



사용자 질의어 특징을 반영한 하이라이트 기반 노래 가사 검색

Highlight based Lyrics Search Considering the Characteristics of Query

김권양*

Kweon Yang Kim

*경일대학교 컴퓨터공학과

*Dept. of Computer Engineering, Kyungil University

요 약

본 논문에서는 사용자가 노래 가사를 입력으로 음악을 검색할 때 사용자의 질의어 특징을 반영한 검색 방법을 제안한다. 일반적으로 노래 가사 검색에서 사용자가 작성하는 질의어들은 음악 하이라이트 부분에 해당된다는 점을 고려하여 본 논문에서는 노래 가사를 색인할 때, 하이라이트 부분이 더 중요하도록 만든다. 이를 위해 본 논문에서는 응집 계층 군집화를 사용하여 자동으로 음악 하이라이트 부분을 찾고, 하이라이트 부분과 그 주변 부분을 중요하게 고려할 수 있는 가우시안 중요도를 제안한다. 이 가우시안 함수는 평균을 하이라이트 부분으로 설정함으로써 하이라이트에서 가장 높은 값을 가지며, 주변부는 하이라이트보다 낮은 중요도를 가진다. 이렇게 얻어진 중요도와 함께 노래 가사를 색인함으로써 사용자들이 작성한 질의어에 대해 더 부합하는 검색 결과를 제공해준다. 실험에서 실사용자 5명에 대해 다양한 질의 타입들과 함께 평가하였으며, 가중치를 고려하지 않는 비교 모델보다 제안한 방법이 효과적임을 보인다.

키워드 : 하이라이트 기반 가사 검색, 응집 계층 군집화, 가우시안 중요도

Abstract

This paper proposes a lyric search method to consider the characteristics of the user query. According to the fact that queries for the lyric search are derived from highlight parts of the music, this paper uses the hierarchical agglomerative clustering to find the highlight and proposes a Gaussian weighting to consider the neighbor of the highlight as well as highlight. By setting the mean of a Gaussian weighting at the highlight, this weighting function has higher weights near the highlight and the lower weights far from the highlight. Then, this paper constructs a index of lyrics with the gaussian weighting. According to the experimental results on a data set obtained from 5 real users, the proposed method is proved to be effective.

Key Words : Highlight based Lyric Search, Hierarchical Agglomerative Clustering, Gaussian Weighting

Received: Jul. 5, 2016
Revised : Aug. 1, 2016
Accepted: Aug. 5, 2016
† Corresponding authors
kykim@kiu.ac.kr

1. 서 론

디지털 음원 시장이 성장함에 따라 사용자가 온라인에서 음악을 찾는 경우가 많아졌다. 사용자들은 가수 이름이나 노래 제목 등을 질의어로 제시하고 음악 검색 시스템들은 입력된 질의를 분석하여 질의에 부합하는 음악을 찾아준다. 최근 사용자에게 편의를 제공하고자 SoundHound¹⁾ 및 Shazam²⁾과 같은 음악 검색 서비스는 허밍, 멜로디 등과 같이 오디오 시그널 정보를 질의로 음악 검색을 시도하고 있다. 하지만 아직 많은 사용자들은 오디오 시그널 대신 텍스트 가사를 입력으로 음악을 검색하고 있으며, 본 논문에서는 노래 가사를 질의어로 하여 음악을 찾아주는 가사 검색만을 다룬다.

노래 가사 검색이란 자연어로 작성된 노래 가사의 일부를 질의어로 제시하면 질의어에 가장 부합하는 노래를 찾아주는 것이다. 예를 들어, “집 떠나와 열차타고”와 같이 자연어로 작성된 질의어에 대해 “김광석”의 “이등병의 편지”를 찾는다. 이러한 노래 가사 검색은 자연어로 처리한다는 특성상 일반적인 텍스트 검색으로 간주할 수 있다. 즉, 노래 가사를 하나의 문서로 간주하며, 가사에 대한 질의어가 주어질 때(질의어 또한 문서로 간주), 질의어에 가장 적합한 노래 가사를 찾아줄 수 있다.

