

의도 정보를 활용한 다중 레이블 오픈 의도 분류

박나현^{o,1}, 조성민², 송현제²

¹전북대학교 IT정보공학과 ²전북대학교 전자정보대학원

{nastela, smcho, hyunje.song}@jbnu.ac.kr

Multi-label Open Intent Classification using Known Intent Information

Nahyeon Park^{o,1}, Seongmin Cho², Hyun-Je Song²

¹Department of Information Technology, Jeonbuk National University

²Graduate School of Electrical and Information Engineering, Jeonbuk National University

요약

다중 레이블 오픈 의도 분류란 다중 의도 분류와 오픈 의도 분류가 합쳐져 오픈 도메인을 가정하고 진행되는 다중 의도 분류 문제이다. 발화 속에는 여러 의도들이 존재한다. 이때 사전에 정의된 의도 여부만을 판별하는 것이 아니라 사전에 정의되어 있는 의도에 대해서만이라도 어떤 의도인지 분류할 수 있어야 한다. 본 논문에서는 발화 속 의도 정보를 활용하여 다중 레이블 오픈 의도를 분류하는 모델을 제안한다. 먼저, 문장의 의도 개수를 예측한다. 그리고 다중 레이블 의도 분류기를 통해 다중 레이블 의도 분류를 진행하여 의도 정보를 획득한다. 획득한 의도 정보 속 다중 의도 개수와 전체 의도 개수를 비교하여 전체 의도 개수가 더 많다면 오픈 의도가 존재한다고 판단한다. 실험 결과 제안한 방법은 MixATIS의 75% 의도에서 정확도 94.49, F1 97.44, MixSNIPS에서는 정확도 86.92, F1 92.96의 성능을 보여준다.

주제어: 의도 정보, 다중 레이블 분류, 오픈 의도 분류

1. 서론

의도 분류는 발화 속에 포함되어 있는 의도를 분류하는 문제이다. 사용자들의 의도를 알아내어 정보를 제공하거나 대화를 만족하는 방향으로 이어나가는 챗봇, ChatGPT 등 AI 어시스턴트들의 발달로 인해, 의도 분류는 중요한 문제로 여겨진다. 기존 의도 분류들은 발화 텍스트를 입력으로 받아 사전에 정의된 의도 중에 하나로 분류하는, 즉 닫힌 세계를 가정한 의도 분류가 진행된다[1, 2, 3, 4]. 하지만 사용자의 의도를 우리가 전부 고려할 수 없어 닫힌 세계를 가정할 경우 잘못된 의도를 예측한다. 발화 텍스트 속 의도가 사전에 정의된 의도 중에 존재하지 않아 알 수 없는 의도를 오픈 의도라고 한다[5]. 실제 세계에는 오픈 의도가 존재하므로 적용 가능한 오픈 도메인에서 의도를 분류하는 연구들이 많이 주목을 받으며 연구되고 있다[5, 6, 7].

일반적인 닫힌 세계에서의 의도 분류는 표1에서 1번째와 같이 하나의 의도를 가진 발화 속 의도에 대해 예측하는 것이다. 단일 의도 분류지만 오픈 도메인에서의 대부분의 기존 연구에서는 표1에서 2번째 발화 속 하나의 의도가 오픈으로 존재할 경우 오픈으로 분류하는 연구들이 많이 진행되었고 좋은 성능을 보여주고 있다[5, 6]. 그러나 실제 세계에서는 표1에서 3번째와 같이 하나의 발화 속 여러 의도가 존재하는 경우가 많이 존재한다. 이 문제에 대해 풀기 위해 대부분의 기존 다중 의도 분류 모델들은 닫힌 세계에서 다중 의도 탐지 문제만을 위해 사전에 정의된 의도만을 탐지한다[8, 9]. 실제 세계에 존재하는 오픈 의도를 고려하여 단일 의도 분류일 때와 마찬가지로 오픈

