A Natural Language Search Engine for Music driven by Moods

By Dang Trung Thanh

A thesis submitted to
School of Information Science,
Japan Advanced Institute of Science and Technology,
in partial fulfillment of the requirements
for the degree of
Master of Information Science
Graduate Program in Information Science

Written under the direction of
Associate Professor Kiyoaki Shirai

March, 2009

A Natural Language Search Engine for Music driven by Moods

By Dang Trung Thanh (710046)

A thesis submitted to
School of Information Science,
Japan Advanced Institute of Science and Technology,
in partial fulfillment of the requirements
for the degree of
Master of Information Science
Graduate Program in Information Science

Written under the direction of
Associate Professor Kiyoaki Shirai

and approved by
Professor Akira Shimazu
Professor Satoshi Tojo

February, 2009 (Submitted)

Acknowledgements

I would like to thank the following people for their supports and contributions in my thesis. At first, I would like to give thanks to my advisor, Kiyoaki Shirai professor, who gives me inspiration, direction and encourages me through this research. Second, I would like to thanks TIS company, especially Yamamoto san and Tenda san, who supports me during this master course. Finally, I thanks to my family who is always beside me, gives me a great inspiration.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

Nowadays music is playing a more and more important role in human's life, whereas digital catalogs rapidly become larger and more inconvenient to access. If we don't have a good method to explore music, a large amount of music will be fallen into oblivion. Our music search engine integrate two important approaches to explore effectively music collections: searching music by natural language queries and exploring related songs by mood.

1.2 Motivation

1.2.1 Natural language music search engine

There are two major methods to search music on the Internet: searching by text and searching by audio. Almost existing music-searching systems make use of manually assigned subjective meta-information such as genre or style to index the underlying music collections. The intrinsic problem of the metadata-based systems is the limitation to a small set of meta-data, while musical, or more general, cultural context of music pieces is not taken into play. Another approach utilizes audio samples to help users to find interested stuffs. For example, users can mimic melody of the song that they want to find by humming, tapping, or beatboxing to the system. You must have a little talent at music to be able to use such systems.

Nowadays we can find anything by a textual search engine like Google since text can describe almost complicated ideas. Whereas music is recommended everywhere in blog systems which is expanding rapidly with rich information like topic of song, personal emotion about a song or instrument-related information and so on. These kinds of information are always useful for who has no idea about title or artist of songs, who is in a particular context like jogging, falling in love... A system can capture that information and enable searching by natural language queries would be overcome current music search

engines difficulties and help users exploring in every respect of music.

1.2.2 Exploring Music by Moods

Mood as well as topic of a song is very important information to explore music. Human has a habit listening to a song that fit best his current emotion. But not like topic of a song, mood can be learned from content inside a song. This content includes melody and lyric. A grasp of emotions in songs is of great help for us to effectively discover music, thus make our system asymptotically perfect. Imagine that one day you are walking with your girl friend in a cherry blossoms in Japanese park. You would be very happy. So I'm sure that both of you might like listening to happy songs. You can just turn on your iPod, input "happy" mood category and "sakura" query, search and enjoy happy songs together.

1.3 Goal

There are two goals that we aim in this research:

1. Design a music search engine system that enables searching music not only with metadata but also with cultural information.

2. Build a mood classification system using lyric that supports for the music search engine.

There are some previous music search engines before but almost of them can not treat with cultural information. Using Knees approach [56], we build first steps of a base music search engine system that enables searching not only with metadata but also with cultural information.

Mood is a psychological human status that can be affected by music. We can say that music is our close friend since it can share emotion with us, make us happy or sad. That is the main reason why we want to integrate mood exploring ability into our search system. There are two major approaches to solve mood detection problem. The first one, also is the main stream, bases on acoustical data of music. Almost current mood detection researches extract features from acoustical data to discover the mood of a song. The second one use textual data of song like metadata and lyric. Lyric contains almost meaning and mood of a song. However it is not concerned much in current researches. All textual-based researches apply semantic analysis techniques using a common knowledge database to extract emotion from lyric. The supervised machine learning methods are not applied yet because lacking a great enough music dataset tagged with mood. We proposed a method to build such a dataset using LiveJournal blog site. Then, some supervised machine learning methods are applied to detect mood from lyric and metadata on this dataset.