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

하지만 노래 가사 검색을 단순히 기존 일반적인 텍스트 검색으로 해결하기에는 노래 가사가 가지는 특수성을 반영하지 못함으로 인해 높은 검색 성능을 기대하기 어렵다. 일반적으로 작사자와 작곡가들은 비슷한 리듬이나 가사 반복 등 여러 작사/작곡 기법을 통해 음악을 작성하며, 이를 통해 음악을 듣는 사람들이 음악을 대표하는 부분(하이라이트 부분)을 기억하도록 만든다[1]. 즉, 사용자들은 음악의 모든 부분을 기억하고 있는 것이 아니라 음악의 하이라이트 부분을 일반적으로 기억하고 있으며, 이에 따라 노래 가사 검색 시, 하이라이트 부분에 해당하는 노래 가사를 질의어로 사용한다. 예를 들어, “싸이(PSY)”의 “강남스타일”을 찾을 경우, 대부분 사용자들은 “강남스타일”의 시작 부분인 “낮에는따사로운”이나 반복 부분인 “오빤 강남스타일”등을 질의어로 사용한다. 반면 “근육보다 사상이”, “울퉁불퉁한 사나이”등과 같이 하이라이트가 아닌 부분에 위치한 가사들을 질의어로 사용하는 경우는 드물다. 즉, 노래 가사 검색에서의 질의어는 일반적인 텍스트 검색에서의 질의어와는 다른 특징을 가지기 때문에 효과적인 노래 가사 검색을 위해서는 이러한 특징을 고려하여야 한다.

본 논문에서는 질의어의 특징을 고려한 노래 가사 검색을 제안한다. 제안한 방법은 노래 가사를 색인할 때, 가사의 부분, 부분마다 서로 다른 가중치를 가지고 색인을 수행한다. 즉, 하이라이트 부분에 해당하는 가사에 중요도를 높임으로써, 하이라이트 부분의 가사가 하이라이트 부분에 해당하지 않는 가사보다 더 중요하게 색인될 수 있도록 한다. 그 결과, 질의어가 하이라이트에서 색인된 가사와 일치하면, 하이라이트 외의 부분에서 일치된 것보다 높은 유사도 값을 갖도록 한다. 이 때, 중요한 것은 하이라이트 부분은 같은 장르의 음악이라 할지라도 서로 다르게 나타나며, 음악 전반에 걸쳐 여러 군데 나타날 수 있다. 본 논문에서는 이를 해결하기 위해 응집 계층 군집화(hierarchical agglomerative clustering)를 사용하여 하이라이트 개수와 가사 위치와는 무관하게 하이라이트 부분을 찾는다.

본 논문에서는 하이라이트 부분뿐 아니라 하이라이트 주변부까지 가중치를 고려한다. 이는 사용자들이 하이라이트 부분뿐만 아니라 주변부에 있는 가사를 질의어로 사용하는 것을 고려하기 위함이다. 이를 위해 본 논문에서는 가우시안 함수(Gaussian)를 가중치 함수로 사용함으로써 하이라이트 부분에서 조금 떨어진 가사가 멀리 떨어진 가사보다 더 중요하도록 만든다.

제안한 방법의 유용성을 확인하고자 한국어로 작성된 22,605개의 노래 가사를 수집하고, Lucene³⁾을 사용하여 노래 가사 검색 시스템을 구축한 후, 사용자 5명을 대상으로 노래 가사 검색을

수행하였다. 질의어 유형에 따른 성능을 보기 위해 총 7가지 질의어를 작성하여 실험을 수행하였으며, n그램으로 작성한 질의어가 가장 좋은 성능을 보였다. 최종적으로 제안한 방법이 하이라이트를 고려하지 않는 검색과 단순히 하이라이트만을 고려한 검색보다 사용자에게 원하는 음악을 찾아줌을 보였다.

본 논문의 구성은 다음과 같다. 2장에서는 음악 검색과 관련된 기존 연구들을 살펴본다. 3장에서는 본 논문에서 제안한 하이라이트 기반 노래 가사 검색에 대해 살펴본다. 하이라이트 부분을 찾기 위한 응집 계층 군집화 및 가중치 함수에 대해 자세히 다룬다. 4장에서는 실험 환경 설정 및 결과 분석에 관해 설명하며, 마지막 5장에서 결론과 향후 연구를 다룬다.

2. 관련연구

음악 검색은 다양한 사용자 요구에 따라 오래전부터 여러 분야에서 연구되어 왔다[2,3]. 음악 검색은 문서 검색을 넘어 다양한 질의 방법 및 입력 내용을 처리하기 위한 연구들이 주로 이뤄져 왔다. 대표적으로 허밍 질의(Query-by-humming)기반의 음악 검색 방법 [4], 예제 질의(Query-by-example) 기반 음악 검색[5,6] 방법이 있다. 이들은 음악을 오디오 시그널로 표현되기 때문에, 시그널 정보를 질의로 하여 질의된 시그널과 가장 유사한 시그널을 찾음으로 원하는 음악을 검색하였다. 반면, 음악을 단순히 문서로 간주하여 일반적인 문서 검색에서 사용한 방법을 확장하여 사용한 연구도 있었다[3,7,8]. 이들은 가사를 질의어로 간주하고, 입력된 질의어와 가장 유사한 문서(음악)를 찾음으로 음악 검색을 수행해왔다. 최근에는 오디오 시그널과 가사를 결합하여 음악 검색을 수행하고자 하는 연구들도 진행되고 있다[9]. 또한 가사를 토픽 모델[10]을 통해 토픽으로 표현한 후, 이를 검색에 사용하고자 한 연구[11] 및 트위터와 last.fm과 같이 의미적 태그와 같은 메타 정보를 사용한 연구들도 동시에 이뤄져 왔다. 음악 검색에 대한 자세한 것은 [12]와 [13]에서 자세히 살펴볼 수 있다.

