202402960 신새봄

스페인어 동사 분류기

1. 프로젝트 개요

1.1.

스페인어는 동사의 활용이 다양하고 복잡한 언어입니다. 하나의 동사가 법, 시제, 인칭에 따라 100가지 이상의 활용형을 가집니다. 따라서 학습 시 문장 속에서 변형된 동사를 접할 때, 그 문법적 형태를 파악하기 어려운 경우가 많습니다. 그러나 기존의 사전과 번역기는 동사의 문법적 역할보다는 의미에 초점을 두어 문법적인 형태 정보를 제공하지 않습니다. 이러한 문제점을 해결하고자 이번 프로젝트를 통해 스페인어 동사 분류기를 만들게 되었습니다.

이 도구는 변형된 형태의 동사를 입력으로 받아, 그 동사가 지닌 세 가지의 문법적 요소, 법(Mood), 시제(Tense), 인칭(Person)을 동시에 예측하고 분류합니다. 이번 프로젝트의 목표 중 하나가 '실제로 5번 이상 사용할 만한 도구를 만드는 것'이었기 때문에, 사용의 편의성을 중요하게 고려하였습니다. 문장을 입력하면 문맥을 반영하여 정확도는 높일 수 있지만, 불편하여 결국 사용하지 않게 될 것이라 생각하였습니다. 따라서 입력을 문장 전체가 아닌 동사 활용형으로 정하였습니다.

2. 진행 과정

2.1. 데이터 수집 및 분석

모델의 학습을 위해 처음 수집한 데이터셋에는 스페인어의 모든 품사가 혼재되어 있었습니다. 이 데이터셋에서 동사 원형을 선별하여 규칙 기반 활용형 생성기를 만들고, [동사 활용형 – 라벨] 형태의 CSV 데이터셋을 구성할 계획을 세웠습니다. 이에 대하여 "규칙 기반 생성기를 만들 수 있다면 왜 그걸 다시 ML로 만들어야 하나?"라는 의문이 든다는 리뷰를 받았습니다.

ML을 통하여 스페인어 동사 분류기를 만드는 가장 큰 장점은 불규칙 동사의 복잡성을 효과적으로 다룰 수 있다는 점입니다. 스페인어는 동사의 형태가 다양하게 변화하는 만큼 불규칙하게 변화하는 동사들도 많습니다. 이 불규칙 변화하는 동사들을 사람이 일일이 손으로 코드를 작성하여 예외를 두기에는 한계가 있습니다. 또한 코드로 작성한다고 하더라도 예외 속의 예외, 또는 모호한 하위 규칙이 존재하여 이를 코드로 표현하기가 쉽지 않습니다. 그러나 ML 통해 모델을 학습시킨다면 명확한 문장으로 표현될 수 없는

불규칙 동사들의 모호한 형태적 패턴을 모델이 학습하여 분류하는 것이 가능해집니다.

위의 이유들로 처음 계획하였던 데이터 수집 계획은 실행하지 못하였습니다. 규칙 기반 활용형 생성기를 통해 만들어진 데이터로 모델을 학습시키면 불규칙 변화 동사를 전혀 다룰 수 없는 모델이 되기 때문입니다. 따라서 규칙 동사와 불규칙 동사의 변형된 활용형을 모두 포함하고 있는 다른 데이터셋을 새로 확보하여 필요한 속성들만 선별하여 최종 CSV로 정리하였습니다.

데이터의 수는 총 65,927개로 verb, mood, tense, person 단어와 총 3가지의 주요 속성들로 구성되어 있습니다. Mood는 indicative, subjunctive, imperative 3가지, tense는 present, present_perfect, pluperfect, future_perfect, future, imperfect, preterite_anterior, conditional, conditional_perfect, preterite 10가지, person은 1sg부터 3pl까지 총 6가지로 구성되어 있습니다. 동사 활용형의 길이는 주로 5~10글자였으며 대부분의 라벨들은 서로 낮은 상관관계를 보였습니다.

2.2. ML 모델 학습 및 평가

스페인어 동사 분류기를 만들기 위하여 Character-based Multi-Head LSTM 모델을 학습시켰습니다. 입력이 문장이 아닌 단어이기 때문에 문맥의 고려가 필요하지 않다고 생각하여 Char기반의 모델을 선택하였습니다.

