

4장 Content-Based Recommender Systems

이마태

목차

1. 개요
2. 예측 단계
3. Feature Extraction
4. Learning User Profiles
5. Q&A

1. 개요

- Collaborative Filtering과의 차이
 - CF는 예측을 위해 user rating 이용
 - Content-based는 item의 고유한 특성 이용
- Content-based system에 사용되는 데이터
 - **Content-centric attributes**: 아이템의 특성 (descriptive sets of attributes)
 - **User profile**: 유저가 다른 item에 매긴 feedback (rating, action)

1. 개요

- 다른 유저의 rating은 반영되지 않음
 - 장점: cold-start problem을 부분적으로 해결
 - 다른 유저가 해당 아이템에 매긴 정보가 부족할 때, 해당 유저의 관심사를 가지고 추천
 - 새로운 유저가 추가되었을 때, 콘텐츠
 - 새로운 아이템이 추가되었을 때, 기존 아이템의 정보 활용
- 단점: 추천 아이템의 다양성 감소
 - 다른 유저들이 평가한 새로운 아이템이 추천될 가능성이 적음
 - 해당 유저가 이미 평가했거나, 뻔한(obvious) 아이템을 추천해 줌

1. 개요

- 텍스트가 많고 구조화 되어 있지 않은 도메인에 적합
 - 웹 페이지
 - 이커머스 (제품 description)
 - 생산자, 장르, 가격 등 구조화된 특성까지 사용
- 비정형 텍스트에서 keyword를 추출하여 벡터화, 예측에 사용
 - keyword-based vector-space
 - 예: $\{ \text{'apple'} : 0.3, \text{'tomato'} : 0.9, \text{'banana'} : 0.1 \}$

2. 예측 단계

1. Preprocessing + Feature Extraction (*Offline phase*)

- 웹 페이지, 제품 정보, 뉴스, 음악 등 다양한 소스로부터의 비정형 데이터를 **keyword-based vector-space**로 변형

2. Content-based learning of user profiles (*Offline Phase*)

- 유저의 피드백 (explicit or implicit) 을 통해 **user-specific** 모델 생성
- **해당 유저**에게 제공되는 모델이기 때문

3. Filtering + Recommendation (*Online Phase*)

- 학습 결과를 통해 예측
- 실시간으로 이루어져야 하기 때문에 **처리 속도**가 중요

3. Feature Extraction

- 아이템의 고유한 특성을 추출하는 과정은 도메인에 따라 다름

1. 영화

- 시놉시스
 - 슈렉: “늪에 마법의 생물이 가득하고 나서, 한 오우거가 땅을 되찾기 위해 악당 영주로부터 공주를 구하기로 결심한다.”
- 감독
- 배우
- 장르

→ 여러 종류의 키워드들의 중요도가 다름

→ 도메인 지식을 통해 직접 매기거나, 모델을 통해 계산 (feature weighting)

3. Feature Extraction

- 아이템의 고유한 특성을 추출하는 과정은 도메인에 따라 다름

2. 웹 페이지

- 웹 문서의 제목, 메타데이터, body 내용 등 여러 요소들의 중요도 계산
 - 제목은 body보다 높은 weight
- Anchor text
 - 링크로 연결되는 페이지에 대한 description
 - 해당 페이지 자체와는 연관성이 낮을 수 있음 → 해당 페이지에서는 제거되기도 함
- 광고 배너 등 해당 문서와 관련 없는 내용이 있음
 - Main block 내용만을 추출
 - 사이트의 구조에 따라서 main block을 정의하기 어려운 경우, 사이트의 tag tree를 가지고 구조를 학습하기도 함
 - Main block을 Labeling하는 것도 일종의 분류 문제

3. Feature Extraction

- 아이템의 고유한 특성을 추출하는 과정은 도메인에 따라 다름

3. 음악

- Pandora Internet Radio
 - Music Genome Project에서 매긴 트랙들의 피쳐 이용하여 아이템 추천
 - *Synth riffs, straight drum beats, ...*
- 자신이 원하는 Feature를 선택하면, 해당 Feature와 유사한 노래가 재생됨
- 노래에 좋아요/싫어요 평가 가능 → 유저의 Feedback을 이용하여 더 정교한 추천 모델 개발
- 초기에는 몇 가지의 피쳐만을 참고하는 Knowledge-Based System
- 이후 Rating이 추가되어 Content-Based System

