제28회 한글 및 한국어 정보처리 학술대회 논문집 (2016년)

격틀과 워드 임베딩을 활용한 유사도 기반 대화 모델링

이호경○, 배경만, 고영중
동아대학교 컴퓨터공학과
{hogay88, kmbae0722, youngjoong.ko}@gmail.com

A Similarity-based Dialogue Modeling with Case Frame and Word Embedding

Hokyung Lee○, Kyoungman Bae, Youngjoong Ko
DongA University, Department of Computer Engineering

요 약

본 논문에서는 격틀과 워드 임베딩을 활용한 유사도 기반 대화 모델링을 제안한다. 기존의 유사도 기반 대화 모델링 방법은 형태소, 형태소 표지, 개체명, 토픽 자질, 핵심단어 등을 대화 말뭉치에서 추출하여 BOW(Bag Of Words) 자질로 사용하였기 때문에 입력된 사용자 발화에 포함된 단어들의 주어, 목적어와 같은 문장성분들의 위치적 역할을 반영할 수 가 없다. 또한, 의미적으로 유사하지만 다른 형태소를 가지는 문장 성분들의 경우 유사도 계산에 반영되지 않는 형태소 불일치 문제가 존재한다. 이러한 문제점을 해결하기 위해서, 위치적 정보를 반영하기 위한 문장성분 기반의 격틀과 형태소 불일치 문제를 해결하기 위한 워드 임베딩을 활용하여 개선된 유사도 기반 대화 모델링을 제안한다. 개선된 유사도 기반 대화 모델링은 MRR 성능 약 92%의 성능을 나타낸다.

주제어: 대화 모델링, 유사도 계산 방법, 워드임베딩, 격틀

1. 서 론

대화 시스템은 입력된 사용자의 발화를 분석하여 적절한 응답을 생성해주는 시스템이다. 일반적으로 대화시스템은 자연어 이해모듈, 대화 관리 모듈 그리고 자연어 생성 모듈 세 가지의 모듈로 구성되어있다. 자연어 이해 모듈은 형태소 분석, 화행분석, 개체명 인식 등의 여러 가지 언어 분석을 통해 사용자 의도를 파악하는 모듈이며, 대화 관리 모듈은 사용자와 시스템간의 모든 대화 과정을 제어하고 관리하며 사용자 입력에 대한 응답을 결정하는 모듈이다. 자연어 생성 모듈은 결정된 시스템 응답을 시스템이 발화할 문장으로 생성해주는 모듈이다. 자연어 이해 모듈과 대화 관리 모듈을 이용하여 대화시스템이 사용자의 의도를 파악하고 사용자 입력에 대한 적절한 응답을 결정하도록 대화 시스템을 전반적으로 모델링하는 것을 대화 모델링이라고 한다.

대화 모델링에 관한 연구는 크게 세 가지 연구 분야로 진행되어왔다. 첫째, 실제 환경에 적용할 수 있도록 시스템을 모델링하는 스크립트 기반 대화 모델링[1]이 제안되었고, 둘째, 복잡하고 다양한 대화 형태를 모델링하기 위한 계획 기반 모델링이 대화 시스템에서 사용되었으며, 셋째, 스크립트 기반 모델과 계획 기반 모델의 영역 전환의 유연성과 대화 시스템의 확장성이 낮은 문제점을 해결 하고자 미리 수집된 대화쌍들로 이루어진 대화 말뭉치를 구축하여 화행정보를 이용한 대화 전이망[2]을 설계하여 대화 모델링에 적용한 코퍼스 기반 대화 모델링을 제안되었다. 코퍼스 기반 대화모델링에서는 사용자 입력 발화를 분석하기 위해서 사용자 입력 발화와 대화 말뭉치의 각 발화들과의 유사도 계산을 통하여 의미적으로 가장 유사한 발화를 찾는다. 일반적인 코퍼스 기반 대화 모델링에서 유사도 계산을 위해 형태소, 형태소 표지, 개체명, 토픽 자질, 핵심단어 등을 대화 말뭉치에서 추출하여 BOW(Bag Of Words) 자질로 사용하였기 때문에 입력된 사용자 발화에 포함된 단어들의 주어, 목적어와 같은 문장성분들의 위치적 역할을 반영할 수 가 없다. 유사도 계산을 할 때 의미적으로 유사하지만 다른 형태소를 가지는 문장 성분들의 경우 유사도 계산에 반영되지 않는 형태소 불일치 문제가 존재한다.