표 1. 주방과 식사 도메인에 대한 의도 분류 예시

순번	발화 텍스트	예측 라벨	방법 및 종류
1	I need a recipe for chicken pot pie.	Recipe	single intent classification
2	My card is unusable because it's damaged	Open	single open intent classification
3	Give me a beef recipe and what kind of nutritional content do carrots have?	Recipe, Nutrition Info	multi intent classification
4	Can you find me a recipe for pie and set a timer for two minutes.	Open	multi open intent detection
5	Can you find me a recipe for pie and set a timer for two minutes.	Recipe, Open	multi open intent classification

도메인에서의 다중 의도 탐지 연구도 진행되고 있다. 이에 대한 연구들은 알고 있는 의도와 모르는 의도에 관하여 이진 분류를 진행한다[7]. 그러므로 표1의 4번째 발화에서는 오픈으로 탐지만 가능하다. 만약 사전에 정의된 의도와 모르는 의도 모두 포함된 다중 의도 발화에서, AI 어시스턴트는 알고 있는 의도에 대해서는 이에 맞는 기능을 수행하고 그 후 자신이 모르는 의도에 대해서는 사용자에게 다시 질문할 수 있어야 한다. 하지만 다중 의도 탐지를 이진 분류로 진행하게 되면 발화 속에 어떤 의도가 존재하는지를 알 수 없다. 그러므로 다중 의도 분류 문제에서는 여러 의도 중 알고 있는 의도인지 혹은 모르고 있는 의도인지 판단하는 것을 넘어서 알고 있는 의도에 대해서

는 어떤 의도인지까지 파악할 수 있도록 표1의 5번째 발화를 예측한 것과 같은 결과가 필요하다.

본 논문에서는 의도 정보를 활용하여 다중 레이블 오픈 의도를 분류하는 모델을 제안한다. 제안한 모델은 다중 레이블 분류와 오픈 의도 분류를 동시에 수행한다. 다중 레이블 분류를 수행할 때, 모델이 출력한 예측값은 임계값을 넘지 못하면 탐지되지 않는다. 탐지된 다중 레이블 분류의 결과를 의도 정보라고 한다. 오픈 의도 분류를 위해서, 제안 모델은 발화 내의 전체 의도 레이블 개수를 예측한다. 예측된 전체 의도 개수와 의도 정보를 비교하여 전체 의도 개수가 더 많다면 오픈 의도로 분류된다. 예를 들어 표1와 같이 발화 텍스트 속 의도가 두 개 존재하고 문장 속 사전에 정의된 의도가 ‘Recipe’이라고 예측하면 총 두 개의 의도 중 나머지는 오픈 의도라고 판별할 수 있다.

제안한 모델의 성능을 평가하기 위해서 다중 의도 데이터셋인 MixATIS, MixSNIPS[9]로 실험을 진행했다. 기존 연구의 실험 설정은 전체 의도 클래스 중 25%, 50%, 75%를 알고 있는 의도로 간주하고 본 논문의 실험도 위 설정을 따른다. 알고 있는 의도 클래스 비율이 75%일 때 제안 모델은 MixATIS 데이터셋에 대해 정확도 94.90%, F1 82.69%, MixSNIPS 데이터셋에 대해서 정확도 88.47, F1 85.58을 기록한다.

2. 관련 연구

2.1 오픈 의도 탐지

오픈 의도 탐지는 닫힌 세계를 가정하는 기존 연구와 다르게 오픈 도메인에서 문장 속 의도를 탐지하는 문제이다. 오픈 도메인에서 단일 의도 분류는 좋은 성능을 보이고 있다[5, 6]. 오픈 의도 탐지 문제는 크게 두 가지 방식으로 나뉘며 연구가 진행되고 있다. 학습하는 데이터에 존재하는 의도만을 가지고 학습을 진행하는 방법과 추가로 알지 못하는 의도의 라벨인 오픈이라는 라벨을 학습 전에 추가하여 학습 진행 시 기존 데이터의 의도에 하나가 추가된 채로 학습을 진행하는 방법이 존재한다. 학습 시 기존 데이터에 존재하는 의도만을 가지고 학습하는 가장 대표적인 방법으로 ADB와 KNNCL이 존재한다[5, 6]. ADB는 단순하면서도 좋은 성능을 보이며 오픈 의도 탐지의 대표적인 방법으로 손꼽히고 있다. 학습 데이터에 존재하는 각각의 의도별 결정 경계를 학습하여 발화 텍스트가 입력으로 들어올 경우 결정 경계에 속하는지를 파악하여 특정 결정 경계에 속하면 가장 가까운 의도로 판단하고, 어떤 결정 경계에도 속하지 않으면 오픈이라고 판단한다.

