단어 유사도를 이용한 뉴스 토픽 추출

News Topic Extraction based on Word Similarity

저자 (Authors)
김동욱, 이수원
Dongxu Jin, Soowon Lee

출처 (Source)
정보과학회논문지 44(11), 2017.11, 1138-1148(11 pages)
Journal of KIISE 44(11), 2017.11, 1138-1148(11 pages)

발행처 (Publisher)
한국정보과학회
KOREA INFORMATION SCIENCE SOCIETY

URL
http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE07261640

APA Style
김동욱, 이수원 (2017). 단어 유사도를 이용한 뉴스 토픽 추출. 정보과학회논문지, 44(11), 1138-1148

ISSN 2383-630X(Print) / ISSN 2383-6296(Online)
Journal of KIISE, Vol. 44, No. 11, pp. 1138-1148, 2017. 11
https://doi.org/10.5626/JOK.2017.44.11.1138

단어 유사도를 이용한 뉴스 토픽 추출
(News Topic Extraction based on Word Similarity)

김 동 욱 † 이 수 원 ††
(Dongxu Jin) (Soowon Lee)

요 약 토픽 추출은 문서 집합으로부터 그 문서 집합을 대표하는 토픽을 자동 추출하는 기술이며 자연어 처리의 중요한 연구 분야이다. 대표적인 토픽 추출 방법으로는 잠재 디리클레 할당과 단어 군집화 기반 토픽 추출방법이 있다. 그러나 이러한 방법의 문제점으로는 토픽 중복 문제와 토픽 혼재 문제가 있다. 토픽 중복 문제는 특정 토픽이 여러 개의 토픽으로 추출되는 문제이며, 토픽 혼재 문제는 추출된 하나의 토픽 내에 여러 토픽이 혼재되어 있는 문제이다. 이러한 문제를 해결하기 위하여 본 연구에서는 토픽 중복 문제에 대해 강건한 잠재 디리클레 할당으로 토픽을 추출하고 단어 간 유사도를 이용하여 토픽 분리 및 토픽 병합의 단계를 거쳐 최종적으로 토픽을 보정하는 방법을 제안한다. 실험 결과 제안 방법이 잠재 디리클레 할당 방법에 비해 좋은 성능을 보였다.
키워드: 텍스트 마이닝, 토픽 추출, LDA, 기계 학습

Abstract Topic extraction is a technology that automatically extracts a set of topics from a set of documents, and this has been a major research topic in the area of natural language processing. Representative topic extraction methods include Latent Dirichlet Allocation (LDA) and word clustering-based methods. However, there are problems with these methods, such as repeated topics and mixed topics. The problem of repeated topics is one in which a specific topic is extracted as several topics, while the problem of mixed topic is one in which several topics are mixed in a single extracted topic. To solve these problems, this study proposes a method to extract topics using an LDA that is robust against the problem of repeated topic, going through the steps of separating and merging the topics using the similarity between words to correct the extracted topics. As a result of the experiment, the proposed method showed better performance than the conventional LDA method.
Keywords: text mining, topic extraction, LDA, machine learning

· 이 논문은 2015년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2015R1D1A1A01056622)

† 정 회 원 : 숭실대학교 대학원 컴퓨터학과
jinesse.me@gmail.com
†† 종신회원 : 숭실대학교 소프트웨어학부 교수(Soongsil Univ.)
swlee@ssu.ac.kr
(Corresponding author임)

논문접수 : 2017년 1월 26일
(Received 26 January 2017)
논문수정 : 2017년 8월 6일
(Revised 6 August 2017)
심사완료 : 2017년 8월 8일
(Accepted 8 August 2017)

정보과학회논문지 제44권 제11호(2017. 11)

1. 서 론

토픽 추출(Topic Extraction)은 문서 집합으로부터 그 문서 집합을 대표하는 토픽을 자동 추출하는 기술이며 자연어 처리의 중요한 연구 분야이다[1]. 토픽 추출을 위한 방법으로 LSA[2] 등 벡터 기반의 모델은 단어 벡터를 이용해 문서를 다차원으로 표현하는 것에 비해, 토픽 모델은 단어의 분포가 특정 토픽에 따라 다르다는 것을 기반으로 문서에 포함된 토픽을 확률 분포로 표현한다. 토픽 모델을 사용하게 되면 문서를 저차원으로 표현할 수 있고 또한 잠재적인 토픽을 추출할 수 있다.

대표적인 토픽 모델인 잠재 디리클레 할당(Latent Dirichlet Allocation, LDA)은 토픽을 문서에 할당하는 확률 모형이다[3]. 주어진 문서로부터 주제별 단어의 분포를 추정하고, 주어진 문서에서 발견된 단어의 분포를 분석하는 것으로 해당 문서가 어떤 주제들을 다루고 있

는지를 관측할 수 있다.

잠재 디리클레 할당은 간결하고 실용적인 토픽모델로 여러 연구나 제품에 많이 응용되고 있다. 중국의 텐센트(Tencent)에서는 잠재 디리클레 할당을 이용한 대규모 잠재 토픽 추출 프로젝트 Peacock을 상용화하였다[4]. Peacock에서는 10억×1억 크기의 행렬을 분해하여 계산하는 병렬계산 방법을 통하여 10만~100만개의 토픽을 학습한다. 학습된 토픽은 텍스트 단어 의미 추출, 추천 시스템, 사용자의 취향 파악, 광고 추천 등의 영역에 사용된다.

토픽추출에 있어서 LDA외에 단어 군집화 방법을 이용한 토픽 추출방법도 있다. [5]에서는 지역별 뉴스를 이용하여 지역별 토픽을 단어 군집화 방법을 통해 추출하는 방법을 제안하였다.

그러나 LDA 또는 단어 군집화 방법을 사용하면 토픽 중복 문제와 토픽 혼재 문제가 발생할 수 있다. 토픽 중복은 특정 토픽이 여러 개의 토픽으로 추출되는 문제이며, 토픽 혼재는 추출된 하나의 토픽 내에 여러 토픽이 혼재되어 있는 문제이다. 그림 1에서 (1)과 (2)는 토픽 중복의 예이고, (3)은 토픽 혼재의 예이다.

이러한 문제를 해결하기 위하여 본 논문에서는 기존의 잠재디리클레 할당으로 토픽을 추출한 다음, 단어 간 유사도를 이용한 토픽 분리 및 토픽 병합의 방법을 통하여 추출된 토픽을 보정하는 방법을 새로이 제안한다.

본 논문의 구성은 다음과 같다. 1장의 서론에 이어 2장에서는 관련연구들을 소개하며, 3장에서는 제안하는 토픽 추출 모듈 및 토픽 보정 모듈을 설명하고, 4장에서는 제안 방법에 대한 실험 및 평가결과를 제시하며 5장에서는 결론 및 향후연구에 대해 기술한다.

2. 관련 연구

LDA는 이산 데이터에 대한 확률 모델이며, 기존의 정보 검색분야에서 연구되어 온 모델들의 발전된 형태이다[6]. TF-IDF를 필두로 하여 잠재 의미 인덱싱(Latent Semantic Indexing, LSI)[7], 확률 잠재 의미 분석(Probabilistic Latent Semantic Analysis, PLSA)[8] 등을 거쳐 LDA로 발전하였다. 확률 잠재 의미 분석은 확률 잠재 의미 인덱싱(Probabilistic Latent Semantic Indexing, PLSI)[9]이라고도 한다.