본 논문에서는 위치적 정보를 반영하기 위해 문장성분 기반의 격틀정보를 반영하여 개선된 유사도 기반 대화 모델링 방법을 제안한다. 형태소 불일치 문제를 해결하기 위해 워드 임베딩을 추가적으로 이용하여 형태소 불

이 논문은 2015년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2015R1D1A1A01056907)

일치 문제를 해결한다.

제안한 유사도 기반 대화모델링은 입력된 사용자 발화에서 고정된 격틀을 구성하기 위해서 의존파서[3]를 이용하여 문장성분을 규칙기반으로 추출한다. 추출된 격틀의 문장성분과 연관된 워드 임베딩 벡터를 이용하여 격틀기반 벡터를 구축한 후 입력된 사용자 발화의 격틀 벡터와 대화 말뭉치내의 발화들의 격틀 벡터와의 유사도를 계산한다. 발화 간 유사도 계산은 선행연구에서 활용한 코사인 유사도(cosine similarity) 방법을 사용하여 유사도를 계산한다.

격틀과 워드임베딩을 활용한 유사도 기반 대화모델링의 성능은 약 92%의 MRR(Mean reciprocal rank)을 나타냈으며, 이러한 결과를 통하여 개선된 유사도 방법이 효과적인 대화 모델링을 수행하였다고 판단할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구들을 소개하며, 3장에서는 유사도 방법에 대해 상세히 기술한다. 4장에서는 성능을 비교하며, 5장에서는 결론 및 향후 연구 계획을 기술한다.

2. 관련 연구

대화 모델링에 관한 연구에서 [4]는 사용자와 시스템의 주도가 혼합된 혼합주도 대화에 적합한 모델을 제안하였고, 시스템의 응답 성향을 조절할 수 있는 방법을 포함 시켰다. [5]는 한국어 대화에서 나타내는 생략 현상을 고려하여 한국어에 적합한 대화 모델링을 제안하였으며, [6]은 대화 전략과 담화 스택을 이용한 대화 모델링을 제안하였다.

유사도 계산을 이용한 대화 모델링 방법에 관한 연구로 [7]은 두 개의 짧은 문장 사이의 의미적 유사도를 측정하는 방법인 LSA(Latent Semantic Analysis)방법을 이용하여 대화 시스템에 적용하였다. [8]은 사용자 입력 문장과 말뭉치 발화들과의 유사도 계산을 위해 형태소, 형태소 표지, 개체명, 개체명 표지 그리고 토픽 자질을 추출하고 코사인 유사도를 통해 유사도를 측정하였으며, [9]는 핵심 단어(keyword)를 고려한 코사인 유사도 계산 방법을 사용하였다.[10]은 TV Guide 대화 시스템에서 영역별 행동을 나눈 뒤 영역별 단어 가중치 방법을 고려한 코사인 유사도를 이용하여 유사도를 측정하였다. 따라서 본 논문에서는 선행연구의 제안방법들을 참조하고 행위 목적과 행위를 활용하여 개선된 유사도 방법을 제안하여 성능을 개선한다.

대화 모델링에서 딥러닝을 적용하는 방법 중 하나는 워드 임베딩을 활용하는 것이다. 워드 임베딩(Word Embedding)은 신경망(Neural Network)을 이용하여 단어들의 의미를 특정 차원의 벡터 값으로 계산하는 비지도 학습 방법(Unsupervised Learning)이다. [11]이 제안하는 워드 임베딩 학습 모델에는 CBOW(Continuous Bag-of-Words)와 Skip-Gram 모델이 존재 있다.CBOW 모델은 은닉층(Hidden Layer)를 제거하였기 때문에 학습 시간을 비약적으로 단축하였다.

[12]는 대화 시스템에서 각 발화들의 의도를 파악하기 위해서 사전 학습(Pre-train)된 워드 임베딩을 활용하였다. [13]은 대화 시스템의 semantic slot-filling영역에서 다양하게 사전학습된 워드임베딩을 활용하여 성능을 개선하였다. [14]는 사용자의 의도 식별 및 목적 예측을 위해 사전학습된 워드 임베딩을 활용하여 변형된 RNN(Recurrent Neural Network)에 적용하였다.