학습하는 데이터 속에 존재하지 않는 의도를 위해 학습 전 기존 의도 라벨에 오픈 의도 라벨을 추가하여 학습을 진행하는 방식의 연구 중 가장 대표적인 방법은 (K+1)-classifier 방식이다[10]. 위 방법은 외부 데이터를 추가하고 기존 학습 데이터를

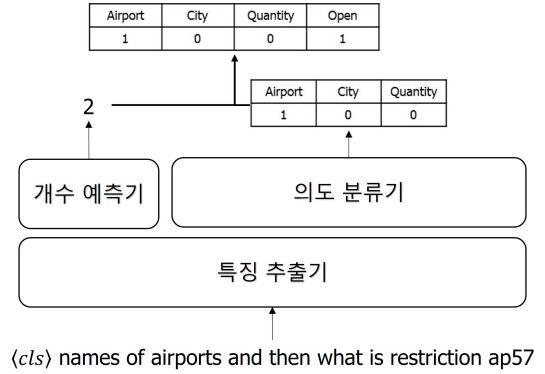


그림 1. 다중 레이블 오픈 의도 탐지 모델.

가지고 새로운 임시 데이터를 만들어 오픈 라벨 데이터로서 학습을 진행한다. 학습할 때부터 분류기를 기존 데이터의 의도뿐 아니라 오픈 라벨을 판별할 수 있도록 만들어 오픈 데이터에 대한 존재를 기존의 데이터와 함께 분류기가 학습하여 데이터가 새로 들어왔을 때 오픈에 대하여 판별할 수 있도록 하는 것이다.

2.2 다중 의도 탐지

다중 의도 탐지 문제에 대한 연구는 의도에 많은 영향을 끼치는 슬롯 채우기 문제와 의도 탐지 문제를 동시에 다루는 방식으로 연구가 많이 이루어지고 있다. 두 문제를 동시에 진행하는 모델로 비자기회귀 모델인 GL-GIN이 존재한다[8]. 또한 AGIF[9], SSRAN[11]은 토큰 수준에서 진행한 다중 의도 탐지 방식이다. SSRAN은 가장 최근에 발표된 방식으로 각각의 의도가 구역을 가지는 사실을 이용하여 GL-GIN보다 더 좋은 성능을 보여주었다. 이 방식들은 모두 닫힌 세계에서 진행하여 사전에 정의된 의도만을 탐지하지만 실제 세계에서는 하나의 발화 텍스트 속에 우리가 생각하지 못한 의도들 또한 존재한다.

닫힌 세계에서 나아가 오픈 도메인에서 다중 의도 탐지를 진행하는 연구로 AIK가 있다[7]. 위 방법은 먼저 한 문장 속 전체 의도 개수를 예측하고 문장 속 사전에 정의된 의도의 개수를 예측하여 두 개의 예측한 수를 비교한다. 전체 의도 개수와 알고 있는 의도의 개수가 같으면 사전에 정의된 의도만으로 이루어진 알고 있는 발화라고 말한다. 만약 두 개수가 다르다면 나머지 의도는 알지 못하는 의도가 존재하고 있다는 뜻을 의미하므로 오픈 의도라고 판단한다. 하지만 오픈 도메인에서 진행하는 다중 의도 탐지에서 모르는 의도가 존재하여 오픈 의도라고 탐지된 경우, 나머지 사전에 정의된 의도라 하여도 어떤 의도인지에 대해 알 수가 없다. 또한 다중 의도 모두 사전에 정의된 의도라고 탐지되었어도 각각이 어떤 의도인지에 대해 알 수 없다는 문제들이 존재한다.