LDA 문서 생성의 일반적인 과정은 다음과 같다.

1. $Dirichlet\ Distribution(\alpha)$에서 문장 d_i의 토픽 분포 θ_i을 샘플링 한다.
2. 토픽의 $Multinominal\ Distribution\ \theta_i$로부터 문장 d_i의 j번째 단어의 토픽 $z_{i,j}$을 샘플링 한다.
3. $Dirichlet\ Distribution(\beta)$에서 토픽 $z_{i,j}$에 해당되는 단어 분포 $\Phi_{z_{ij}}$을 샘플링 한다.
4. 단어의 $Multinominal\ Distribution\ \Phi_{z_{ij}}$에서 최종 단어 $w_{i,j}$을 샘플링 한다.

[5]의 연구에서는 “같은 맥락에서 발생하는 단어는 유사한 의미를 가지는 경향이 있다”고 가정하고, 두 단어

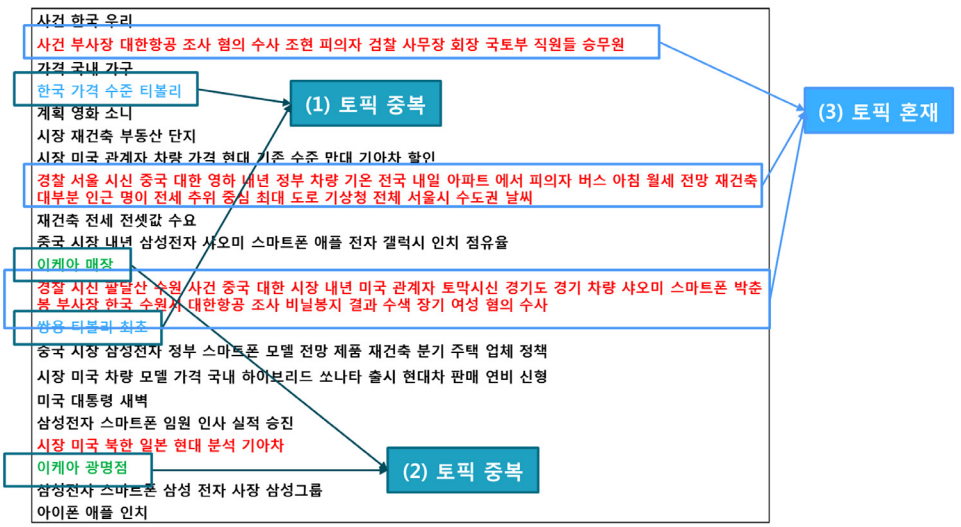


그림 1 토픽 중복 문제 및 토픽 혼재 문제 예시
Fig. 1 Example of Topic Duplication and Topic Mix

의 연관성을 등장 확률로 측정하기 위하여 식(1)을 정의하였다.

$$PMI(word_1, word_2) = \log_2 \frac{P(word_1 \cap word_2)}{P(word_1)P(word_2)} \quad (1)$$

식 (1)에서 $P(word_i)$는 $word_i$가 출현할 확률이며, $P(word_1 \cap word_2)$는 $word_1$과 $word_2$가 하나의 문장에 동시 출현할 확률을 의미한다. 즉, PMI는 두 단어가 하나의 문장에 동시 출현할 확률에 각각 출현한 확률의 곱한 것을 나눈 값으로, 두 단어의 연관성이 높을수록 PMI값이 높다.

토픽 모델의 파라미터 선정의 정확도는 Perplexity[11]로 평가할 수 있으며, 식 (2)와 같이 계산된다. LDA토픽 모델에서는 Perplexity가 Likelihood의 수치를 의미한다.

$$Perplexity(D_{test}) = \exp\left\{\frac{-\sum_d \log(P(w_d))}{\sum_d N_d}\right\} \quad (2)$$

새로운 파라미터의 토픽모델이 구성되었을 때 LDA은 테스트 집합에서의 Perplexity와 비교하여 보다 작은 값을 가지는 모델의 성능이 향상된 토픽 모델이다.

[5]에서는 토픽을 단어 집합으로 정의하고, 지역별 토픽을 추출하기 위한 단어 군집화 방법을 제안하였다. 제안 방법에서는 지역별 토픽 추출을 위하여 수집된 지역별 뉴스 문서에서 추출된 단어 집합으로부터 지역별 문서 집합을 대표하는 핵심어들을 추출한다. 추출된 핵심어 집합에서 단어 군집화를 위한 시드를 선정하고, 선정된 시드를 중심으로 단어 군집화의 첫 번째 단계인 초기 군집화를 수행한다. 초기 군집화는 시드와 동일한 문장에서 출현한 핵심어 사이의 연관성 값이 일정 임계치 이상이면 하나의 군집으로 군집화하는 단계이다. 생성된 초기 군집 중에는 단어의 구성이 유사한 군집이 존재할 수 있다. 따라서 군집 병합 단계를 통해 유사한 군집을 찾아 병합한다. 유사한 군집이 모두 병합되면 최종적으로 지역별 토픽이 추출된다.

3. 제안 방법

3.1 개요

본 연구에서는 토픽 추출방법에서 토픽 혼재 문제와 토픽 중복 문제를 해결하기 위해 추출된 토픽 내에서 단어 간 유사도를 이용하여 추출된 토픽을 보정하는 방법을 제안한다. 그림 2는 본 연구에서 제안한 방법의 구조도이다. 제안 방법은 전처리(Preprocessing) 과정을 거친 문서 집합을 입력으로 하여 LDA 기반 토픽 추출(Topic Extraction) 모듈을 수행한 후 토픽 보정(Topic Correction) 모듈을 통하여 토픽을 생성한다.

토픽 보정 모듈에서는 단어 간 PMI를 계산하여 단어

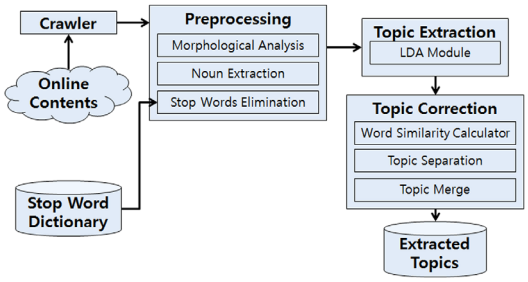


그림 2 단어 유사도를 이용한 뉴스 토픽 추출 구조도
Fig. 2 Structure of News Topic Extraction using Word Similarity

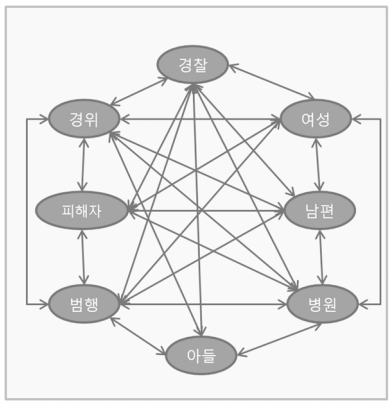


그림 3 Topic Clique (TC)
Fig. 3 Topic Clique (TC)

간 유사도(Word Similarity) Matrix를 생성한다. 이때 $PMI \leq 0$이면 두 단어는 연관성이 없는 것으로 볼 수 있다. 토픽 분리(Topic Seperation) 과정에서는 이 Matrix를 이용하여 각 토픽의 Topic Clique(TC)를 생성한다.