질의어의 특징을 고려한 가사 검색 관련 연구들은 대부분 질의어에 포함된 발음상의 오류를 처리하고자 하는 연구들이었다. 이들 연구는 가사를 검색할 때 잘못된 기억(mismemorized)이나 잘못된 인식(mishearing)으로 인해 오류가 포함된 질의어가 입력될 때, 이들을 해결하고자 하였다[14,15,16,17]. 기존 연구에서 사용자들이 입력한 질의어 중 약 19% 정도가 발음상 오류가 있음을 보이고, 이를 해결하기 위해 발음 유사도를 측정하여, 유사한 발음을 보정하여 검색을 수행하고자 하였다. 국내에서도 이러한 발음상의 오류를 해결하고자 한 연구들이 제안되어 왔다[18]. 기존 연구들은 질의어를 가사로 사용함에 있어 발음상 오류를 처리하고자 하는

1) <http://www.soundhound.com/>

2) <http://www.shazam.com/>

3) <https://lucene.apache.org/>

반면, 본 연구에서는 질의어 대상인 노래 가사가 가지는 내재적인 특징을 반영하여 가사 검색 시스템의 성능을 올리고자 하는 점이 차이가 있다.

지역적 중요도를 고려하고자 가우시안 함수를 사용한 연구는 POI(Point-of-Interest)의 카테고리를 분류하는 문제에서 진행되어 왔었다[19]. 이 연구에서는 POI 이름 가까이 나온 단어들 POI 카테고리를 결정하는 데 더 중요하다고 간주하였다. 따라서 주어진 문서를 Bag-of-Words로 표현할 때, POI 이름 주변 단어들 그렇지 않은 단어보다 더 중요하도록 가우시안 함수를 사용하였다. 본 연구는 가우시안 함수 기반 지역적 중요도를 고려하여 가사 검색을 수행하고자 한다.

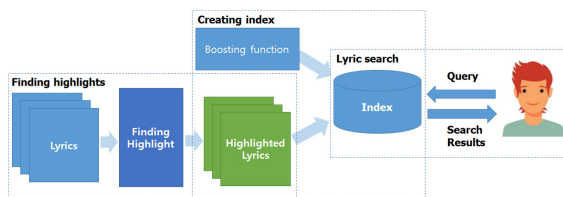


그림 1. 하이라이트 기반 노래 가사 검색의 전체 흐름도
Fig. 1. Overall process of highlight-based lyric search

3. 하이라이트 기반 노래 가사 검색

3.1 하이라이트 기반 노래 가사 검색 흐름도

그림 1은 본 논문에서 제안한 노래 가사 검색 방법을 도식화한 것이다. 먼저, 노래 가사들이 주어질 때, 각 가사마다 하이라이트 부분을 찾는다. 하이라이트 부분이 표시된 가사와 하이라이트 주변부를 고려할 수 있는 가중치 함수를 입력으로 노래 가사 검색을 위한 색인을 구축한다. 본 논문에서는 Lucene을 사용하여 노래 가사 검색 시스템을 구축하였다. 최종적으로 사용자가 찾고자 하는 노래 가사 일부를 질의어로 입력하면, 검색 시스템은 구축된 색인을 통해 가장 적합한 노래 가사를 찾고, 이를 사용자에게 제공한다.

3.2 군집화를 사용한 노래 가사 하이라이트 부분 찾기

본 논문에서는 주어진 노래 가사에서 하이라이트 부분을 찾기 위해 비지도 학습인 군집화(clustering)를 사용한다. 비지도 학습을 사용한 이유는 노래 가사마다 서로 다른 수의 하이라이트를 가지고 있어 규칙 기반 방법의 한계를 극복함과 동시에 많은 수의 학습 데이터를 필요로 하는 지도 학습의 문제점을 해결하기 위함이다. 먼저, 주어진 가사를 군집 단위로 나누고, 군집화를 수행하여 군집들을 얻는다. 이렇게 얻어진 군집들 중 하이라이트 군집을 선택함으로써 그 군집에 속한 가사들을 그 음악의 하이라이트 부분이라 한다.

군집화를 위해 본 논문에서는 가사 중에 한 라인(일반적으로 두 마디 또는 작은 악절, phrase)을 군집 단위로 간주한다. 그림 2는 “김광석”의 “이등병의 편지”가사이다. 그림에서 보듯이, “이등병의 편지”가사에는 총 16개의 라인이 존재하며, 각 라인에 포함된 가사를 하나의 군집 단위로 설정한다.