There are two approaches to model emotion: categorical and dimensional. The categorical model often uses a list of basic adjective words as emotion categories. The dimensional model uses some dimensions, each one of which relates to some specific acoustical features. Since our mood classification system bases on textual data, dimensional model is not suitable. Mood clusters in a famous contest about Audio Music Mood Classification (MIREX 07) [64] are used as basic mood categories in our system. We applied some state of the art text categorization methods and proposed new methods to classify moods of songs using lyric and metadata. Three classifiers are applied: SVM, Naive Bayes and Graph-based method.

1.4 Thesis Structure

My thesis is organized in 5 chapters: Introduction, Background, Proposed Methods, Evaluation and Conclusion.

In Chapter 2, we introduce various background knowledge relating to our research. At first, music theory and psychology of music are mentioned to help us understanding the relationship between emotion and music. Then, previous researches about mood detection are reviewed for building our classification system better. Some text categorization methods are also considered to apply for mood classification system. Finally, we take an overview about previous music search engine systems.

In Chapter 3, we describe how to build automatically a natural language music search engine from a music collection. Then, in order to explore the music collection by mood we apply some methods for detecting moods of songs from their lyrics and metadata information.

In Chapter 4, we evaluate methods in Proposed Methods. First, the methods creating datasets are investigated to choose the best dataset for mood detection. Next, we analyze and discuss characteristics of mood detection methods using SVM, Naive Bayes and graph-based classifier.

In Chapter 5, we summarize results achieved in this research and mention some future work.

Chapter 2

Background

In this chapter, we introduce various background knowledge relating to our research. Music theory and relationship between music and emotion are important to extract salient features relating to emotion from music. An accurate understanding of how emotions are represented both in the human mind and in the computer is essential in the design of a mood classification. Some previous researches about mood detection are reviewed for building our classification system better. Some text categorization methods are also considered to apply for classifying mood based on lyric. Beside, a good understanding of information retrieval help us to build a good music search engine system.

2.1 Music Theory

Music theory is the field of study that deals with how music works. It examines the language and notation of music. It identifies patterns that govern composers' techniques. In a grand sense, music theory distills and analyzes the parameters or elements of music – rhythm, harmony (harmonic function), melody, structure, form, and texture. Broadly, music theory may include any statement, belief, or conception of or about music (Boretz, 1995). People who study these properties are known as music theorists. Some have applied acoustics, human physiology, and psychology to the explanation of how and why music is perceived. Music has many different elements. The main elements are: rhythm, melody, harmony, structure, timbre and dynamics.

Melody A melody is a series of notes sounding in succession. The notes of a melody are typically created with respect to pitch systems such as scales or modes. The rhythm of a melody is often based on the inflections of language, the physical rhythms of dance, or simply periodic pulsation. Melody is typically divided into phrases within a larger overarching structure. The elements of a melody are pitch, duration, dynamics, and timbre.

In the context of theory, a piece of music may be melodically based. In this instance, a composer will first take a melody, and use that to create his work. A harmonically based

piece, on the contrary, will focus on a chord progression, with the melody as a secondary or incidental factor of composition [65].

Pitch Pitch is determined by the sound's frequency of vibration. It refers to the relative highness or lowness of a given tone: the greater the frequency, the higher sounding the pitch [65].

Mode Mode is a set of musical notes forming a scale and from which melodies and harmonies are constructed. Early Greek modes include Ionian, Dorian and Hypodorian, Phrygian and Hypophrygian, Lydian and Hypolydian, Mixolydian, Aeolian, and Locrian. The Ionian, Lydian, and Mixolydian modes are of major flavor and the Dorian, Phrygian, Ae-olian, and Locrian modes are of minor descent. Major modes are often associated with happiness, gracefulness and solemnity while minor modes are related to the emotions of sadness, dreaminess, disgust, and anger [41].