데이터셋은 train 80%, validation 10%, test 10%로 나누었습니다. 학습을 시작하기 위해 우선 입력된 동사 활용형을 문자 단위로 토큰화 합니다. 이후 글자 하나하나로 쪼개어 각 고유한 토큰을 부여하고, 토큰화된 각 문자를 벡터로 변환하여 임베딩 합니다.

변환된 문자 벡터들은 LSTM 층을 통과합니다. 양방향 LSTM을 활용하여 정보를 읽어내고, 두 방향의 분석 결과를 결합하여 동사 활용형이 가진 모든 형태소적 정보를 압축한 문맥 벡터를 생성합니다. 이후 문맥 벡터는 mood, tense, person 세 개의 독립적인 head로 전달이 됩니다.

모델의 오류는 mood, tense, person 각 head의 오류를 계산한 후, 이 세 가지 오류 값을 단순히 합산하여 모델의 최종 오류(total loss)로 정의했습니다. 모델은 이 총 오류를 최소화하는 방향으로 학습을 진행하여, 세 가지 특징 모두에서 동시에 성능이 향상되도록 하였습니다. 학습 속도를 높이고 효율성을 올리기 위해 Adam 최적화 도구를 사용했습니다.

또한 과적합 방지를 위해 Dropout 기법을 적용했습니다. 모델 성능이 더 이상 개선되지 않으면 훈련을 종료시키도록 Early Stopping기능을 적용하였습니다.

초기 모델의 정확도는 약 0.86이었고 이 모델의 성능을 높이기 위해 여러 변화를 주어 다양한 다른 모델들을 실행시켰습니다. 은닉층을 감소시키기도 하고 증가시키기도 하며 학습률, 에포크 수도 다양하게 시도하였습니다. Attention을 추가하였으나 큰 성능의 변화가 있지는 않았습니다. 입력이 단일 단어이며 문맥 정보가 없기 때문에 Attention의 효과가 없었던 것으로 판단됩니다. 또한 4th assignment PR에 class weight를 조정해 보라는 리뷰가 달려 class weight를 조정해 보았으나 성능의 변화는 없었습니다. 총 10가지의 모델을 실행한 결과, 3개의 은닉층, 20에포크, 1e-3학습률을 가진 모델이 가장 높은 성능을 보여 최종 모델로 선정하였습니다.

모델의 성능은 다음과 같습니다.

평가 지표 **(Evaluation Metric)**	결과 **(Result)**
Exact Accuracy (mood, tense, person 모두 정확)	**0.8715**
Individual Head Accuracy	
Mood	0.9690
Tense	0.9895
Person	0.8980
Individual Head Macro-F1	
Mood	0.9199
Tense	0.9883
Person	0.8945

세 가지 특징을 모두 맞추는 Exact Accuracy는 87.15%로, 스페인어 활용의 복잡성을 고려했을 때 꽤 괜찮은 결과를 얻었습니다.

3. 모델을 서비스로 만든 구조

3.1. 학습 모델의 서비스 전환

학습을 마친 모델을 실제로 사용할 수 있도록 CLI 형태의 서비스로 구현했습니다. 사용자는 터미널 환경에서 필요한 패키지를 설치한 후 모델을 실행하여 동사 활용형을 입력하고 결과를 즉시 확인할 수 있습니다. 또한 'quit'을 입력하면 프로그램이 종료됩니다.