3. 제안 방법

본 논문에서는 발화가 주어지면 여러 개의 의도를 탐지함과 동시에 오픈 의도를 탐지하는 모델을 제안한다. 제안한 모델은 그림 1과 같이 발화를 표현 벡터로 변환하는 특징 추출기, 멀티 라벨 분류 방식으로 여러 의도를 탐지하기 위한 의도 분류기와 의도 개수를 예측하는 개수 예측기로 구성된다. 텍스트가 입력되면 제안한 모델은 개수 예측기를 통해서 전체 의도 개수를 예측하고 의도 분류기는 멀티 레이블 예측 방식으로 의도를 탐지한다. 탐지된 의도의 개수가 전체 의도 개수보다 적다면 오픈 의도를 탐지한 것으로 간주한다. 제안한 모델의 특징 추출기는 BERT[12]와 같은 Pre-trained Language Model(PLM)이다. 특징 추출기는 문장 X 를 입력 받아 $L + 1$ 개의 토큰을 가진 표현 벡터 $\mathbf{H} = \{\mathbf{h}_{cls}, \mathbf{h}_1, \dots, \mathbf{h}_L\}$ 를 출력한다. i 번째 벡터 \mathbf{h}_i 의 크기는 d_{model} 이다. 제안한 모델의 의도 분류기는 표현 벡터 \mathbf{H} 를 입력으로 받아 C 개의 알고 있는 의도 정보를 출력하고 시그모이드 함수를 적용해 개별 의도 확률을 만든다. 의도 분류기는 개별 의도 확률이 0.5보다 클 경우, 개별 의도 예측값은 1이 되고, 0.5이하일 경우 해당 의도는 0으로 하여 멀티 레이블 분류를 수행한다. 개수 예측기는 \mathbf{h}_{cls} 를 입력으로 받아 발화 내에 포함되어 있는 전체 의도의 개수를 예측한다. 의도 분류기가 1로 분류한 의도들의 개수와 개수 예측기의 전체 의도 개수를 비교하여 오픈 의도를 탐지한다.

3.1 의도 분류기

하나의 발화 안에는 사전에 정의된 의도와 오픈 의도가 함께 존재할 수 있기 때문에, 사전에 정의된 의도만을 분류하는 의도 분류기의 다중 레이블 분류 성능이 하락할 수 있다. 이를 방지하고자 각각의 알고 있는 의도 표현에 대해 랜덤으로 초기화한 학습가능한 파라미터인 \mathbf{Q} 로 어텐션 스코어를 계산한다[7].

$$\mathbf{A} = \text{Attention}(\mathbf{Q}, \mathbf{H})$$

식 (1)과 같이, 표현 벡터 \mathbf{H} 에 어텐션 스코어를 반영하면 의도별 문장 표현 $\mathbf{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_i, \dots, \mathbf{v}_C\}$ 을 얻을 수 있다.

$$\mathbf{V} = \mathbf{A}^\top \mathbf{H} \quad (1)$$

i 번째 의도의 문장 표현인 \mathbf{v}_i 는 d_{model} 의 크기이다. \mathbf{v}_i 는 알고 있는 의도의 어텐션 반영하였기에 해당 발화가 i 번째 의도일 경우의 표현이라고 할 수 있다.

의도 정보인 Logit_{intent} 는 의도별 문장 표현 V 의 평균값이고 (2), 이를 이용하여 의도를 탐지한다 (3).

$$\text{Logit}_{intent} = \text{Pool}(\mathbf{V}) \quad (2)$$

전체 C 개의 의도를 개별적으로 파악해 최종 의도를 탐지한다. 이를 위해 개별 의도 정보 ($\text{Logit}_{intent,i}$)에 σ 함수를 취해 개별

의도 확률을 만든다.

$$Y'_i = \begin{cases} 0, & \text{if } \sigma(\text{Logit}_{intent,i}) \leq 0.5 \\ 1, & \text{if } \sigma(\text{Logit}_{intent,i}) > 0.5 \end{cases} \quad (3)$$

오픈 의도는 의도 정보와 개수 예측기의 값과 비교하여 탐지될 수 있다.