Harmony Harmony is the combination of simultaneously sounded musical notes to produce chords and chord progressions having a pleasing effect. Simple harmonies, or consonant chords, such as major chords, are often pleasant, happy, and relaxed. Complex harmonies contain dissonant notes that create instability in a piece of music and activate emotions of excitement, tension, anger, and sadness [41].

Tempo Tempo is defined as the speed at which a passage of music is or should be played, and is typically measured in beats per minute (bpm). A fast tempo falls into the range of 140 to 200 bpm (allegro, vivace, presto) and a slow tempo could be anywhere between 40 and 80 bpm (largo, lento, adagio). Fast tempoes are generally considered lively and exciting, while slow and sustained tempoes are majestic and stately. Depending on other musical factors, a fast tempo can trigger such emotions as excitement, joy, surprise, or fear. Similarly, a slow tempo is typical of calmness, dignity, sadness, tenderness, boredom or disgust [41].

Rhythm The definition of rhythm with respect to emotion is not consistent among various authors, but the most common distinctions include regular/irregular (Watson) [58], smooth/rough (Gundlach) [18], firm/flowing (Hevner) [21], and simple/complex (Vercoe) [56]. Rhythm is officially defined as "the systematic arrangement of musical sounds, principally according to duration and periodic stress"[34]. The features proposed by the aforementioned researchers suggest that variations of the regularity or complexity of a rhythmic pattern in a piece of music trigger emotional responses. Regular and smooth rhythms are representative of happiness, dignity, majesty, and peace, while irregular and rough rhythms pair with amusement, uneasiness, and anger [41].

Dynamic Loudness relates to the perceived intensity of a sound, while dynamics represent its varying volume levels. The dynamics of a piece of music may be either soft or loud. A loud passage of music is associated with intensity, tension, anger, and joy and

soft passages are associated with tenderness, sadness, solemnity, and fear. Large dynamic ranges signify fear, rapid changes in dynamics signify playfulness, and minimal variations relate to sadness and peacefulness [41].

2.2 Psychology of Music

2.2.1 Definition of Emotion

Emotions can usefully be defined as states elicited by rewards and punishments, including changes in rewards and punishments. A reward is anything for which an animal will work. A punishment is anything that an animal will work to escape or avoid. An example of an emotion might thus be happiness produced by being given a reward, such as a pleasant touch, praise, or winning a large sum of money. Another example of an emotion might be fear produced by the sound of a rapidly approaching bus, or the sight of an angry expression on someone's face. We will work to avoid such stimuli, which are punishing. Another example would be frustration, anger, or sadness produced by the omission of an expected reward such as a prize, or the termination of a reward such as the death of a loved one. Another example would be relief, produced by the omission or termination of a punishing stimulus such as the removal of a painful stimulus, or sailing out of danger. These examples indicate how emotions can be produced by the delivery, omission, or termination of rewarding or punishing stimuli, and go some way to indicate how different emotions could be produced and classified in terms of the rewards and punishments received, omitted or terminated [49].

Humans are capable of experiencing a vast array of emotional states. There exist many terms and definitions of emotion as it relates to everyday life. Affect, mood, emotion, and arousal are often used interchangeably though each is unique and differentiable from each other. Emotional states can be broken down into various categories based on how they manifest and exhibit themselves in the individual. An affective state, which is the broadest of emotional states, may have some degree of positive or negative valence, which is the measure of the state's emotional charge. Moods are slightly narrower but provide the basis for more specific emotional states that are typically much shorter in duration and can generally be attributed to a particular stimulus. An emotional state is largely influenced by the underlying mood and affective state of the individual. Thus, an emotional state is often the result of many interrelated and underlying influences, which ultimately manifest themselves visually, through facial expressions, or audibly, through vocalizations and vocal expressions. Lastly, arousal relates to the intensity of an emotional state, similar to how affect and valence trigger positive or negative states in the individual. A highly aroused emotional state will be very apparent in the individual [57].