- 1: 집 떠나와 열차 타고 훈련소로 가는 날
- 2: 부모님께 큰절하고 대문 밖을 나갈 때
- 3: 가슴 속엔 무엇인가 아쉬움이 남지만
- 4: 풀 한 포기 친구 얼굴 모든 것이 새롭다
- 5: 이제 다시 시작이다 젊은 날의 생이며
- 6: 친구들아 군대 가면 편지 꼭 해다오
- 7: 그대들과 즐거웠던 날들을 잊지 않게
- 8: 열차 시간 다가올 때 두 손 잡던 뜨거운
- 9: 기적 소리 멀어지면 작아지는 모습을
- 10: 이제 다시 시작이다 젊은 날의 꿈이며
- 11: 짧게 잘린 내 머리가 처음에는 무섭다가
- 12: 거울 속에 비친 내 모습이 굳어진다 마음까지
- 13: 뒷동산에 올라서면 우리 마을 보일런지
- 14: 나뭇소리 고향의 발 하얗게 퍼지면
- 15: 이등병의 편지 한 장 고이 접어 보내오
- 16: 이제 다시 시작이다 젊은 날의 꿈이며

그림 2. “이등병의 편지”의 가사
Fig. 2. Lyric of “이등병의 편지”

다음으로 응집 계층 군집화 방법을 사용하여 군집화를 수행한다. 그 결과, 가사마다 서로 다른 개수의 하이라이트가 있음을 처리할 수 있다. 군집 단위간의 유사도를 계산하기 위해 본 논문에서는 음절 단위에서 정규화 편집 거리(normalized edit distance)를 사용하였으며, 군집간의 유사도를 측정하기 위해서는 단일결합법(single linkage)을 사용하였다.

응집 계층 군집화는 단순히 군집을 생성할 뿐, 생성된 군집이 하이라이트 군집인지 아닌지를 판단할 수 없다. 본 논문에서는 하이라이트 군집을 찾기 위해 아래의 3가지 휴리스틱 조건을 사용한다.

1. 군집이 둘 이상의 군집 단위를 가지고 있는 경우
2. 군집 단위가 음악 제목을 가지고 있는 경우
3. 군집 단위가 음악 시작 부분인 경우

첫 번째 조건은 후크송(hook-song)에서 주로 볼 수 있는 것으로 유사한 단어들 음악에 전반적으로 반복되는 경우, 이들이 하이라이트가 될 수 있는 조건이다. 이를 통해, 그림 2에서 5, 10, 16 번째 라인이 하이라이트가 됨을 알 수 있다. 두 번째 조건은 가사에 음악 제목이 있는 경우이다. 일반적으로 제목이 음악에 있어 중요한 의미를 가지고 있으며, 가사에 제목이 포함될 경우, 하이라이트로 간주할 수 있다. 예를 들어, 그림 2에서 15번째 라인이 하이라이트가 된다. 마지막으로 세 번째 조건은 가사가 음악 시작 부분인 경우이다. 이는 사용자들이 음악 시작 부분을 주로 기억하기 때문에, 이 부분을 하이라이트로 간주하여 사용자들이 주로 사용하는 질의어와

부합하도록 한다. 예를 들어, 그림 2에서 1번째 라인이 하이라이트가 된다. 최종적으로 하이라이트로 뽑힌 가사와 3.3절에서 제시하는 가중치 함수와 함께 색인을 수행한다.

3.3 가우시안 함수를 통한 가중치 보정

군집화를 통해 찾은 하이라이트 가사에 포함된 단어들은 색인을 구축할 때, 하이라이트 가사에 포함되지 않는 단어들에 비해 더 높은 중요도를 가지도록 한다. 하지만 사용자들은 하이라이트에 포함된 단어뿐만 아니라 하이라이트 주변의 단어들도 질의어로 사용하는 경우가 있다. 따라서 색인을 구축할 때, 주어진 가사의 하이라이트에 포함된 단어뿐만 아니라 하이라이트 부분 주변 단어들에 대해서도 중요도를 반영할 필요가 있다.

본 논문에서는 가우시안 함수로 사용하여 하이라이트 부분 뿐 아니라 주변 단어도 반영할 수 있는 방법을 제안한다. 가우시안 함수는 종 모양을 가진 연속적인 함수(continuous function)로 평균(mean)에서 가장 높은 값을 가지며, 평균에서 멀어질수록 낮은 값을 가진다. 하이라이트 라인을 가우시안 함수의 평균으로 설정함으로써 하이라이트에 인접한 라인에 포함된 단어들은 높은 가중치를 가지는 반면, 그렇지 않은 단어들은 낮은 가중치를 갖게 된다.