3.2 개수 예측기

제안한 모델은 오픈 의도를 분류하기 위해서 전체 의도 개수와 의도 예측값의 합을 비교한다. 의도 예측값의 합은 개별 의도 예측값을 모두 더한 값이고, 전체 의도 개수는 개수 예측기를 통해서 구해진다. 개수 예측기는 발화 내의 전체 의도 개수를 예측하는 선형 분류기이다. \mathbf{h}_{cls} 가 입력되면 개수 예측기는 하나의 실수 N' 를 출력한다.

개별 의도 예측값인 Y'_i 의 값은 0 혹은 1이기 때문에 합할 경우, 탐지한 의도들의 개수 K 를 얻을 수 있다.

$$K = \sum_{i=1}^C Y'_i \quad (4)$$

식 (4)를 통해 얻은 K 는 알고 있는 의도의 개수만을 의미하므로 개수 예측기가 출력한 N' 보다 작을 경우, 발화 내의 오픈 의도가 있다고 간주한다.

3.3 모델 학습

본 논문에서는 의도 분류기와 개수 예측기가 동시에 학습한다. 의도 분류기의 손실 함수는 식 (5)로 정의되고, Y_i 는 i 번째 클래스의 정답 레이블이다.

$$\mathcal{L}_{intent} = -\frac{1}{C} \sum_{i=1}^C \text{BCE}(Y_i, \sigma(\text{Logit}_{intent,i})) \quad (5)$$

개수 예측기의 손실 함수는 식 (6)로 정의되고, N 은 의도 개수의 정답 값이다.

$$\mathcal{L}_{num} = \text{MSE}(N - N') \quad (6)$$

위의 손실 함수 외에도, 알고 있는 의도의 표현을 더 잘 학습하기 위해서 추가적인 손실 함수를 도입한다.

식 (7)은 의도별 문장 표현의 평균 벡터와 현재 문장의 의도별 벡터의 거리를 의미하는 손실 함수이다. 즉, 원-핫 레이블로 구성된 정답 레이블에서 1에 해당하는 의도별 문장 표현이다. 의도별 문장 표현의 평균벡터인 μ 는 학습 에포크가 끝날 때 갱신된다. μ_i 는 학습데이터로부터 얻은 모든 \mathbf{v}_i 의 평균 벡터인 $\mu = \{\mu_1, \dots, \mu_i, \dots, \mu_C\}$ 에서 i 번째 값이다.

$$\mathcal{L}_{pos} = - \sum_{i \in Y_i=1} (\mu_i - \mathbf{v}_i)^2 \quad (7)$$

이러한 손실 함수를 추가함으로써 의도 분류기는 발화 내에 포함된 의도별 문장 표현을 더 잘 학습하게 된다.

문장 내 포함된 의도의 표현만을 학습하게 된다면 더 많이 등장하는 의도만을 학습하게 될 것이다. 문장 내의 포함되지 않은 의도의 표현도 학습하기 위해서 식 (8)을 추가한다. 식 (8)은 발화 내에 포함되지 않은 의도별 문장 표현에 대한 손실 함수이다. 즉, 정답 레이블에서 0에 해당하는 표현에 대한 손실 함수이다. T 는 발화에 포함된 알고 있는 의도에 대한 확률 밀도를 작게 하기 위해 설정한 마진이다.

$$\mathcal{L}_{neg} = - \sum_{i \in Y_i=0} \max(0, (T - (\mu_i - \mathbf{v}_i)^2)) \quad (8)$$

이는 모든 알려진 의도에 대해서 좋은 의도별 문장 표현을 얻기 위함이다.