본 연구에서는 토픽 내 단어를 정점으로 하고, 단어 간 PMI가 0보다 큰 값을 간선의 Weight로 하는 완전 부분 그래프(Complete Subgraph)를 TC로 정의한다. LDA로 추출된 하나의 토픽에서 여러 개의 TC가 생성될 수 있다.

그림 3은 LDA로 추출된 토픽에서 생성된 TC의 예이다. 그림 2의 토픽 병합(Topic Merge) 과정에서는 토픽 분리 과정에서 생성된 TC간 거리를 구하고 한정 범위 내에 있는 TC간 병합을 통하여 최종 토픽을 추출한다.

3.2 전처리

본 연구에서는 토픽을 명사의 집합으로 정의한다. 이에 따라 텍스트 전처리는 형태소 분석, 명사 추출, 불용어 제거의 과정으로 구성된다. 먼저 크롤러를 이용해 수집된 뉴스 데이터 중 중복뉴스를 제거하고 전처리 모듈

의 형태소 분석을 통하여 문장에서 명사를 추출한다. 명사를 추출하기 위해서는 형태소 분석기를 사용하여 품사를 태깅하고 그 중에서 명사에 해당되는 토큰(Token)만 유지하고 다른 부분은 제거한다. 추출된 명사 중 불용어 사전을 기반으로 단어를 필터링한다. 전처리된 문서 집합을 제안한 유사도 기반 토픽 추출기에 입력한다.

3.3 토픽 추출

본 연구에서는 LDA를 이용하여 토픽을 추출한다. 표 1은 LDA로 추출된 토픽의 예시이다. 추출된 각 토픽에서 파랑색은 정답토픽 1에 해당되고 노란색은 정답토픽 2에 해당되며, 빨간색은 정답 토픽과 관련이 없는 잘못 추출된 단어이다. Topic07은 두가지 정답토픽이 혼재되어 추출되어 있으며, Topic03, Topic04를 제외한 토픽에서는 오류 단어들이 존재한다.

3.4 토픽 보정

3.4.1 토픽 내 단어 유사도 계산

LDA는 토픽 내 단어의 출현 확률분포를 이용한 기법으로 토픽 내 단어 간 유사도는 계산하지 않는다. LDA와 같은 출현빈도를 이용한 토픽추출 방법을 사용하면 토픽 혼재 문제가 있을 수 있고 사용자가 요구하는 토픽이 추출되지 않을 수 있다. 본 연구에서 제안하는 방법은 지정된 문서의 토픽 내 단어 간 유사도를 사용하여 토픽 혼재 문제를 해결한다.

토픽 내 단어 간 유사도는 PMI를 사용한다. 식 (3)은 본 연구에 사용된 PMI값의 정의이다.

$$\begin{aligned} &PMI(A,B)=0 \ : \ P(A\cap B)=P(A)\times P(B) \\ &\text{// A와 B는 독립이다.} \\ &PMI(A,B)<0 \ : \ P(A\cap B)<P(A)\times P(B) \\ &\text{// A와 B는 음의 관계를 가진다.} \\ &PMI(A,B)\approx -\infty \ : \ P(A\cap B)=0 \\ &\text{// A와 B는 상호배타적이다.} \end{aligned} \tag{3}$$

표 2는 LDA기법으로 추출된 토픽 내 단어 간 유사도를 계산한 Matrix의 예시이다. 붉은 색은 토픽 내 단어 간 $P(A\cap B)=0$을 의미하고, 분홍색은 $PMI(A,B)<0$ 토픽 내 단어 간 음의 관계를 의미한다. 계산된 토픽 내 단어 간 유사도는 총 190개(Matrix의 상위 절반을 의미)이며, $PMI=-\infty$인 경우는 7개, $PMI\neq -\infty, PMI\leq 0$인 경우는 38개이다.

3.4.2 토픽 분리

LDA로 추출된 토픽 내 단어를 빈도에 따라 정렬한다(그림 4). 토픽 분리 과정에서는 토픽 내 단어 빈도와 단어 간의 PMI값을 이용하여 TC를 생성한다.

먼저 토픽 내 단어의 기준 단어를 변경하면서 기준 단어를 포함한 TC를 생성한다. 그림 5는 추출된 토픽 내 단어 "경찰"을 기준으로 하였을 때 TC검색 과정이다. 초기 상태에서 기준단어 "경찰"과 토픽 내 단어 "아이"에 대하여 $PMI(\text{경찰},\text{아이})<0$이다. Step1에서는 "경찰"을 TC의 정점으로 추가하는 동시에 단어 "아이"는 기존 Matrix에서 삭제한다. Step1에서 다음 단어

표 1 LDA기법으로 추출된 토픽
Table 1 Topics Extracted using the LDA Technique

Topic 01	Topic 02	Topic 03	Topic 04	Topic 05	Topic 06	Topic 07
경찰	경찰	부사장	청와대	가격	소니	대통령
여성	시신	대한항공	문건	달러	영화	새누리당
남편	담배	조현아	검찰	러시아	택시	국회
병원	수원	사무장	비서관	제품	서울시	여당
아들	팔달산	승무원	대통령	모델	해킹	새정치민주연합
차량	용의자	항공기	정윤회	스마트폰	삼단봉	여야
범행	비닐봉지	국토부	경정	유가	버스	일본
사고	토막시신	땅콩	유출	하락	차량	박근혜
피해자	수색	서비스	수사	서비스	공격	위원장
할머니	수원시	검찰	경위	삼성전자	직원들	총리
수사	범행	비행기	인사	판매	에네스	후보
인천	장기	기장	박지만	일본	인터넷	원내대표
신고	피의자	승객	조응천	샤오미	사이버	개혁
살해	수사	조현민	국정개입	가구	감독	여당
경위	수사본부	회항	박근혜	대비	논란	정권
호주	피해자	논란	비서실장	분기	성희롱	선거
아파트	경기도	사과	박관천	출시	온라인	문재인
아이	박춘봉	전무	행정관	자동차	개봉	새정치연합
가방	팔달	매뉴얼	국정	아이폰	김정	공무원연금
신은미	등산로	리턴	논란	차량	카야	국정조사