현재 주어진 라인을 x 라 하자. x_{h_i} 는 3.2절에서 찾은 하이라이트 가사 집합을, $x_{h_i} \in x_h$ 는 현재 가사에서 i 번째 하이라이트를 나타낸다. $position(x)$ 은 가사에서 라인 x 의 줄 번호를 반환하는 함수라 할 때, 라인 x 에 대해 i 번째 하이라이트에서의 가중치 함수 $f(x; x_{h_i}, \sigma)$ 는 다음과 같이 정의한다.

$$f(x; x_{h_i}, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(position(x_{h_i}) - position(x))^2}{2\sigma^2}} \quad (1)$$

이 때, σ 는 가우시안 함수의 분산을 나타낸다. 분산이 작으면 x_{h_i} 중심에 인접한 라인에 있는 단어들만 영향을 크게 받는다. 반면, 분산이 클수록 먼 라인에 속한 단어들도 하이라이트의 영향을 받도록 한다. 중요한 것은, 라인 x 는 각각의 하이라이트에 대해 다른 중요도 값을 가진다. 따라서 하나의 결과값을 얻기 위해 현재 라인 x 의 가중치인 $boost(x)$ 는 각 하이라이트 라인을 입력으로 얻은 결과 중 가장 높은 값으로 결정한다.

$$boost(x) = \max_{x_{h_i} \in x_h} f(x; x_{h_i}, \sigma) \quad (2)$$

최종적으로 얻어진 가중치와 함께 색인을 구축한다.

4. 실험

4.1 실험 데이터

실험을 위해 본 논문에서는 가사 사이트에서 노래 가사를 직접 수집하였다. 표 1은 수집한 데이터의 통계 정보를 보여준다. 총 22,605곡을 수집하였으며, 수집한 노래 가사 중 가장 긴 라인 개수는 152개이며, 가장 적은 라인의 개수는 3개다. 평균적으로 약 24.9개의 라인을 가지고 있다. 본 실험에서는 형태소 분석 및 특수 문자를 제거하지 않았으며 유일한 단어의 개수는 총 256,501개 이다.

표 1. 수집한 데이터의 통계 정보
Table 1. Simple statistics of data set

Information	Value
Number of total songs	22,605
Maximum number of lines in lyric	152
Minimum number of lines in lyric	3
Average number of lines in lyric	24.90
Number of unique tokens	256,501

표 2. 질의어의 통계 정보
Table 2. Simple statistics of query

Information	Value
Number of users	5
Maximum number of tokens in query phrase	9
Minimum number of tokens in query phrase	2
Average number of tokens in query phrase	5.13

노래 가사 검색을 위해 총 5명의 사용자 각각 10개의 질의어를 입력으로 사용하였다. 표 2는 실험에서 사용한 질의어에 대한 통계정보를 보여준다. 질의어의 길이는 최소 2개의 단어에서 최대 9개의 단어로 작성되어 있으며, 평균적으로 5.13개의 단어를 사용하였다. 그림 3은 5명의 사용자가 실험에 사용한 질의어를 분석한 것이다. 50개의 질의어 중, 9개를 제외한 41개 질의어가 본 논문에서 제안한 하이라이트 부분에서 나왔음을 볼 수 있다. 이를 통해 사용자들이 가사로 음악을 검색할 때, 대부분 하이라이트 부분에서 질의어를 작성하는 것을 볼 수 있다.

주어진 라인에 대한 가중치 함수인 수식 (1)의 파라미터인 σ 를 찾기 위해 5명의 사용자 외에 사용자 1명을 추가로 실험에 사용하였다. 이 사용자 또한 총 10개의 질의어를 사용하였고, 10개 중 8개가 음악의 하이라이트 부분에 해당하였다.

본 논문에서 총 7개의 질의어 타입에 대해 검색을 수행하였다. 표 3은 실험에 사용한 질의어 타입과 “렘블피쉬”의 “그대 내게 다시”에 대한 해당 예제를 보여준다.

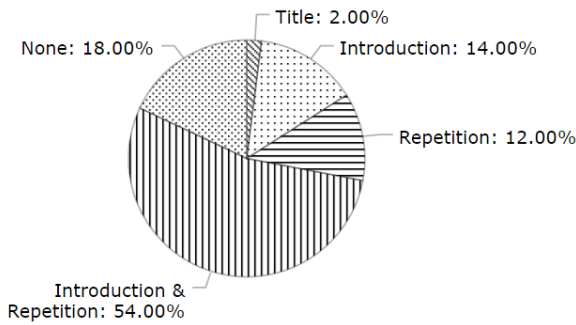


그림 3. 사용자 질의어의 분석
Fig. 3. Details of user queries

표 3. 질의어 타입과 예제
Table 3. Query type and example

Query type	Example
Exact search (EXACT)	“그대 내게 다시”
Exact search with edit distance (EDIT_EXACT)	“그대 내게 다시”~2
Term (TERM)	그대 내게 다시
Term + Exact search (TERM_EXACT)	그대 내게 다시 “그대 내게 다시”
Term + Exact search with edit distance (TERM_EDIT_EXACT)	그대 내게 다시 “그대 내게 다시”~2
N-gram search (NGRAM)	“그대 내게” “내게 다시”
Term + N-gram search (TERM_NGRAM)	그대 내게 다시 “그대 내게” “내게 다시”