모델의 전체 손실 함수는 의도 분류기의 손실 함수와 개수 예측기의 손실 함수, 발화 내 포함된 의도 표현의 손실 함수 그리고 포함되지 않은 의도 표현의 손실 함수의 합으로서 식 (9)로 나타낼 수 있다.

$$\mathcal{L} = \mathcal{L}_{intent} + \mathcal{L}_{num} + \mathcal{L}_{pos} + \mathcal{L}_{neg} \quad (9)$$

3.4 의도 정보를 활용한 다중 레이블 오픈 의도 분류

제안한 모델은 의도 예측값으로 탐지한 의도 개수 K 를 이용하여 다중 레이블 오픈 의도 분류를 진행한다. 모델이 감지한 의도의 개수 K 가 예측된 의도 개수 N' 보다 적을 경우, 오픈 의도가 포함되어 있다고 간주한다.

$$Y'_{ood} = \begin{cases} 1, & \text{if } K < N' \\ 0, & \text{if } K \geq N' \end{cases}$$

이렇게 탐지된 오픈 의도 유무와 다중 레이블 분류의 결과인 의도 예측값을 합쳐서 다중 레이블 오픈 의도 분류가 가능하게 된다.

4. 실험

4.1 실험 설정

제안한 모델의 성능을 평가하기 위하여 MixATIS, MixSNIPS 데이터셋을 사용한다[9]. 사용한 데이터셋은 각각, ATIS, SNIPS 데이터셋의 문장들을 “and”, “and also”, “and then”, “,”를 이용해 한 문장으로 만들어졌다. 데이터셋의 통계 정보는 표 2에 정리해두었다. 본 논문은 오픈 의도 데이터를 만들기 위해서 알고 있는 클래스 개수를 25%, 50%, 75%로 제한하고, 나머지 클래스는 오픈 의도로 할당한다[5]. 제안한 모델은 학습 시, 오픈 의도가 포함되지 않은 문장만 학습한다. 또한, 0부터 9까지 총 10개의 랜덤 시드에 대한 평균 성능을 기록한다.

표 2. 데이터셋 통계 정보

통계 정보	MixATIS	MixSNIPS
학습 데이터셋	13,162	39,776
검증 데이터셋	759	2,198
평가 데이터셋	828	2,199
평균 문장 길이	23.86	19.67
클래스 개수	18	7
의도 개수 최대값	3	3
의도 개수 평균값	2.18	2.01

제안한 모델은 입력 문장의 표현 벡터를 얻기 위해서 특징 추출기를 사용한다. 특징 추출기는 PLM을 사용하는데, 본 논문에서는 bert-base[12], roberta-base[13], electra-base[14], distilbert-base[15]를 사용한다. 표현 벡터의 크기인 d_{model} 은 PLM을 따라서 768로 결정한다.

학습 시, PLM의 학습률은 $1e-5$, 다른 파라미터는 $1e-3$ 으로 설정한다. 데이터의 수가 많지 않기 때문에 배치 크기는 32로 설정한다. 빠른 학습을 위해서 얼리 스탱은 10으로 설정한다.

모델을 평가하기 위한 메트릭으로 의도별 정확도 (Acc), 알고 있는 의도별 정확도 (IND Acc), 오픈 의도 정확도 (OOD Acc), 의도별 Macro F1, 알고 있는 의도별 Macro F1 (IND F1), 오픈 의도의 Macro F1 (OOD F1), Micro-F1, Weighted-F1, Exact Match (EM), AUROC 그리고 의도 개수 예측 정확도 (Accuracy of N)를 이용한다.

4.2 실험 결과

본 논문의 성능은 표 3, 4에서 확인할 수 있다. KR(Known Ratio)는 전체 의도 클래스 중 모델이 알고 있는 클래스의 비율이다. 각 KR에서 가장 좋은 성능은 굵은 글씨로 표시해두었다.

KR이 25% 일 때, 다른 KR보다 평가 지표가 전반적으로 낮다. 이는 학습 데이터에서 오픈 의도에 해당하는 데이터는 제외하여 모델이 학습할 데이터가 충분하지 않기 때문이다. 이러한 현상은 MixSNIPS 데이터셋에서 더욱 심화 되는데, MixSNIPS 의도의 25%는 2개 뿐이라서 대부분의 의도에 해당하는 데이터가 학습 데이터에 포함되지 않기 때문이다. 또한 4개의 특징 추출기 중 BERT가 전반적으로 우수함을 확인할 수 있다.