3. 제안 방법

이 장에서는 tfidf 기반 유사도 계산 방법과 본 논문에서 제안하는 격틀과 워드임베딩을 활용한 유사도 방법에 대해 상세히 설명한다. 그림 1은 사용자 입력 발화가 들어왔을 때 2단계 유사도 계산 과정을 나타낸다.

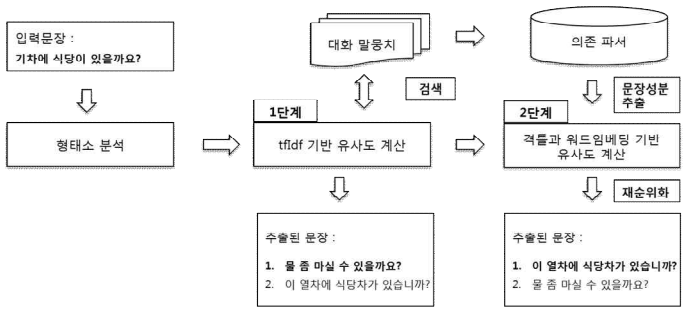

그림 1. 2단계 유사도 계산 과정

먼저, 입력된 사용자 발화가 들어오면 형태소 분석 과정을 거친다. 형태소 분석이 완료되면, 입력된 사용자 발화의 모든 형태소의 tfidf 가중치와 말뭉치내의 각 발화들의 모든 형태소 tfidf 가중치와 유사도를 계산한다. 이렇게 계산된 유사도 결과를 기준으로 말뭉치내의 발화들을 순위화한다. 순위화된 발화들 중 특정 임계값(0.2) 이상인 후보발화들을 추출한다.(임계값은 실험을 통하여 0.2로 결정하였다.) 이렇게 추출된 후보발화들과 입력된 사용자 발화에 대해서 의존파서를 이용하여 의존관계를 추출한 뒤 규칙기반으로 문장성분들을 추출한다. 그리하여 추출된 문장성분들을 가지고 격틀을 생성하고, 각 문장성분들의 워드임베딩 결과를 가지고 격틀을 확장한다. 확장된 각각의 격틀을 가지고 유사도를 계산한 뒤 추출된 후보발화들의 tfidf 가중치에 새롭게 계산된 유사도를 결합하여 후보발화들을 재순위화 한다.

3.1. tfidf 기반 유사도 계산 방법

사용자 입력 발화가 입력될 때, 입력된 사용자 발화와 대화 말뭉치내의 발화들의 모든 형태소에 대한

tfidf 가중치를 계산 한 뒤 tfidf 기반 코사인 유사도를 계산한다. 식 (1)은 tfidf 가중치에 대한 설명이고, 식 (2)는 코사인 유사도에 대한 설명이다.

$$tfidf\_Weight_{t,u} = \log(1+tf_{t,u}) \times \log(N/uf_t) \qquad (1)$$

- t는 단어
- u는 발화
- $tf_{t,u}$는 단어가 해당 발화에서 나타난 횟수
- N은 대화말뭉치내의 전체 발화 수
- uf_t 는 대화 말뭉치에서 해당 단어가 나타난 발화들의 수
- 식(1)의 가중치는 식(2)의 iu_vec, cu_vec 벡터값으로 사용

$$Sim(IU,\ CU) = \frac{\sum_{i=1}^{n} iu\_vec_i \times cu\_vec_i}{\sqrt{\sum_{i=1}^{n} iu\_vec_i^{\,2}}\ \sqrt{\sum_{i=1}^{n} cu\_vec_i^{\,2}}} \qquad (2)$$

- IU는 입력 발화
- CU는 대화 말뭉치내의 발화
- iu_vec는 입력 발화의 벡터
- cu_vec는 대화 말뭉치내의 발화들의 벡터

이렇게 계산 된 유사도를 가지고 대화 말뭉치 내에서 발화들을 순위화한다. 순위화된 후보 발화들 중 임계값 0.2 이상인 후보 발화들만 추출한다.