질의어 타입으로는 완전 일치 검색(Exact search) 및 편집 거리를 고려한 완전 일치 검색(Exact search with edit distance), 단어 일치 검색(Term), 단어 일치와 완전 일치를 같이 사용한 검색(Term + Exact search), 단어 일치와 편집 거리를 고려한 완전 일치를 같이 사용한 검색(Term + Exact search with edit distance), n그램 검색(N-gram search) 및 단어 일치와 n그램 검색을 같이 사용한 검색(Term + N-gram search)을 사용하였다. 예제에서 따옴표는 완전 일치를, ~는 편집 거리를 의미한다. 편집 거리는 두 음절까지 음절 단위 차이를 고려하였으며, n그램에서 n은 편의상 2로 설정하였다.

제안한 방법을 평가하기 위해 비교 모델로는 하이라이트를 고려하지 않고 색인을 구축한 방법과 가우시안 함수를 통해 보정하지 않고 하이라이트만을 고려하여 색인을 구축하는 방법을 사용하였다. 평가 척도로는 본 논문에서는 Mean Reciprocal Rank (MRR)[20]을 사용하였으며, 다음과 같이 정의된다.

$$MRR = \frac{1}{|Q|} \sum_i |Q| RR_i \quad (3)$$

MRR은 Reciprocal Rank의 평균을 의미하며, i번째 질의어에 대한 Reciprocal Rank인 RR_i 는 다음과 같이 정의된다.

$$RR_i = \frac{1}{rank_i} \quad (4)$$

$rank_i$ 는 i번째 질의어에 대해 시스템 출력한 결과 중, 정답 위치를 나타낸 것이다. 예를 들어, 한 질의어에 대해 시스템이 출력한 결과물 중 정답이 첫 번째에 존재할 때, RR_i 은 1이며, 정답이 두 번째에 존재할 때, RR_i 은 0.5가 된다. MRR은 전체 질의어 집합 Q에 대한 RR_i 의 평균이기 때문에, MRR이 높다는 것은 검색 결과 상위에 정답이 존재함을 의미하며, 이는 검색 결과가 우수함을 나타낸다. 본 논문에서는 총 10개까지 검색 결과를 출력하였으며, 검색 결과에서 정답이 포함되지 않는 경우, RR_i 은 0으로 계산하였다.

4.2 실험 결과

제안한 방법의 성능을 비교하기 전에 σ 에 따른 성능 변화를 살펴보았다. σ 가 1에서 5까지는 성능이 점차적으로 증가하였으며, σ 가 6에서 가장 높은 성능을 보여주며, 그 이후로는 점점 떨어짐을 알 수 있었다. σ 값이 커지면 커질수록 중요도를 고려하지 않고 찾는 방법과 동일하게 된다. 본 논문에서는 σ 를 6으로 설정하여 이후, 실험에서 사용하였다.

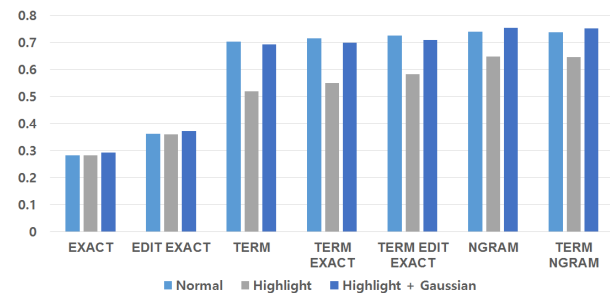


그림 4. 질의어 타입에 따른 실험 결과
Fig. 4. Experimental results according to query types

그림 4는 질의어 타입에 대한 비교 모델과 제안한 방법의 성능을 보여준다.

- 완전 일치 검색이 다른 질의어 타입에 비해 낮은 성능을 보여준다. 실험에 참여한 사용자들이 기억에 의존하여 질의어를 입력하였기 때문에 오류를 가진 질의어로 인해 가사와 질의어와 완전히 일치하지 않는 경우가 많았으며, 그로 인해 낮은 성능을 보인 것으로 분석된다. 이는 Xu[15]의 연구 결과와 유사한 결과를 보여준다.

- 편집 거리를 고려한 검색 결과가 완전 일치 검색보다 향상된 성능을 보여줌을 알 수 있다. 이는 앞선 결과 분석과 동일하게

질의어의 오류를 어느 정도 보완할 수 있음을 보여준다.