KR 50%의 MixATIS 결과를 보면 Electra가 가장 높은 Acc와 EM 스코어를 확인할 수 있다. 이는 다른 PLM들 보다 의도의 개수 정확도 (Acc of N)를 10%더 정확히 예측했기 때문에 가능한 성능이다. MixSNIPS의 경우는 distil bert가 가장 좋은 성능을 기록한다.

KR 75%에서, MixATIS는 데이터의 불균형 때문에 오픈 의

표 3. MixATIS 실험 결과

KR	PLM	Acc	IND Acc	OOD Acc	Macro F1	IND F1	OOD F1	Micro F1	Weighted F1	EM	AUROC	Acc of N
25	BERT	76.15	77.43	71.03	38.38	28.29	78.72	60.30	62.76	23.81	0.7307	58.58
	RoBERTa	68.86	73.85	48.87	45.57	41.56	61.61	52.43	57.09	14.54	0.6649	58.57
	Electra	73.93	75.41	67.76	36.95	26.60	78.34	57.90	62.39	20.15	0.7082	59.55
	Distil	74.49	76.53	66.30	43.65	34.93	78.54	59.54	66.07	19.85	0.7244	60.97
50	BERT	91.12	92.72	78.29	54.35	50.26	87.07	75.66	72.97	41.85	0.8441	71.38
	RoBERTa	91.01	93.37	72.18	61.63	58.97	82.87	76.66	76.07	40.09	0.8648	61.11
	Electra	91.41	93.34	76.00	61.40	58.44	85.11	77.40	76.78	43.96	0.8639	82.60
	Distil	91.21	93.29	74.58	59.66	56.61	84.06	76.51	75.26	42.07	0.8552	77.42
75	BERT	94.49	97.30	58.00	78.04	79.64	57.22	81.18	81.61	43.62	0.9014	99.21
	RoBERTa	94.48	97.44	56.09	78.34	80.25	53.47	81.15	81.33	42.78	0.9013	84.89
	Electra	93.73	96.52	57.47	74.47	75.66	59.07	78.19	78.82	37.92	0.8829	97.51
	Distil	93.84	96.65	57.31	74.45	75.73	57.92	78.48	78.59	39.57	0.8796	98.43

표 4. MixSNIPS 실험 결과

KR	PLM	Acc	IND Acc	OOD Acc	Macro F1	IND F1	OOD F1	Micro F1	Weighted F1	EM	AUROC	Acc of N
25	BERT	57.91	63.85	46.02	62.55	64.80	58.03	60.33	60.55	21.70	0.5793	59.06
	RoBERTa	56.81	61.70	47.03	54.30	52.15	58.58	58.37	56.08	21.67	0.5681	46.74
	Electra	51.75	56.56	42.13	57.13	58.35	54.70	56.09	56.12	15.53	0.5180	58.80
	Distil	55.19	60.67	44.21	59.81	61.44	56.57	58.49	58.40	19.14	0.5522	58.88
50	BERT	80.87	87.99	52.37	77.62	82.72	57.21	74.45	73.14	35.18	0.7975	76.13
	RoBERTa	78.29	86.21	46.65	74.80	82.57	43.75	71.48	67.98	33.20	0.7732	69.42
	Electra	79.59	86.04	53.81	77.54	82.52	57.60	74.06	73.12	34.17	0.7938	83.00
	Distil	82.22	89.32	53.80	80.55	85.93	59.04	76.58	75.83	37.66	0.8143	81.85
75	BERT	86.38	92.62	55.14	82.67	89.66	47.72	79.74	78.62	43.51	0.8522	87.64
	RoBERTa	86.45	92.96	53.89	82.32	90.45	41.69	79.75	77.60	44.61	0.8518	83.22
	Electra	86.48	92.30	57.38	83.24	89.55	51.68	80.32	79.56	44.16	0.8593	91.00
	Distil	86.92	92.96	56.68	83.49	89.86	51.68	80.69	79.80	45.30	0.8604	91.55

도 정확도 (OOD Acc)이 감소하는 현상이 일어난다. 또한 두 데이터셋 모두 학습할 데이터 양이 많아지므로 전체 의도 개수를 정확하게 맞추는 현상을 확인할 수 있다.