정답토픽1 정답토픽2 오류토픽

표 2 토픽 내 단어 간 유사도의 예시

Table 2 Example of Similarity between Words in a Topic

	경찰	여성	남편	병원	아들	차량	범행	사고	피해자	할머니	수사	인천	신고	살해	경위	호주	아파트	아이	가방	신은미
경찰	-	1.12	1.09	1.17	0.84	1.10	2.06	0.43	1.57	1.54	1.34	0.53	1.56	2.11	1.73	0.54	0.60	-0.44	1.88	1.65
여성	1.12	-	1.85	0.94	1.00	0.10	1.61	0.04	1.62	-0.09	0.30	-0.52	0.81	1.68	0.32	0.90	0.84	0.64	0.72	0.36
남편	1.09	1.85	-	1.83	2.76	0.00	0.96	0.68	0.52	1.89	0.23	-0.13	1.36	1.13	1.26	0.55	2.58	0.89	-∞	0.53
병원	1.17	0.94	1.83	-	1.69	0.75	0.66	1.49	0.77	0.42	0.14	0.22	1.73	-0.35	1.22	0.26	0.29	1.03	-1.69	-0.18
아들	0.84	1.00	2.76	1.69	-	-0.37	0.41	0.50	0.66	2.32	-0.13	0.94	1.14	1.15	0.67	0.57	1.42	1.46	1.52	-0.26
차량	1.10	0.10	0.00	0.75	-0.37	-	-0.24	1.32	1.09	-∞	-0.21	0.24	1.25	-0.09	0.58	-0.99	0.75	-0.53	-0.93	-∞
범행	2.06	1.61	0.96	0.66	0.41	-0.24	-	-0.84	2.82	2.30	1.66	0.95	1.80	3.54	1.31	0.48	0.42	-0.60	3.02	2.35
사고	0.43	0.04	0.68	1.49	0.50	1.32	-0.84	-	0.33	-0.61	-0.04	0.12	1.07	-1.39	0.84	0.22	0.11	0.06	0.18	0.78
피해자	1.57	1.62	0.52	0.77	0.66	1.09	2.82	0.33	-	1.45	1.25	0.55	1.44	2.57	0.09	-0.30	-1.05	-0.35	1.76	0.94
할머니	1.54	-0.09	1.89	0.42	2.32	-∞	2.30	-0.61	1.45	-	0.87	2.83	2.16	3.52	0.21	-0.10	0.79	1.03	4.44	-∞
수사	1.34	0.30	0.23	0.14	-0.13	-0.21	1.66	-0.04	1.25	0.87	-	0.45	0.97	1.61	1.67	-1.71	-1.14	-0.75	1.34	0.93
인천	0.53	-0.52	-0.13	0.22	0.94	0.24	0.95	0.12	0.55	2.83	0.45	-	0.70	2.05	0.37	0.05	0.32	-0.05	2.96	-1.98
신고	1.56	0.81	1.36	1.73	1.14	1.25	1.80	1.07	1.44	2.16	0.97	0.70	-	1.88	1.01	-0.12	0.33	0.24	1.74	-1.73
살해	2.11	1.68	1.13	-0.35	1.15	-0.09	3.54	-1.39	2.57	3.52	1.61	2.05	1.88	-	1.38	-0.10	0.88	-0.87	3.82	-∞
경위	1.73	0.32	1.26	1.22	0.67	0.58	1.31	0.84	0.09	0.21	1.67	0.37	1.01	1.38	-	-1.37	0.87	-1.33	1.35	1.19
호주	0.54	0.90	0.55	0.26	0.57	-0.99	0.48	0.22	-0.30	-0.10	-1.71	0.05	-0.12	-0.10	-1.37	-	-∞	0.13	1.11	-∞
아파트	0.60	0.84	2.58	0.29	1.42	0.75	0.42	0.11	-1.05	0.79	-1.14	0.32	0.33	0.88	0.87	-∞	-	0.87	-0.64	-0.72
아이	-0.44	0.64	0.89	1.03	1.46	-0.53	-0.60	0.06	-0.35	1.03	-0.75	-0.05	0.24	-0.87	-1.33	0.13	0.87	-	-1.08	0.11
가방	1.88	0.72	-∞	-1.69	1.52	-0.93	3.02	0.18	1.76	4.44	1.34	2.96	1.74	3.82	1.35	1.11	-0.64	-1.08	-	3.08
신은미	1.65	0.36	0.53	-0.18	-0.26	-∞	2.35	0.78	0.94	-∞	0.93	-1.98	-1.73	-∞	1.19	-∞	-0.72	0.11	3.08	-

LDA기법으로 추출된 토픽 내 단어 출현 빈도 순

단어	경찰	여성	남편	병원	아들	차량	범행	사고	피해자	할머니	수사	인천	신고	살해	경위	호주	아파트	아이	가방	신은미
빈도	1409	378	253	247	237	214	195	192	179	174	167	160	155	146	145	139	138	136	135	114

그림 4 LDA로 추출된 토픽의 단어빈도 순서

Fig. 4 Word Frequency order of Topics Extracted by LDA

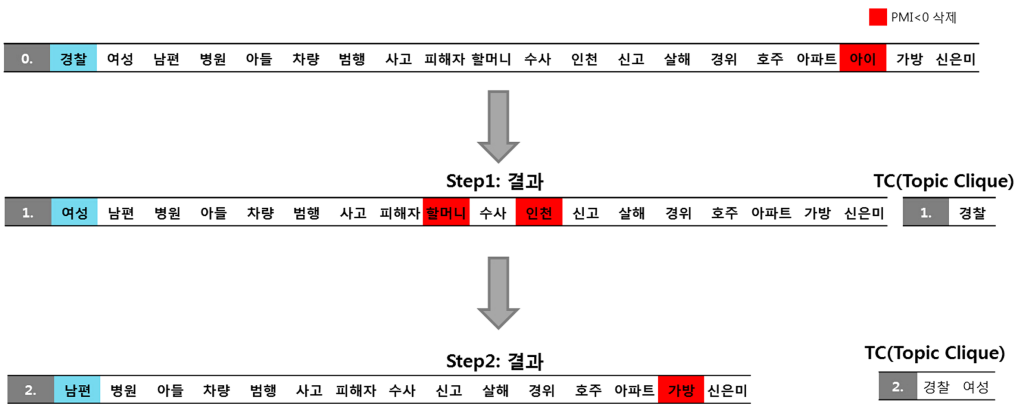


그림 5 "경찰"을 기준으로 하였을 때 TC

Fig. 5 Example of TC Search Process Based on "경찰"

표 3 "경찰"을 기준단어로 한 TC 결과
Table 3 TC Results with Reference to "경찰"

	경찰	여성	남편	병원	아들	범행	피해자	경위
경찰	-	1.12	1.09	1.17	0.84	2.06	1.57	1.73
여성	1.12	-	1.85	0.94	1.00	1.61	1.62	0.32
남편	1.09	1.85	-	1.83	2.76	0.96	0.52	1.26
병원	1.17	0.94	1.83	-	1.69	0.66	0.77	1.22
아들	0.84	1.00	2.76	1.69	-	0.41	0.66	0.67
범행	2.06	1.61	0.96	0.66	0.41	-	2.82	1.31
피해자	1.57	1.62	0.52	0.77	0.66	2.82	-	0.09
경위	1.73	0.32	1.26	1.22	0.67	1.31	0.09	-

"여성"을 기준단어로 하였을 때 PMI(여성, 할머니) <0 이며 PMI(여성, 인천) <0이다. Step2에서는 "여성"을 TC의 다음 정점으로 저장하고 동시에 기존 Matrix에서 단어 "할머니"와 "인천"을 삭제한다. 또한 Step2에서는 빈도순 다음 단어 "남편"을 기준단어로 하였을 때 PMI(남편, 가방) <0을 구한다. 기존 Matrix에 단어가 하나 남았을 때 까지 상기 과정을 반복하여 실행한다.

표 3은 LDA로 추출된 토픽 내 단어 "경찰"을 기준으로 하였을 때 생성된 TC이다. 생성된 TC는 $PMI>0$인 단어 쌍들만 포함한다.