The subjectivity of emotions creates ambiguity in terminology, and thus emotion remains a vague and relatively undefined area of the human experience. For the most part, everyone understands what an emotion is, and can differentiate amongst them, but when

asked to define what an emotion is they hesitate [14].

2.2.2 Emotion Model

Models of human emotion are diverse and varied owing to the subjective nature of emotion. The two major approaches to emotional modeling that exist in the field today are categorical and dimensional. Each type of model helps to convey a unique aspect of human emotion and together such models can provide insight into how emotions are represented and interpreted within the human mind. When using an emotion model for the use of automated emotion detection in music, one has to pay attention to three aspects [60]:

- The model has to be a good representation of reality
- The model should contain emotions which are often evoked by music
- The model should contain one or more dimensions on which the emotions are measured

A categorical approach is one that consists of several distinct classes that form the basis for all other possible emotional variations. Categorical approaches are most applicable to goal-oriented situations. The most common of the categorical approaches to emotion modeling is that of Paul Ekman's basic emotions, which encompasses the emotions of anger, fear, sadness, happiness, and disgust[12].

A dimensional approach classifies emotions along several axes, such as valence (pleasure), arousal (activity), and potency (dominance). Thayers model of mood (Figure 2.1) offers a simple but quite effective model for moods [58]. Along the horizontal axis the amount of stress is measured and along the vertical axis the amount of energy. In music one can think of energy as the volume or the intensity of sound. Stress can be translated to "having to do too many things" so the difference in tonality and tempo would be a good mapping.

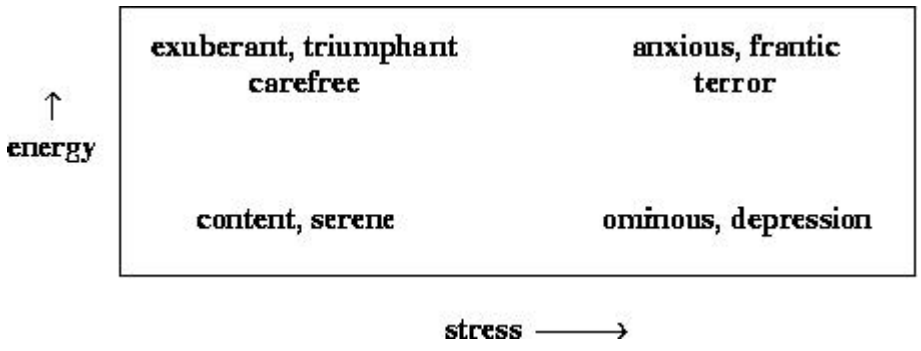

Figure 2.1: Thayers model of mood

Another mood model is the Tellegen-Watson Clark model of mood [2]. This model contains a lot of emotions or moods and use the positive/negative affect as one dimension and the pleasantness/unpleasantness versus engagement/disengagement (45 degrees rotated) as the other (Figure 2.2). What the best emotion model is depends on the application