- n그램 질의어 타입이 단어 일치 검색에 비해 높은 성능을 얻었음을 볼 수 있다. 일반적인 문서 검색과 달리 가사 검색에서는 사용자가 가사의 일부를 질의어로 사용하기 때문에 단어들의 연속성을 고려한 n그램 질의어 타입이 더 좋은 성능을 얻을 수 있었다.

표 4는 위 실험을 통해 최종 성능을 정리한 것이다. 표에서 보듯이 제안한 방법이 기존 비교 모델들에 비해 0.01에서 0.1만큼 더 높은 MRR을 얻음을 볼 수 있다. 이를 통해, 가사 검색 시, 하이라이트 부분의 가중치를 고려하여야 하며, 추가적으로 하이라이트 부분 주변까지 고려한 것이 가장 좋은 성능을 보임을 알 수 있었다.

표 4. 가사 검색의 최종 MRR
Table 4. Final MRR of lyric search

Method	MRR
Normal (ngram)	0.740
Highlight (ngram)	0.649
Proposed method (ngram)	0.754

4.3 실험 결과 분석

본 논문에서 제안한 방법은 질의어에 포함된 단어에 대하여 비록 노래 가사 내에서 해당 단어의 빈도수가 크지 않지만 하이라이트에 속하는 경우 적절한 음악을 찾아주는 강점을 보였다. 예를 들어, 한 사용자가 “성시경”의 “거리에서”를 찾기 위해, 입력 질의를 “나가 없는 거리에는”이라고 작성하였다. 하이라이트 부분을 고려하지 않는 기존 방법은 “없는데”, “거리에는” 단어가 자주 나타난 “V-One”의 “하루만에” 음악을 상위로 검색하였다. 반면, 제안한 방법은 음악의 하이라이트 부분에 해당하는 단어에 높은 가중치를 주었기에, 질의어에 사용된 단어(“없는데”와 “거리에는”)들이 “성시경”의 “거리에서” 가사에 비록 한번 나왔을지라도 하이라이트에 나왔기에 “V-One”의 “하루만에”보다 상위에 검색되었다.

하지만 여러 노래 가사에서 널리 쓰이는 단어들로 질의어가 작성된 경우, 제안한 방법은 하이라이트를 중점적으로 보기 때문에 비교 모델에 비해 성능이 떨어지는 경우가 있었다. 예를 들어, “박지윤”의 “환상”을 찾기 위해 사용자가 “그대 떠난 걸 헤어졌다는 걸 혼자라는 걸”로 질의어를 입력하였다. 제안한 방법은 질의어의 “그대”, “떠난”과 “걸” 단어가 하이라이트에 자주 나온 “별”의 “슬픈 건망증”을 상위로 검색하였다. 즉, 질의어가 가사에서 널리 쓰이는 단어들로 구성된 경우, 제안한 방법은 하이라이트와 그 주변을 특별히 고려하였기에 몇몇 질의어에 대해 낮은 성능을 보여주었다. 이 경우, 질의 타입을 n그램으로 설정하여 위 문제점을 어느 정도 해결할 수 있었다.

5. 결론 및 향후 연구

본 논문에서는 사용자의 질의어 특징을 반영한 노래 가사 검색 시스템을 제안하였다. 노래 가사 검색에서 사용자들은 음악의 하이라이트 부분에 해당하는 노래 가사를 질의어로 사용한다는 점에 착안하여, 노래 가사 검색 시스템을 구축할 시, 가사의 하이라이트 부분과 더불어 하이라이트 부분 주변 단어들의 가중치를 고려하여 색인을 구축하였다. 실험을 통해, 제안한 방법이 사용자의 질의 유형을 고려하지 않는 모델보다 더 유용함을 보였다. 그뿐만 아니라 단순한 단어 일치 검색이나 완전 일치 검색보다 n그램으로 질의어를 생성한 후, 검색을 수행하는 것이 가사 검색에서 좋은 성능 결과를 보였다.

사용자가 음악의 하이라이트 부분을 기억할 시, 단순히 가사 뿐만 아니라 오디오 시그널 정보에도 영향을 받는다. 향후 연구에는 텍스트 가사 분석의 한계를 넘어 운율이나 코러스 같은 오디오 시그널 정보를 분석함으로써 하이라이트 부분 추출의 성능을 높이고 나아가서 노래 가사 검색 시스템의 검색 성능을 높일 예정이다.