LDA로 추출된 토픽 하나에서는 여러 개의 TC가 생성될 수 있다. 그림 6은 그림 4의 단어 빈도순으로 생성된 TC에서 중복 값을 제거한 결과이다. 추출된 TC에 순번을 추가하여 TC_i 로 정의한다.

그림 7은 토픽 분리 알고리즘이다.

3.4.3 토픽 병합

토픽 병합 과정에서는 TC간의 거리를 이용하여 병합

```
1. //TCs generation from LDATopic
2. Function ldaTopicSeparation(LDATopic)
3.   Let List TCs be an empty list
4.   FOR Term in LDATopic
5. //calculate PMI between Term and TermCross
6.     calPMI(Term, TermCross)
7.     IF PMI<=0
8.       //delete termCross from LDATopic
9.       deleteVertice(LDATopic, TermCross)
10.    END IF
11.    add(TCs, LDATopic)
12.    //add LDATopic to TCs
13.  END FOR
14.  return TCs
15. END
```

그림 7 토픽 분리 알고리즘
Fig. 7 Topic Separation Algorithm

여부를 판단한다. 본 연구에서는 TC간의 거리는 TC간의 점정의 합집합으로 구성된 새로운 그래프에서 $PMI\leq 0$인 간선의 비율로 정의한다. TC간의 거리가 작을수록 그래프의 유사도가 높고 병합성이 높다. $V(TC_i)$을 TC_i에서의 정점의 집합이라 할 때, 그림 6에서 $V(TC_1)$ = {경찰, 여성, 남편, 병원, 차량, 사고, 피해자, 신고, 경위}이고 TC_2 = {경찰, 여성, 남편, 병원, 아들, 범행, 피해자, 신고, 경위}이며, $V(TC_1)\cup V(TC_2)$ = {경찰, 여성, 남편, 병원, 피해자, 신고, 경위, 아들, 차량, 범행, 사고}이다. 표 4는 TC_1와 TC_2로 구성된 새로운 그래프에서 정점들의 간선 값을 Matrix로 표현한 것이다. 표 4에서 $PMI\leq 0$인

TC1. [경찰, 여성, 남편, 병원, 차량, 사고, 피해자, 신고, 경위]
TC2. [경찰, 여성, 남편, 병원, 아들, 범행, 피해자, 신고, 경위]
TC3. [경찰, 여성, 남편, 병원, 아들, 사고, 피해자, 신고, 경위]
TC4. [경찰, 남편, 병원, 아들, 범행, 피해자, 할머니, 신고, 경위]
TC5. [경찰, 여성, 남편, 병원, 범행, 피해자, 수사, 신고, 경위]
TC6. [경찰, 병원, 아들, 범행, 피해자, 할머니, 인천, 신고, 경위]
TC7. [경찰, 여성, 남편, 아들, 범행, 피해자, 신고, 살해, 경위]
TC8. [경찰, 여성, 남편, 병원, 아들, 범행, 호주]
TC9. [경찰, 여성, 남편, 병원, 아들, 범행, 신고, 경위, 아파트]
TC10. [여성, 남편, 병원, 아들, 사고, 신고, 아파트, 아이]
TC11. [경찰, 여성, 아들, 범행, 피해자, 신고, 살해, 경위, 가방]
TC12. [경찰, 여성, 남편, 범행, 피해자, 수사, 경위, 신은미]

그림 6 LDA로 추출된 토픽에서의 TC 생성 예시
Fig. 6 Example of TC Generation in a Topic Extracted by LDA

표 4 TC간 거리 계산 예시

Table 4 Example of TC Distance Calculation

	경찰	여성	남편	병원	피해자	신고	경위	아들	차량	범행	사고
경찰	-	1.12	1.09	1.17	1.57	1.56	1.73	0.84	1.10	2.06	0.43
여성	1.12	-	1.85	0.94	1.62	0.81	0.32	1.00	0.10	1.61	0.04
남편	1.09	1.85	-	1.83	0.52	1.36	1.26	2.76	0.00	0.96	0.68
병원	1.17	0.94	1.83	-	0.77	1.73	1.22	1.69	0.75	0.66	1.49
피해자	1.57	1.62	0.52	0.77	-	1.44	0.09	0.66	1.09	2.82	0.33
신고	1.56	0.81	1.36	1.73	1.44	-	1.01	1.14	1.25	1.80	1.07
경위	1.73	0.32	1.26	1.22	0.09	1.01	-	0.67	0.58	1.31	0.84
아들	0.84	1.00	2.76	1.69	0.66	1.14	0.67	-	-0.37	0.41	0.50
차량	1.10	0.10	0.00	0.75	1.09	1.25	0.58	-0.37	-	-0.24	1.32
범행	2.06	1.61	0.96	0.66	2.82	1.80	1.31	0.41	-0.24	-	-0.84
사고	0.43	0.04	0.68	1.49	0.33	1.07	0.84	0.50	1.32	-0.84	-

간선의 수는 6개, 총 간선의 수는 110개이므로 $Distance(TC_1, TC_2) = \frac{6}{110}$이다.

LDA로 추출된 각 토픽에서 생성한 TC간의 최소 거리로부터 $Distance$가 $Threshold$보다 작은 경우 두 TC를 하나의 토픽으로 병합한다. $Threshold$의 최적값은 실험으로부터 학습된 값을 사용한다. TC간 병합을 진행하기 위해 본 연구에서는 그림 8과 같은 4가지 병합 조건을 제안한다.

표 5는 그림 6의 TC_1, TC_2간 병합 예시이다. 그림 8 (1)에 따르면 TC_1, TC_2의 토픽 병합 결과는 {경찰, 여성, 남편, 병원, 피해자, 신고, 경위, 아들, 차량, 범행, 사고}이며, 그림 8 (2)에 따르면 TC_1, TC_2의 토픽 병합의 결과는 {경찰, 여성, 남편, 병원, 피해자, 신고, 경위}이다.

그림 8 (3)에 따르면 V'_-의 정렬결과는 {아들, 차량, 범행, 사고}이다. 정점단어 {아들}을 V'_+에 추가 후 정점단어 {차량}을 V'_+에 추가 시 $PMI \le 0$인 간선이 하나 생성되므로 정점단어 {차량}을 삭제하고, 정점단어 {범행}을 V'_+에 추가하고, 정점단어 {사고}를 V'_+에 추가하였을 때 $PMI \le 0$인 간선이 하나 생성되므로 정점단어{사고}를 삭제한다. (3)에 따른 TC_1, TC_2 병합결과는 {경찰, 여성, 남편, 병원, 피해자, 신고, 경위, 아들, 범행}이다.

그림 8 (4)에 따르면 $\max_i(avgPMI(TC_i))$에 해당되는 TC_i은 {경찰, 여성, 남편, 병원, 아들, 범행, 피해자, 신고, 경위}이며 이 경우 $avgPMI(TC_i)$은 1.26이다. 그림 9는 TC 병합 알고리즘이다.

(1) Merge topics into a word set $V' = V(TC_i) \cup V(TC_j)$

(2) Merge topics into a word set $V'_+ \subseteq V'$, where $\forall u, v \in V'_+, PMI(u,v) > 0$

(3) For $V'_- \subseteq V'$, where $\forall u \in V'_-, \forall v \in V', PMI(u,v) \le 0$, sort each element in V'_- in descending order and add it into V'_+ one by one. When a vertex is added, if an edge with $PMI \le 0$ is created, the vertex is deleted.