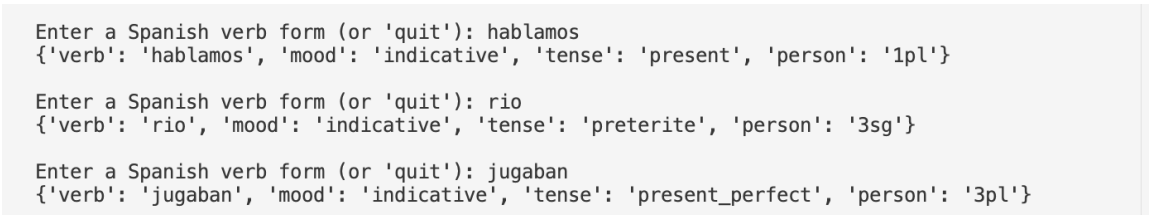
4. 실제 사용 결과

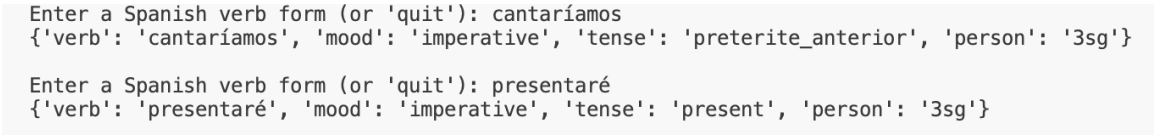
```
Enter a Spanish verb form (or 'quit'): hubiera tenido
{'verb': 'hubiera tenido', 'mood': 'indicative', 'tense': 'preterite_anterior', 'person': '3sg'}

Enter a Spanish verb form (or 'quit'): visitaba
{'verb': 'visitaba', 'mood': 'indicative', 'tense': 'preterite_anterior', 'person': '2sg'}

Enter a Spanish verb form (or 'quit'):
```

```
Enter a Spanish verb form (or 'quit'): había terminado
{'verb': 'había terminado', 'mood': 'indicative', 'tense': 'future_perfect', 'person': '3pl'}
```





실제 사용에서의 모델은 테스트 데이터셋 평가 시의 성능보다 낮은 정답률을 보였습니다. 기존의 테스트 데이터셋을 활용하여 평가했을 때와는 달리, 법과 시제의 분류에서 오답률이 높게 나타났습니다.

5. 개선사항 및 배운 점

5.1. 개선사항

모델을 학습시키며 0.9 이상의 정확도를 달성하고자 다양한 시도를 했으나, 이를 넘어서지 못했습니다. 스페인어는 동사 활용형이 다양한 만큼, 같은 철자 형태를 가지면서 서로 다른 문법적 속성을 가지는 동사들이 존재합니다. 문맥의 고려 없이 단어의 형태로만 head를 예측하는 모델은 이러한 같은 형태의 다른 동사를 구분할 수 없습니다. 이를 해결하기 위해서는 활용형 뿐만 아니라 해당 동사의 원형이 포함된 대규모 데이터셋이 필요할 것 같습니다.

또한 모델을 학습시키고 테스트 데이터셋으로 평가했던 것과 달리 실제 사용에서는 예측했던 것보다 더 많은 오류가 발생했습니다. 이 문제는 데이터 편향성에서 비롯된 문제로 보입니다. 학습에 사용된 데이터셋은 주로 동일한 동사 원형에서 파생된 다양한 활용형들로 구성되어 있습니다. 테스트셋은 훈련셋에서 이미 등장한 동사 원형의 다른 활용형들이 대부분이었습니다. 반면 실제 사용에서는 모델이 학습 과정에서 한 번도 접해보지 못한 새로운 동사 원형의 활용형이 입력으로 들어왔고, 이로 인해 성능이 크게 떨어진 것으로 보입니다.

5.2. 배운 점

이번 프로젝트를 통해 문제의 정의부터 실제로 도구를 개발하는 모든 과정을 경험할 수 있어 좋았습니다. 이론에서 배운 개념들을 실제로 학습에 사용하여 보니 글로만 보았던 것들이 어떻게 작동하는지 생생하게 알 수 있었습니다. 또한 생각했던 것보다 적절한 방법을 찾아 코드를 구현하는 과정이 어렵다고 생각되었습니다. 너무 다양하게 존재하는 모델들과 기법들 중 어떤 것을 적용해야 최적의 결과가 나오는지 생각하고 실행하는 것이 쉽지 않았습니다.

가장 크게 깨달은 것은 데이터의 품질과 구성의 중요성입니다. 어떤 데이터를 준비하여 어떤 형식으로 학습에 활용하는지가 최종 결과물에 큰 영향을 미친다는 것을 알게 되었습니다.

비록 실제로 사용할 수 있는 완벽한 모델을 만들지는 못하였지만 프로젝트의 전 과정을 완수함으로써 더 성장할 수 있었던 좋은 경험이었습니다.