5. 결론

본 논문은 다중 레이블 오픈 의도 분류 태스크를 정의한다. 다중 레이블 오픈 의도 분류 태스크란 다중 레이블의 의도를 분류하고 동시에 오픈 의도를 분류하는 태스크이다. 해당 태스크를 해결하기 위해 본 논문에서는 의도 정보를 활용한 모델을 제안하고, MixATIS, MixSNIPS 데이터셋에 대해 성능을 평가한다. 실험 결과, MixATIS 75%의도에 대해서 정확도 94.49, F1 97.44를 기록하고 MixSNIPS에 대해서 정확도 86.92, F1 92.96을 기록한다. 후속 연구로는 낮은 Exact Match를 해결하는 연구가 진행될 예정이다.

감사의 글

본 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2021R1F1A1048181)

참고문헌

- [1] Q. Min, L. Qin, Z. Teng, X. Liu, and Y. Zhang, "Dialogue state induction using neural latent variable models," *Proceedings of IJCAI*, C. Bessiere, Ed., pp. 3845–3852, 2020.
- [2] L. Qin, W. Che, Y. Li, M. Ni, and T. Liu, "Dcr-net: A deep co-interactive relation network for joint dialog act recognition and sentiment classification," *Proceedings of AAAI*, pp. 8665–8672, 2020.
- [3] C. Zhang, Y. Li, N. Du, W. Fan, and P. S. Yu, "Joint slot filling and intent detection via capsule neural networks," *Proceedings of ACL*, A. Korhonen, D. R. Traum, and L. Màrquez, Eds., pp. 5259–5267, 2019.
- [4] L. Qin, W. Che, Y. Li, H. Wen, and T. Liu, "A stack-propagation framework with token-level intent detection for spoken language understanding," *Proceedings of EMNLP-IJCNLP*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds., pp. 2078–2087, 2019.
- [5] H. Zhang, H. Xu, and T.-E. Lin, "Deep open intent classification with adaptive decision boundary," *Proceed-*

- ings of the AAAI Conference on Artificial Intelligence*, Vol. 35, No. 16, pp. 14374–14382, 2021.
- [6] Y. Zhou, P. Liu, and X. Qiu, “KNN-contrastive learning for out-of-domain intent classification,” *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5129–5141, May 2022. [Online]. Available: <https://aclanthology.org/2022.acl-long.352>
- [7] Y. Ouyang, Z. Wu, X. Dai, S. Huang, and J. Chen, “Towards multi-label unknown intent detection,” *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 626–635, Oct. 2022. [Online]. Available: <https://aclanthology.org/2022.coling-1.52>
- [8] L. Qin, F. Wei, T. Xie, X. Xu, W. Che, and T. Liu, “GL-GIN: Fast and accurate non-autoregressive model for joint multiple intent detection and slot filling,” *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 178–188, Aug. 2021. [Online]. Available: <https://aclanthology.org/2021.acl-long.15>
- [9] L. Qin, X. Xu, W. Che, and T. Liu, “AGIF: An adaptive graph-interactive framework for joint multiple intent detection and slot filling,” *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1807–1816, Nov. 2020. [Online]. Available: <https://aclanthology.org/2020.findings-emnlp.163>
- [10] L.-M. Zhan, H. Liang, B. Liu, L. Fan, X.-M. Wu, and A. Y. Lam, “Out-of-scope intent detection with self-supervision and discriminative training,” *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3521–3532, Aug. 2021. [Online]. Available: <https://aclanthology.org/2021.acl-long.273>
- [11] L. Cheng, W. Yang, and W. Jia, “A scope sensitive and result attentive model for multi-intent spoken language understanding,” *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, No. 11, pp. 12691–12699, 2023.
- [12] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” *Proceedings of NAACL-HLT*, J. Burstein, C. Doran, and T. Solorio, Eds., pp. 4171–4186, 2019.
- [13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.
- [14] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, “Electra: Pre-training text encoders as discriminators rather than generators,” *arXiv preprint arXiv:2003.10555*, 2020.
- [15] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter,” *arXiv preprint arXiv:1910.01108*, 2019.