(4) TC_i such that $\max_i(avgPMI(TC_i))$

$avgPMI(G)$: The average PMI for edges in G

그림 8 TC간 4가지 병합 조건

Fig. 8 Four Merge Conditions between TCs

표 5 TC_1, TC_2간 병합 예시

Table 5 Example of Merge between TC_1 and TC_2

	경찰	여성	남편	병원	피해자	신고	경위	아들	차량	범행	사고
경찰	-	1.12	1.09	1.17	1.57	1.56	1.73	0.84	1.10	2.06	0.43
여성	1.12	-	1.85	0.94	1.62	0.81	0.32	1.00	0.10	1.61	0.04
남편	1.09	1.85	-	1.83	0.52	1.36	1.26	2.76	0.00	0.96	0.68
병원	1.17	0.94	1.83	-	0.77	1.73	1.22	1.69	0.75	0.66	1.49
피해자	1.57	1.62	0.52	0.77	-	1.44	0.09	0.66	1.09	2.82	0.33
신고	1.56	0.81	1.36	1.73	1.44	-	1.01	1.14	1.25	1.80	1.07
경위	1.73	0.32	1.26	1.22	0.09	1.01	-	0.67	0.58	1.31	0.84
아들	0.84	1.00	2.76	1.69	0.66	1.14	0.67	-	-0.37	0.41	0.50
차량	1.10	0.10	0.00	0.75	1.09	1.25	0.58	-0.37	-	-0.24	1.32
범행	2.06	1.61	0.96	0.66	2.82	1.80	1.31	0.41	-0.24	-	-0.84
사고	0.43	0.04	0.68	1.49	0.33	1.07	0.84	0.50	1.32	-0.84	-

(1) $PMI > 0$인 값만 포함

(2) $PMI \le 0$값 포함

```
16. //Merge TCs
17. Function topicMerge(TCs)
18.   calDistance(TCs)
        //calculate distance among TCs
19.   distance = the smallest distance
      between TC_i and TC_j, for all i,j
20.   WHILE distance<=THRESHOLD and size(TCs)>1
21.     //get mergedTC by merging TC_i
      and TC_j with one of four merging
      conditions
22.     mergedTC = getMergedTC(TC_i, TC_j)
23.     delTC(TCs, TC_i, TC_j)
          //delete TC_i and TC_j from TCs
24.     addTC(mergedTC, TCs)
                //add mergedTC to TCs
25.   calDistance(TCs)
26.   distance = the smallest distance
        between TC_i and TC_j, for all i,j
27.   END WHILE
28.  return TCs
29. END
```

그림 9 TC 병합 알고리즘

Fig. 9 TC Merge Algorithm

4. 실험 및 결과

4.1 데이터 수집 및 실험 방법

4.1.1 데이터 수집

본 연구에서는 뉴스 토픽을 추출하기 위해 다음(www.

표 6 뉴스 데이터
Table 6 News Data

Source	Daum News
Period	2014.12.01. ~ 2014.12.31
Number of Documents	5,216

daum.net)에서 제공하는 뉴스 중 인기 뉴스를 수집하였다. 인기 뉴스에는 중복되는 뉴스가 존재하므로 중복된 뉴스는 따로 제거한다. 표 6은 수집된 뉴스 데이터에 대한 정보이다.

4.1.2 실험 방법

토픽 추출을 위하여 Stanford Topic Modeling Toolbox[14]를 사용하여 LDA 토픽을 추출하였다. 본 연구에서는 LDA 토픽 추출에서 2음절이상인 명사를 토픽 후보 단어로 선정하였다. 파라미터 추정을 위하여 사용한 알고리즘은 EM 알고리즘[14]이며 최대 Iterator 회수는 1,000번으로 하였다. LDA로 추출된 결과로부터 토픽을 보정하고 그 결과를 정답셋과 비교하여 성능을 평가하였다.

4.2 실험 결과

4.2.1 LDA 기반 토픽 추출

제공된 데이터에 대한 최적 LDA 토픽 모델의 추출을 위하여 Perplexity방법으로 LDA의 최적 파라미터를 추정하였다. 그림 10은 토픽 개수에 따른 Perplexity의 최소값이다. 그림 10의 X축은 $(TopicNum, \alpha, \beta)$이며 α은 LDA 토픽의 사전 분포(Prior Distribution), β은 LDA 특정 토픽의 단어 사전 분포,, $TopicNum$은 LDA 토픽의 개수를 의미한다.

실험 결과에 따르면 토픽 개수가 증가함에 따라 Perplexity는 감소하는 추세임을 알 수 있다. 실험 결과에 따라 본 연구에서는 파라미터 조합 $TopicNum = 35$, $\alpha = 1.0$, $\beta = 0.1$을 Local Optimum 파라미터로 설정하였다. 표 7은 4.2.1절의 파라미터 조합에 따라 LDA로 추출한 35개 토픽 중 7개이다.

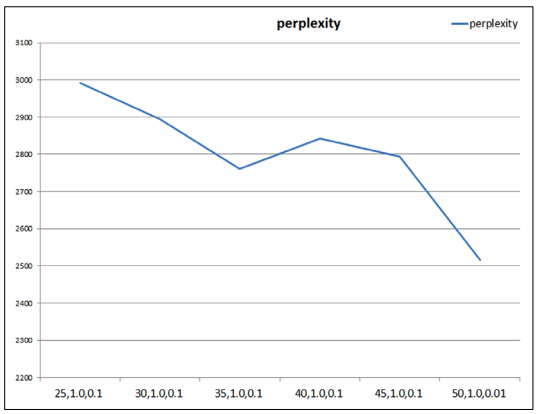


그림 10 토픽 개수별 최소값
Fig. 10 Minimum by Topic Count

표 7 LDA로 추출된 토픽
Table 7 Topics Extracted with LDA

Topic01	Topic02	Topic03	Topic04	Topic05	Topic06	Topic07
경찰	담배	부사장	서비스	대통령	경찰	국회
호주	인상	대한항공	세금	박근혜	시위	예산
신은미	담뱃값	조현아	모바	청와대	법원	개정안
카페	가격	사무장	혜택	인사	징역	가석방
콘서트	쿠바	승무원	은행	논란	판결	법안
인질	편의점	항공기	호텔	정권	소송	예산안
테러	금연	국토부	카드	권력	뉴욕	여야
시드니	흡연	땅콩	금융	새누리당	재판부	의원들
신씨	전자담배	서비스	결제	정윤회	범죄	본회의
금괴	천원	비행기	연말정산	대선	처벌	기업인
논란	사제기	기장	고객	국정	시위대	새정치민주연합
인질극	판매	회항	신용카드	발언	선고	일정
토크콘서트	니코틴	승객	연봉	국정원	혹인	공직자
종북	복지부	사과	스마트폰	실세	홍콩	지원
이라크	새해	논란	공제	파문	피해자	논란
마약	보육료	매뉴얼	소득공제	지지율	재판	도입
익산	물량	리턴	사례	나라	벌금	제출
재미동포	정책	박창진	핀테크	위기	불법	심사
토크	청소년	뉴욕	규제	후보	성폭행	출장
발언	단속	견과류	금액	평가	백인	새누리당

4.2.2 정답 토픽 추출

본 연구에서는 제안한 방법으로 추출된 토픽을 평가하기 위하여 [5]에서 제안한 F_{as}-$measure$ 방법을 사용하여 토픽추출의 정확도를 평가한다. 우선 추출된 토픽을 평가하기 위해 수집된 문서의 정답 토픽을 추출한다. 정답토픽의 추출방법은 전문가가 정한 토픽명칭으로 수집된 각 문서를 태깅하고 빈도를 측정한다. 본 실험에서는 40개 정답 토픽을 추출하며 토픽 당 단어는 20개를 정답으로 한다. 표 8은 추출된 40개 정답 토픽 중 7개이다.