3.2. 워드임베딩을 활용한 격틀 기반 유사도 계산 방법

두 번째 유사도를 계산하기 전에 먼저 입력된 사용자 발화에 포함된 단어들의 위치적 역할을 반영하기 위해 의존파서를 통해서 입력된 사용자 발화의 의존 그래프가 형성되면 의존관계명이 SBJ일 경우 주어, OBJ일 경우 목적어 그리고 ROOT일 경우 동사로 문장성분을 규칙기반으로 추출한다.[15](주어와 목적어를 제외한 모든 성분을 보어로 간주하였다.) 그림 2는 문장성분 추출 과정이다.

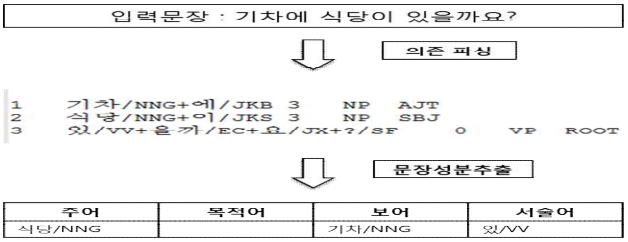

그림 2. 문장성분 추출 과정

추출된 문장성분으로 격틀을 구성한다. 격틀은 주어, 목적어, 보어, 동사와 같은 문장성분으로 구성되어 있으며 워드 임베딩 결과의 64차원 Lookup Table을 활용하여 각 문장성분을 1*64차원으로 각각 구성한다. 그림 3은 워드 임베딩을 이용한 격틀 생성과정이다. (그림 2와 그림 3의 과정을 대화 말뭉치에도 동일하게 적용하였다.)

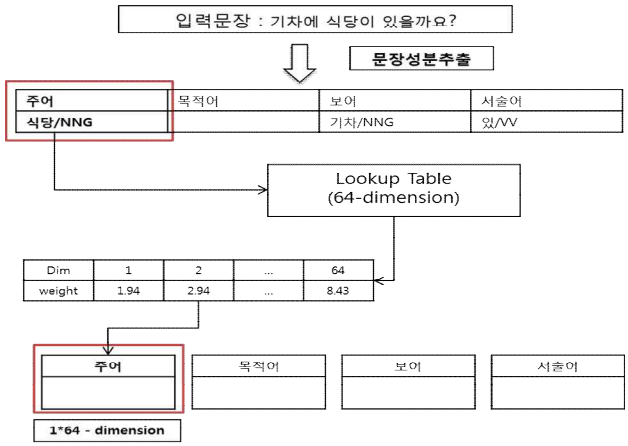

그림 3. 워드임베딩을 이용한 격틀 생성 과정

워드 임베딩은 형태소가 불일치하더라도 대화 말뭉치내에 존재하는 형태소간의 관계를 벡터 공간에 나타내기 때문에, 의미가 유사한 형태소일수록 유사도가 높게 나타남을 알 수 있다.(워드 임베딩은 3GB 뉴스 말뭉치를 활용하여 Word2Vec의 CBOW 모델 사용[16])

두 번째 유사도 계산 방법은 사용자 입력 발화와 앞서 추출된 후보 발화들의 문장성분 격틀에 활용하여 유사도를 계산한다.

식 (3)의 각각의 벡터를 식 (2)의 iu_vec, cu_vec에 적용하여 코사인 유사도를 계산한다.

$$Sim_{cfw}(IU, CU) = Sim(vec_{s,iu},\ vec_{s,cu}) + Sim(vec_{o,iu},\ vec_{o,cu}) + Sim(vec_{v,iu},\ vec_{v,cu}) + Sim(vec_{p,iu},\ vec_{p,cu}) \qquad (3)$$

- vec_s : 1*64차원의 주어 성분 벡터
- vec_o : 1*64차원의 목적어 성분 벡터
- vec_c : 1*64차원의 보어 성분 벡터
- vec_v : 1*64차원의 서술어 성분 벡터
- 각 성분의 벡터값은 64차원의 워드임베딩 결과

그런 다음 첫 번째 단계에 계산해놓은 tfidf 기반 코사인 유사도(Sim_{tfidf})와 두 번째 단계에서 계산한 격

틀 기반 코사인 유사도(Sim_{cfw})와의 선형결합($Sim_{tfidfNcfw}$)을 수행하여 후보발화들을 식 (4)를 통해서 재순위화 한다.

$$Sim_{tfIdfNcmf} = (1-\alpha) \cdot Sim_{tfIdf} + \alpha \cdot Sim_{cmf} \quad (4)$$

- α는 실험을 통하여 0.1로 결정

유사도는 0~1사이의 값을 나타나고, 유사도는 1에 가까울수록 유사한 발화라고 판단한다.