References

- [1] Arnold Schoenberg, Gerald Strang, and Leonard Stein, *Fundamentals of musical composition*, Faber & Faber, 1999.
- [2] Rodger J. McNab, Lloyd A. Smith, Ian H. Witten, Clare L. Henderson, and Sally JoSu Cunningham, “Towards the Digital Music Library: Tune Retrieval from Acoustic Input,” In *Proceedings of the First ACM International Conference on Digital Libraries*, pp. 11-18, 1996.
- [3] J. Stephen Downie, “Music information retrieval,” *Annual Review of Information Science and Technology*, Vol. 37, No. 1, pp. 295-340, 2003.
- [4] Asif Ghias, Jonathan Logan, David Chamberlin, and Brian C. Smith, “Query by humming: musical information retrieval in an audio database,” In *Proceedings of the third ACM international conference on Multimedia*, pp. 231-236, 1995.
- [5] Wei-Ho Tsai, Yu Hung-Ming, and Hsin-Min Wang. “Query-By-Example Technique for Retrieving Cover Versions of Popular Songs with Similar Melodies,” In *Proceedings of 6th International Society for Music Information Retrieval Conference*, pp. 183-190, 2005.
- [6] Avery Wang, “The Shazam music recognition service,” *Communications of the ACM*, Vol. 49, No. 8, pp. 44-48, 2006.

- [7] Zhiyuan Guo, Qiang Wang, Gang Liu, and Jun Guo, "A Music Retrieval System Based On Spoken Lyric Queries," *International Journal of Advancements in Computing Technology*, Vol. 4, No. 8, pp. 173-180, 2012.
- [8] Geun-Tak Chang and Jung-Yun Seo, "Music Recommender System based on Lyrics Information," In *Proceedings of the 22th Annual Conference on Human & Cognitive Language Technology*, pp. 42-45, 2010.
- [9] Tao Wang, Dong-Ju Kim, and Kwang-Seok Hong, "Music Information Retrieval System through Integration of Lyrics and Melody Information," In *Proceedings of the Spring Conference of Korean Society for Internet Information*, pp. 437-442, 2009.
- [10] David Blei, Andrew Y. Ng, and Michael I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, Vol. 3, pp. 993-1022, 2003.
- [11] Shoto Sasaki, Kazuyoshi Yoshii, Tomoyasu Nakano, Masataka Goto, and Shigeo Morishima, "LYRICSRADER: A Lyrics Retrieval System Based on Latent Topics of Lyrics," In *Proceedings of the 15th International Society for Music Information Retrieval Conference*, pp. 585-590, 2014.
- [12] Jung-Soon Ro, "A Comparative Analysis of Content-based Music Retrieval Systems," *Journal of the Korean Society for Information Management*, Vol. 30, No. 3, pp. 23-48, 2013.
- [13] Peter Knees and Markus Schedl, *Music Similarity and Retrieval: An Introduction to Audio- and Web-based Strategies*, Springer-Verlag Berlin Heidelberg, 2016.
- [14] Nicholas Ring and Alexandra L. Uitdenbogerd, "Finding 'Lucy in disguise' : The misheard lyric matching problem," In *Proceedings of the 5th Asia Information Retrieval Symposium*, pp. 157-167, 2009.
- [15] Xin Xu and Tseneo Kato, "Robust and Fast Two-Pass Search Method for Lyric Search Covering Erroneous Queries Due to Mishearing," In *Proceedings of CICLing*, pp. 306-317, 2012.
- [16] Baoxiang Li, Fengxiang Chang, Qiang Wang, Gang Liu, and Jun Guo, "Robust Lyric Search based on Weighted Syllable Confusion Matrix," In *Proceedings of the 21st International Conference on Pattern Recognition*, pp. 613-616, 2012.
- [17] Hussein Hirjee and Daniel G. Brown, "Solving misheard lyric search queries using a probabilistic model of speech sounds," In *Proceedings of the 11th International Society for Music Information Retrieval Conference*, pp. 147-152, 2010.
- [18] Hyungjong Noh, Jeong-Won Cha, and Gary Geunbae Lee, "A Joint Statistical Model for Word Spacing and Spelling Error Correction Simultaneously," In *Proceedings of the ACL 2007 Demo and Poster Sessions*, pp. 61-64, 2007.
- [19] Su Jeong Choi, Hyun-Je Song, Seong-Bae Park, and Sang-Jo Lee, "A POI Categorization by Composition of Onomastic and Contextual Information," In *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 38-45, 2014.
- [20] E. M. Voorhees, "TREC-8 Question Answering Track Report," In *Proceedings of the 8th Text Retrieval Conference*, pp. 77-82, 1999.

저자 소개



김권양(Kweon Yang Kim)

1983년 : 경북대학교 전자공학부(학사)

1990년 : 경북대학교 전자공학부(석사)

1998년 : 경북대학교 컴퓨터공학부(박사)

1983년~1988년 : ETRI 연구원

1999년~2000년 : University of Central Florida

방문교수

1991년~현재 : 경일대학교 컴퓨터공학과 교수

관심분야 : 시멘틱웹, 한글공학

Phone : +82-53-600-5561

E-mail : kykim@kiu.ac.kr