추출된 토픽 적합성 평가는 정답토픽 집합과 제안방법으로 추출된 토픽의 집합과 비교하여 계산된다. 정답토픽 집합과 비교하여 추출된 토픽을 평가하기 위해 평가수식 $Precision_i$ 및 $Recall_i$을 사용한다. $Precision_i$은 i번째 토픽단어 중 정확하게 추출된 토픽단어의 비율을 의미하고, $Recall_i$은 i번째 토픽단어 중 정확하게 추출된 단어와 정답 셋 단어의 비율을 의미한다.

T : 뉴스 문서 집합에 대한 정답 토픽 집합

$Topic_j$: 뉴스 문서 집합에 대한 j번째 정답 토픽

표 8 정답 토픽
Table 8 Right Topics

Topic01	Topic02	Topic03	Topic04	Topic05	Topic06	Topic07
부사장	청와대	경찰	통진당	선원	모델	원전
대한항공	문건	시신	해산	오룡	하이브리드	한수원
조현아	검찰	팔달산	결정	러시아	현대차	호기
사무장	비서관	수원	통합진보당	사조산업	쏘나타	유출
승무원	대통령	비닐봉지	정당	사고	차량	공격
항공기	정윤회	토막시신	재판관	선장	쌍용차	악성코드
검찰	경정	수색	민주주의	침몰	기아차	해킹
국토부	유출	용의자	선고	부산	연비	도면
서비스	경위	수원시	의원직	선박	가격	사고
땅콩	수사	장기	정당해산	세월호	자동차	사이버
비행기	박지만	범행	판결	시신	판매	직원
승객	인사	수사	헌법재판소	가족들	출시	해커
기장	조용천	피의자	헌법재판	수색	신형	신고리원전
회항	국정개입	수사본부	김이수	한국인	현대	가동
조현민	비서실장	박준봉	활동	조업	티볼리	인터넷
논란	박근혜	피해자	이석기	선원들	디자인	인물
박창진	박관천	팔달	국회	구조	엔진	수사
사과	국정	박씨	대통령	오룡호	굴뚝	트위터
매뉴얼	행정관	등산로	민주적	파도	공장	합수단
리턴	이재만	토막	헌법	어획물	가솔린	가스

$\hat{T}$: 뉴스 문서 집합에서 자동 추출된 토픽 집합
$\widehat{Topic_i}$: 뉴스 문서 집합에서 자동 추출된 토픽 중 i번째 토픽

$$Precision_i = \max_{Topic_j \in T}\left\{\frac{|Topic_j \cap \widehat{Topic_i}|}{|\widehat{Topic_i}|}\right\}$$
$$Recall_i = \max_{\widehat{Topic_i} \in \hat{T}}\left\{\frac{|Topic_j \cap \widehat{Topic_i}|}{|Topic_j|}\right\} \quad (4)$$

본 연구에서는 $Precision_i$ 및 $Recall_i$을 이용하여 추출된 토픽 집합에 대한 적합성 평가식인 $F_{AS}-measure$을 사용한다. ASP(Average Set Precision)은 자동 추출된 모든 토픽에 대해 정답 토픽 집합과의 Precision의 평균(자동 추출된 토픽이 정답 토픽을 맞춘 수준)을 의미하며, ASR(Average Set Recall)은 모든 정답 토픽에 대해 자동 추출된 토픽과의 Recall의 평균(정답 토픽이 자동 추출된 토픽 집합에 재현된 수준)을 의미한다.

$$F_{AS}-measure = 2\times\frac{ASP\times ASR}{ASP+ASR}$$
$$ASP = avg_{i\in I}\{Pricision_i\}$$
$$ASR = avg_{i\in I}\{Recall_i\} \quad (5)$$

4.2.3 제안 방법에 의한 토픽 추출

제안방법의 토픽 분리 모듈을 사용하여 토픽을 추출한다. 그림 11은 4가지 병합 방법의 최고값에 대한 비교 실험 결과이다. 그림 11의 실험 결과에 따르면 병합 방법 (4)에서 $Threshold = 0.3$일 때 최고 F_{AS}-$measure$및 ASP값이 산출되었다.

본 연구에서는 최고 F_{AS}-$measure$값을 나타낸 병합 방법4로 토픽을 추출하였다. 표 9는 병합 방법 4, $Threshold$의 값이 0.3일 때 추출된 38개 토픽 중 7개 토픽이다.

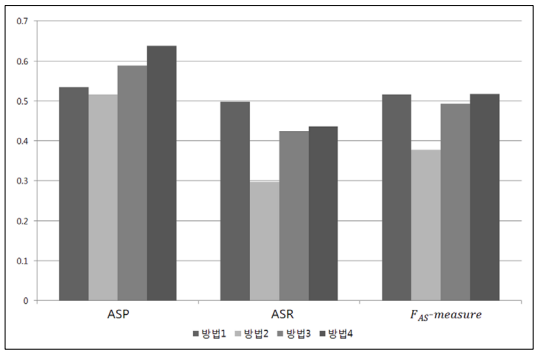


그림 11 네 가지 제안방법의 성능 비교
Fig. 11 Performance Comparison of Four Proposed Methods

표 9 제안 방법으로 추출된 토픽
Table 9 Topics Extracted by Suggestion Method

Topic01	Topic02	Topic03	Topic04	Topic05	Topic06	Topic07
경찰	호주	담배	부사장	서비스	대통령	시위
신은미	카페	인상	대한항공	세금	박근혜	장역
콘서트	인질	담뱃값	조현아	혜택	청와대	소송
테러	테러	가격	사무장	은행	인사	뉴욕
신씨	시드니	편의점	승무원	카드	논란	범죄
토크콘서트	인질극	금연	항공기	금융	정권	처벌
종북	이라크	흡연	국토부	결제	권력	시위대
익산		전자담배	땅콩	연말정산	새누리당	흑인
재미동포		천원	서비스	신용카드	정윤회	벌금
토크		사재기	비행기	공제	대선	백인
		판매	기장	소득공제	국정	
		니코틴	회항	사례	발언	
		복지부	승객	금액	국정원	
		세해	사과		실세	
			논란		파문	
			매뉴얼		지지율	
			리턴		나라	
			박창진		위기	
			뉴욕		후보	
			견과류		평가	

4.2.4 실험 평가

그림 12는 토픽 병합 방법4를 사용한 제안방법과 LDA의 토픽 추출 단어의 적합성의 비교이다. 실험 결과에 따르면 제안방법의 ASR은 약 0.05 하락하였지만 ASP가 약 0.14 상승하여 F_{as}-$measure$가 LDA보다 약 0.02 상승하였다.

