번 과제 필요한 형태로 필터링하여 저장한 결과 총 779,429개로 줄일 수 있었습니다.

**10페이지**

KT에서 제공한 학습 데이터를 앞서 구축한 외래어 DB를 이용하여 매칭 시켜본 결과 통계는 다음과 같았습니다.

여기서 대소문자를 구분하여 통계를 낸 이유는 영어 단어를 DB와 매칭하는 순서가

첫번째로 문장에서 영어 단어를 뽑아와서 대소문자를 구분하여 DB에서 확인하고,

DB에 존재하지 않을 경우, 대소문자를 구분하지 않고 재확인한 뒤,

그래도 DB에 존재하지 않으면 알파벳 단위로 끊어서 읽는 방식으로 진행되기 때문입니다.

**11페이지**

데이터를 전처리하고 학습 데이터를 DB와 매칭시켜 본 결과, 몇 가지 보완해야 할 점이 보였습니다.

사전에 DB를 구축할 때, sense no 가 001인 경우만 필터링 하여 저장하도록 하였기 때문에

sense no이 001이 아닐 경우 DB에 존재하지 않아서 발음열 매칭이 불가능합니다.

아래의 예시를 보면 ‘엔’의 경우는 일본의 화폐 단위를 뜻하는 엔화가 sense no이 001이기 때문에

영어 알파벳을 뜻하는 ‘N’의 경우는 DB에 저장되지 않습니다.

따라서 아래의 문장 같은 경우, N극의 ‘N’ 발음을 제대로 변환하지 못합니다.

**12페이지**

또한 구축한 DB에 존재하지 않으면서 대문자의 경우 알파벳 단위로 끊어서 그대로 발음하는 방법이 있지만,

해당 단어가 DB에 존재하지 않으면서 대문자로만 이루어지지 않은 경우에 문제가 발생합니다.

예를 들어 아래의 문장의 iOS, Hello, World 는 DB에 존재하지 않고 대소문자가 혼용되어 알파벳 그대로 발음할 수도 없게 됩니다.

**13페이지**

따라서 sense number로 인해 DB에 저장되지 않은 단어와

우리말샘 사전에 미등재된 단어로 인해 DB만으로 외래어 TTS를 구현하기에 부족합니다.

이를 보완하기 위해 transformer 나 LSTM 방식을 사용한 딥러닝 모델을 구축하여 DB에 없는 단어들을 처리하려고 계획 중입니다.

저희가 준비한 발표는 여기까지입니다. 감사합니다.