그림 4는 입력된 사용자 발화에 대해 2단계 유사도 법을 적용했을 때의 결과를 나타낸다.

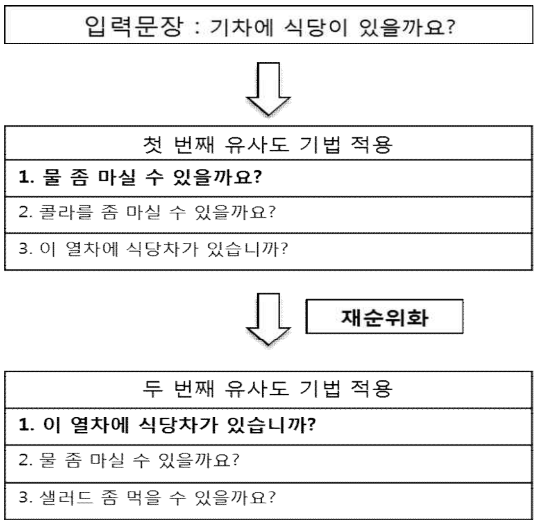

그림 4. 2단계 유사도 계산 결과

4. 실험

이 장에서는 개발 데이터를 통해 tfidf 기반 유사도 방법의 성능평가에 따른 임계값 설정에 대해 설명하고, 본 논문에서 제안된 유사도 방법에 대한 성능에 평가에 대해 설명한다.

4.1. 대화 말뭉치

본 논문에 사용된 대화 말뭉치는 대학생 4명이 각종 여행서적 및 개인의 여행 경험을 바탕으로 수집한 여행 영역의 대화 말뭉치이다. 총 1,061개의 발화로 구성 되어 있다.(사용자 발화 : 586개, 시스템 발화 : 475개) 표 1은 각 여행 영역에 대한 발화의 수를 나타낸다.

표 1. 각 여행 영역의 발화의 수

영역		발화(개)
공항	기내	143
	수하물	67
	면세점(상점)	79
	탑승수속	61
	보안 검색대 /입국심사대	51
	세관/환전소	23
식당		107
숙박		258
교통	길거리에서	73
	렌터카	61
	택시	38
	버스	38
	열차/지하철	62
발화의 총 개수		1061

4.2. 유사도 방법에 대한 임계값 설정

유사도 계산 시 임계값 결정과 유사도 방법의 성능 평가를 위해 대학생 4명에게 여행 영역의 세부 영역 4개(공항, 교통, 식당, 숙박)의 여행 영역에서 일어날 수 있는 상황을 제시하여 각각 100개, 150개의 발화를 수집하였다. 100개의 발화는 개발 데이터로 사용하였고, 150개의 발화는 평가 데이터로 사용하였다.

임계값 결정은 유사도 성능에 영향을 미치기 때문에 유사도 기반 대화모델링에서는 중요한 역할을 한다. 따라서, 그림 5는 개발 데이터를 사용하여 각 임계값에 따른 tfidf 기반 유사도 방법의 MRR 성능 평가이다.

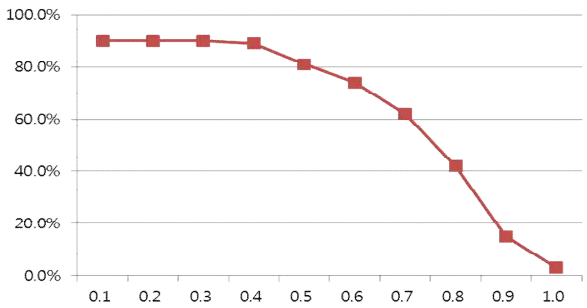

그림 5. 임계값에 따른 tfidf 기반 유사도의 MRR

임계값 0.1 ~ 0.3의 경우 모두 MRR 성능이 동일하기 때문에 본 논문에서는 0.2로 임계값을 결정하였다.