3. Feature Extraction – Cleaning

- Feature 추출 후 적절한 형태로 가공하는 단계
 1. Stop-words 제거
 - 의미를 가지고 있지 않은 단어 제거
 - *A, the, It, They*
 2. Stemming
 - 같은 의미이지만 형태가 다른 단어를 묶음
 - *Hope, Hoping*
 3. Phrase 추출
 - 여러 어절로 된 단어 추출
 - *Hog Dog*: 각 단어의 뜻과는 다른 새로운 단어
- 이후 키워드들은 **vector-space representation**으로 변환하여 유의미한 단어(*informative words*) 추출

3. Feature Extraction – Cleaning

- TF-IDF: **vector-space representation** 예시
 - 각 단어를 *term* 이라고 함
 - TF (Term Frequency)
 - $TF(d,t)$: 문서 d 에서 단어 t 의 개수
 - IDF (Inverse Document Frequency)
 - $IDF(t)$: 단어 t 가 등장한 문서 개수
 - $Log \frac{n}{1+n_i}$, where n = 전체 문서 개수, n_i = 해당 단어가 있는 문서 개수
 - 문서별 $tf-idf = tf(d,t) \times idf(t)$
 - **vector-space representation**: [단어1의 $tf-idf$, 단어2의 $tf-idf$, ...]
 - 모든 문서에 자주 등장하는 단어는 **중요도 낮음**
 - 특정 문서에서만 자주 등장하는 단어는 **중요도 높음**
 - 문서의 단어별 중요도를 embedding하는 과정

3. Supervised Feature Extraction

- 유저 Rating을 반영하여 Feature Importance 계산하는 지도 방법
 - 참고: TF-IDF와 같은 방법은 유저의 피드백이 반영되지 않은 비지도 방법
- 예: Gini Index

$$\text{Gini}(w) = 1 - \sum_{i=1}^t p_i(w)^2$$

- Feature selection 지표
- 각 value가 분산된 정도를 나타냄
- w : 각 단어, $p_i(w)$: 각 rating이 차지하는 비율
- 경제학에서는 소득 분배 정도. 0이면 완전 평등, 1이면 완전 불평등
- 작을수록 고르게 분산된 것 (smaller value \rightarrow greater discriminative power)

3. Supervised Feature Extraction

Gini Index — Binary rating일 때 (좋아요/싫어요)

$$\text{Gini}(w) = 1 - \sum_{i=1}^t p_i(w)^2$$



	좋아요 비율	싫어요 비율	Gini(w)
단어 1	1	0	1
단어 2	0.9	0.1	0.18
단어 3	0.6	0.4	0.48
단어 4	0.5	0.5	0.5

- Gini Index가 작은, 즉 고르게 분산된 단어 Selection
- 문서의 단어별 Gini Index 임베딩(**vector-space representation**): $[Gini(w_1), Gini(w_2), \dots]$

4. Learning User Profiles

- User Rating이 discrete 할 경우 (ex: 좋아요/싫어요) → **분류** 문제
- User Rating이 numeric 할 경우 (ex: 0-10점) → **회귀** 문제

4. Learning User Profiles

	Label(Y, Rating)	X (Item Attributes)
Train Items	좋아요	vector-space representation
	좋아요	
	싫어요	
	싫어요	
	좋아요	
	좋아요	
	좋아요	
Test Items	?	Unsupervised: $[tf-idf(w_1), tf-idf(w_2), \dots]$ Supervised: $[Gini(w_1), Gini(w_2), \dots]$
	?	

Train Items를 이용하여 Test Items의 Rating 예측

- Nearest Neighbor Classifier: 아이템 간의 유사도 기반 (ex: cosine similarity)
- Bayes Classifier
- Rule-Based Classifier

요약

- Content-Based System의 설명 가능성

- 아이템의 특성을 이용하기 때문에, 유저에게 어떤 아이템으로 인해 추천되었는지 설명 가능

“We are playing this track because it features trance roots, four-on-the-floor beats, disco influences, a knack for catchy hooks, beats made for dancing, straight drum beats, clear pronunciation, romantic lyrics, storytelling lyrics, subtle buildup/breakdown, a rhythmic intro, use of modal harmonies, the use of chordal patterning, light drum fills, emphasis on instrumental performance, a synth bass riff, synth riffs, subtle use of arpeggiated synths, heavily effected synths, and synth swoops.”

감사합니다