그림 13은 설문조사를 통하여 LDA와 제안방법의 토픽추출 결과를 각각 정답토픽과 비교하고 F-$measure$로 평가한 결과이다. LDA는 추출된 35개 토픽 중 28개가 정답이고 제안방법은 추출된 38개 토픽 중 31개가 정답이었다. 제안방법의 F-$measure$는 약 0.07 상승하였고, $Precision$은 약 0.09 상승하였으며, $Recall$은 약 0.05 상승하여 기존 LDA기법보다 좋은 성능을 보였다. 본 결과는 제안방법의 토픽 주제에 대한 정확도의 향상을 보여준다.

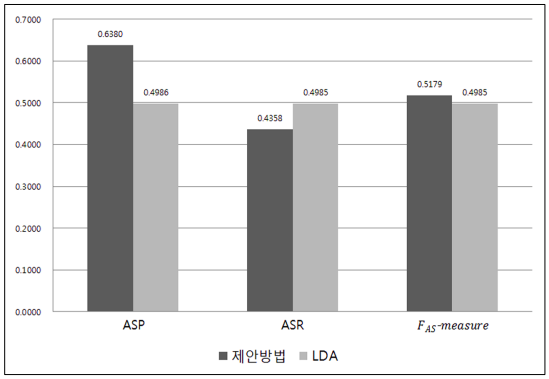


그림 12 제안방법과 LDA의 F_{AS}-$measure$ 비교

Fig. 12 F_{AS}-$measure$ Comparison of the Proposed Method and LDA

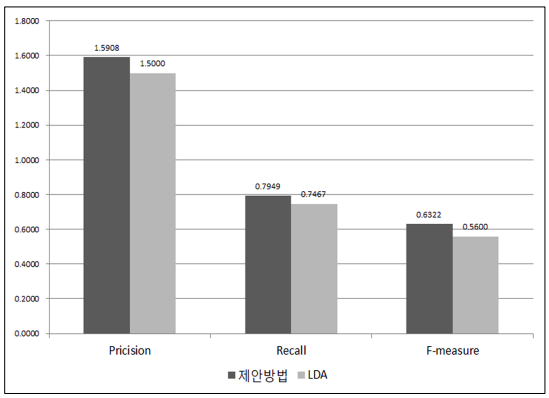


그림 13 제안방법과 LDA의 F-$measure$ 비교

Fig. 13 F-$measure$ Comparison of the Proposed Method and LDA

5. 결론 및 향후 계획

본 연구에서는 LDA로 추출한 토픽 내 단어 간 유사도를 사용하여 토픽 혼재 문제 및 토픽 중복 문제를 해결하는 방법을 제안하였다. 본 논문에서는 토픽 중복 문제 대해 강건한 잠재 디리클레 할당으로 토픽을 추출하고, 단어 간 유사도를 이용하여 토픽 분리 및 토픽 병합의 단계를 거치는 새로운 토픽 추출 방법을 제안하였다. 토픽분리를 위하여 토픽 내 단어를 정점으로 하고, 단어 간 PMI가 0보다 큰 완전 부분 그래프로 TC(Topic Clique)를 정의하였으며, TC간의 정점의 합집합으로 구성된 새로운 그래프에서 $PMI \leq 0$인 간선의 비율을 TC 간의 거리로 정의하고 이를 이용한 네 가지 토픽 병합 방법을 제시하였다.

실험 결과에 따르면 토픽 병합 방법 (4)에서 $Threshold = 0.3$일 때 최고의 F_{AS}-$measure$및 ASP값이 산출되었다. 또한, 토픽 병합 방법 4를 사용한 제안방법과 LDA의 토픽 추출 단어의 적합성의 비교로부터 ASR은 약 0.05 하락하였지만 ASP가 약 0.14 상승하여 F_{as}-$measure$가 LDA보다 약 0.02 상승하였다. 또한, 설문조사를 통하여 제안방법과 LDA의 추출된 토픽의 정확도를 비교한 결과 제안 방법의 F-$measure$은 약 0.07 상승하였고, $Precision$은 약 0.09 상승하였고, $Recall$은 약 0.05 상승하였다.

뉴스문서 셋에는 뉴스의 시간, 지역등 유용한 정보들이 많이 포함되어 있다. 제안 방법은 이러한 시간, 지역 파라미터를 사용되지 않았으므로 시간별 토픽 변화의 추세, 지역별 토픽의 차별성에 대한 추가 연구가 필요하다. 또한 PMI에 의한 유사도 계산시 동음이의어에 대한 처리 등에 대한 고려가 추가적으로 필요하다.

References

[1] Wikipedia, topic model, http://en.wikipedia.org/wiki/Topic_model, 2015.

[2] Landauer, T. K., Foltz, P. W., & Laham, D., An introduction to latent semantic analysis, *Discourse processes*, 25(2-3), pp. 259-284, 1988.

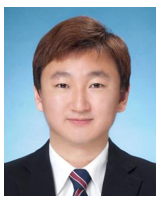
[3] Blei, D. M., Ng, A. Y., & Jordan, M. I., Latent Dirichlet Allocation, *The Journal of Machine Learning Research*, 3, pp. 993-1022, 2003.

[4] Wang, Y., Zhao, X., Sun, Z., Yan, H., Wang, L., Jin, Z., ... & Zeng, J., Peacock: Learning long-tail topic features for industrial applications. arXiv preprint arXiv:1405.4402, 2014.

[5] Noh, J., Lee, S., Extracting and Evaluating Topics by Region, *Multimedia Tools and Application, 75 (20)*, 2016.

[6] Wikipedia, LDA, https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation, 2015.

[7] Dumais, S. T., Latent semantic analysis, *Annual review of information science and technology*, *38(1)*, pp. 188-230, 2004.
[8] Hofmann, T., Probabilistic latent semantic analysis, *Proc. of the Fifteenth conference on Uncertainty in artificial intelligence*, pp. 289-296, Morgan Kaufmann, 1999.
[9] Hofmann, T., Probabilistic latent semantic indexing. *Proc. of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 50-57, ACM, 1999.
[10] Kazama, J. I., De Saeger, S., Kuroda, K., Murata, M., & Torisawa, K., A Bayesian method for robust estimation of distributional similarities, *Proc. of the 48th Annual Meeting of the Association for Computational Linguistics*, pp.247-256, Association for Computational Linguistics, 2010.
[11] Wikipedia, perplexity, [Online]. Available: https://en.wikipedia.org/wiki/Perplexity, 2015.
[12] [Online]. Available: http://media.daum.net/netizen/hotlivenation/
[13] [Online]. Available: http://nlp.stanford.edu/software/tmt/tmt-0.4/
[14] Wikipedia, Expectation - maximization algorithm, [Online]. Available: https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm



김 동 욱

2009년 중국 하얼빈이공대학교 컴퓨터학과(학사). 2015년 숭실대학교 컴퓨터학과(석사). 관심분야는 Data Science, Advertising, Text Mining 등

이 수 원

1982년 서울대학교 계산통계학과(학사) 1984년 한국과학기술원 전산학과(석사) 1994년 University of Southern California 전산학과(박사). 1995년~현재 숭실대학교 소프트웨어학부 교수. 2008년~2009년 한국정보과학회논문지(SA) 편집위원장. 2003년~2004년 한국정보과학회논문지 인공지능연구회 운영위원장. 관심분야는 Data Science, 인공지능, Text Mining 등