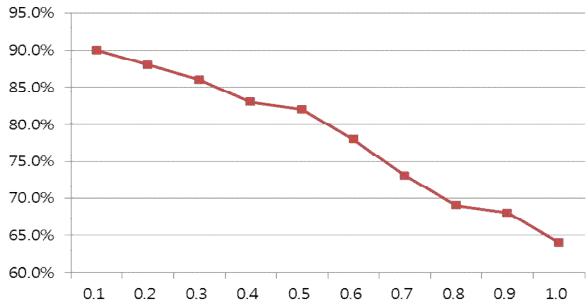

그림 6. 선형 결합 비율에 따른 tfidf 기반과 워드임베딩을 활용한 격틀기반을 결합한 유사도의 MRR

그림 6은 개발 데이터를 사용하여 선형 결합 비율에 따른 tfidf 기반과 워드임베딩을 활용한 격틀기반의 유사도 방법을 선형 결합한 유사도 방법의 MRR 성능 평가이다. 본 논문에서는 선형 결합 비율이 0.1일 때 MRR 성능이 가장 높기 때문에 선형 결합 비율을 0.1로 결정하였다.

4.3. 유사도 방법에 대한 성능평가

유사도 방법의 성능 평가는 식 (5)를 이용하여 유사도에 따라 순위화된 발화들의 순위 1,2,3만을 고려하여 성능을 평가 하였다.(4위 이하는 성능에 반영하지 않았다.)

$$MRR = \frac{1}{|U|}\sum_{i=1}^{|U|}\frac{1}{rank_i} \qquad (5)$$

- U : 평가 데이터의 개수
- $rank_j$: 유사도 값으로 순위화된 발화들의 순위

표 2. 유사도 방법 실험 결과 (순위1)

Method	MRR
tfidf	0.906
word embedding	0.447
tfidf + word embedding	**0.926**

표 2는 tfidf는 tfidf 기반 코사인 유사도 방법의 MRR 성능이며, word embedding는 워드 임베딩을 활용한 격틀 기반 코사인 유사도 방법의 MRR 성능, tfidf + word embedding은 2가지 방법을 선형결합한 코사인 유사도 방법의 MRR 성능이다.(순위는 1위만을 고려하였다.) 따라서 2가지 방법을 선형 결합(임계값 : 0.2, alpha : 0.1)한 유사도 방법이 tfidf 기반 유사도 방법을 단독으로 이용한 것보다 MRR 성능 약 2%를 개선하였다.

표 3. 유사도 방법 실험 결과 (순위2)

Method	MRR
tfidf	0.927
word embedding	0.4667
tfidf + word embedding	**0.937**

표 4. 유사도 방법 실험 결과 (순위3)

Method	MRR
tfidf	0.927
word embedding	0.489
tfidf + word embedding	**0.939**

표 3과, 표 4는 순위를 2,3위까지 고려한 추가 실험으로, 각 유사도 방법의 MRR 성능을 평가 해본 결과 전반적으로 성능이 모두 개선되었다. 그중에서도 순위를 3위까지 고려할 때 2가지 유사도 방법을 선형결합한 유사도 방법이 순위 1위만을 고려할 때 보다 MRR 성능 약 1%가 개선되었다.

그림 7은 순위에 따른 2가지 유사도 방법의 MRR 성능 비교 이다.

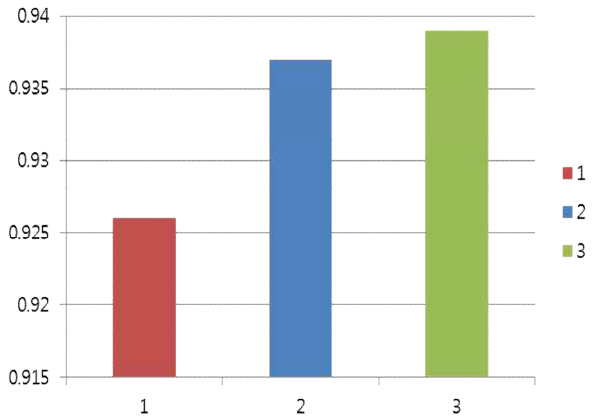

그림 7. 순위에 따른 2가지 유사도 방법의 MRR

5. 결론

본 논문에서는 tfidf 기반 유사도 방법에 워드임베딩을 활용한 격틀 기반 유사도 방법을 선형 결합하여 새로운 유사도 기반 대화모델링을 제안하여 MRR 성능 약 92% 성능을 얻었다. 타 논문과의 직접적인 성능 비교는 할 수 없지만, 기존의 대화 시스템 연구에서 활용한 tfidf 방법 보다 MRR 약 2%의 성능 개선 효과를 얻을 수 있었다.

향후 연구로는, 효과적인 대화 모델링을 위해 유사도 방법의 개선뿐만 아니라 딥러닝을 활용한 새로운 대화 모델링 방법을 제안하여 더욱더 효과적인 대화 모델링을 진행할 계획이며, 일반적인 대화에 존재하는 복문, 중문을 처리할 수 있도록 연구를 진행할 계획이다.

참고문헌

[1] R.S. Russell, "Language use, personality and true conversationalinterfaces", Project Report, AI and CS, University of Edinburgh, 2002.

[2] Sangwoo Kang, Hongmin Park, Youngjoong Ko and Jungyun Seo, "The Corpus-based Dialogue System Using a Dialogue Transition Network and a Similarity Measure Method", *Proceedings of the 20th Annual Conference on Human and Cognitive Language Technology (HCLT 2008), pp. 161-165,* October 2008.

[3] Dependency Parser, [Online]. Available: semanticweb.kaist.ac.kr/home/index.php/KoreanParser.

[4] Young-Hwan Cho, "Mixed-initiative response generation model in a task-execution dialog system", *Ph.D Thesis, Korea Advanced Institute of Science and Technology (KAIST),* August 1997.

[5] Cheol-jin Yoon, Jung-yun Seo, "Plan-based Ellipsis Resolution for Utterance in Noun-Phrase-Form in Restricted Domain Dialogues", *Korean journal of cognitive science, Vol. 11, No. 1,* March 2000.

[6] Sangwoo Kang, Youngjoong Ko, Jungyun Seo, "Using Plan Recognition and a Discourse Stack for Effective Response Generation in a Dialogue System", *Korean Journal of Cognitive Science, Vol. 19, No. 2, pp. 107-124,* June 2008.

[7] J.O' Shea, Z.Bandar, K.Crockett, D.McLean, "A Comparative Study of Two Short Text Semantic Similarity Measures", *Proceeding of theAgentand Multi-AgentSystem : Technologiesand Applications, Second KES International Symposium, Vol. 4953, pp.172-181,* 2008.

[8] Sangwood Kang, Youngjoong Ko and Jungyun Seo, "A Dialogue management system using a corpus-based framework and a dynamic dialogue transition model", *AI Communications, Vol. 26, No. 10, pp. 145-159,* April 2013.

[9] Jaewon Hwang, Yonghyun Park and Youngjoong Ko. "The Effective Dialogue Modeling for Intelligent Companion Robot". *Proceedings of Fall Conference on Korean Institute of Information Scientists and Engineers (KIISE 2010), Vol. 37, No. 2(A), pp. 44-45,* November 2010.

[10] Yonghyun Park, "The Dialogue TV Guide System Using Word Weighting Method for Domain", *Master's Thesis, University of Dong-A,* February 2012.

[11] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient Estimation of Word Representations in Vector Space", *In Proceedings of Workshop at International Conference on Learning Representations(ICLR 2013),* May 2013.

[12] G. Forgues, J. Pineau, J. Larcheveque, R. Tremblay. "Bootstrapping Dialog Systems with Word Embeddings", *Workshop on Modern Machine Learning and Natural Language Processing (NIPS 2014),* December 2014.

[13] Xiaohao Yang, Zhenfeng Chen, Jia Liu. "Word embeddings: A semi-supervised learning method for slot-filling in spoken dialog systems", *The 9th International Symposium on Chinese Spoken Language Processing(ISCSLP 2014), September 2014.*

[14] Y. Luan, S. Watanabe, and B. Harsham, "Efficient learning for spoken language understanding tasks with wordembedding based pre-training," *in Proceedings of the Interspeech (InterSpeech 2015),* September 2015.

[15] Kyungman Bae, "Question Classification and Retrieval Based on the Word Embedding for the cQA Service", Dong-A University, August 2016.

[16] word2vector, [Online]. Available: https://code.google.com/archive/p/word2vec/