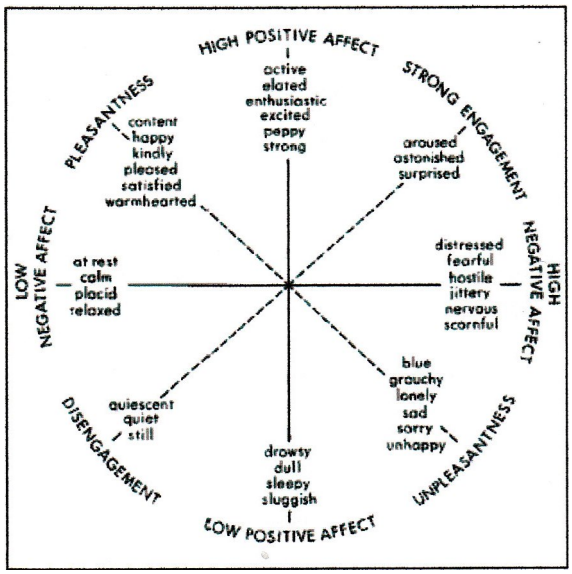

Figure 2.2: Tellegen-Watson Clark model of mood

it is placed in. The number of different emotions and their correlation has its impact on the precision of the method. For some applications (such as the physiotherapist) aren't a lot of emotions necessary as long as the emotion(s) that needs to be evoked is there and the algorithm has a very high recall percentage on that emotion. On the other hand a listener at home that wants to listen to warmhearted music but uses a program with just four emotions will get all happy music. In this case a higher granularity is needed [60].

2.2.3 Music and Emotion

Affections of Melody [41] In the mid 20th century, scholars and researchers such as Hevner, Melvin Rigg, and Karl Watson began to make progress in relating specific musical features, such as mode, harmony, tempo, rhythm, and dynamics (loudness), to emotions and moods. Hevner's studies [17, 18, 19, 20] focus on the affective value of six musical features and how they relate to emotion. The results of these studies are summarized in Figure 2.3. The six musical elements explored in these studies include mode, tempo, pitch (register), rhythm, harmony and melody. These features are mapped to a circular model of affect encompassing eight different emotional categories (Figure 2.4). The characteristic emotions of each of the eight categories are dignified, sad, dreamy, serene, graceful, happy, exciting, and vigorous. Each category contains from six to eleven similar emotions, resulting 67 adjectives in total. This model is closely related to that of Russell [50], and thus provides further validity for the circumplex model of emotion.

Musical element	dignified/ solemn	sad/ heavy	dreamy/ sentimental	serene/ gentle
Mode	major 4	minor 20	minor 12	major 3
Tempo	slow 14	slow 12	slow 16	slow 20
Pitch	low 10	low 19	high 6	high 8
Rhythm	firm 18	firm 3	flowing 9	flowing 2
Harmony	simple 3	complex 7	simple 4	simple 10
Melody	ascend 4	–	–	ascend 3
	graceful/ sparkling	**happy/ bright**	**exciting/ elated**	**vigorous/ majestic**
Mode	major 21	major 24	–	–
Tempo	fast 6	fast 20	fast 21	fast 6
Pitch	high 16	high 6	low 9	low 13
Rhythm	flowing 8	flowing 10	firm 2	firm 10
Harmony	simple 12	simple 16	complex 14	complex 8
Melody	descend 3	–	descend 7	descend 8

Figure 2.3: Hevner's weighting of musical characteristics in 8 affective states [20]

Rigg's experiment includes four categories of emotion; lamentation, joy, longing, and love. Categories are assigned several musical features, for example 'joy' is described as having iambic rhythm (staccato notes), fast tempo, high register, major mode, simple harmony, and loud dynamics (forte) [47, 48]. Watson's studies differ from those of Hevner and Rigg because he uses fifteen adjective groups in conjunction with the musical attributes pitch (low-high), volume (soft-loud), tempo (slow-fast), sound (pretty-ugly), dynamics (constant-varying), and rhythm (regular-irregular). Watson's research reveals many important relationships between these musical attributes and the perceived emotion of the musical excerpt [63]. As such, Watson's contribution has provided music emotion researchers with a large body of relevant data that they can now use to gauge the results of their experiments. Since these initial ground-breaking studies, the field of music and emotion has blossomed into a thriving community whose current researchers include Paul R. Farnsworth, Leonard B. Meyer, Patrik N. Juslin, John A. Sloboda, Emery Schubert, Alf Gabrielsson, Erik Lindstrom, and David Huron, to name a few. The work of these scholars varies from emotional analysis of short musical excerpts to continuous measurement of emotion and its evolution throughout an entire piece of music.

Affections of Lyric Songs are special in that they comprise both melodic and lyrical information. Although the two components can be processed independently (e.g. Bonnel et al. [7]), they are often quite integrated in that recognition of one component is enhanced by the simultaneous presence of the other component (Serafine et al. [53]), and

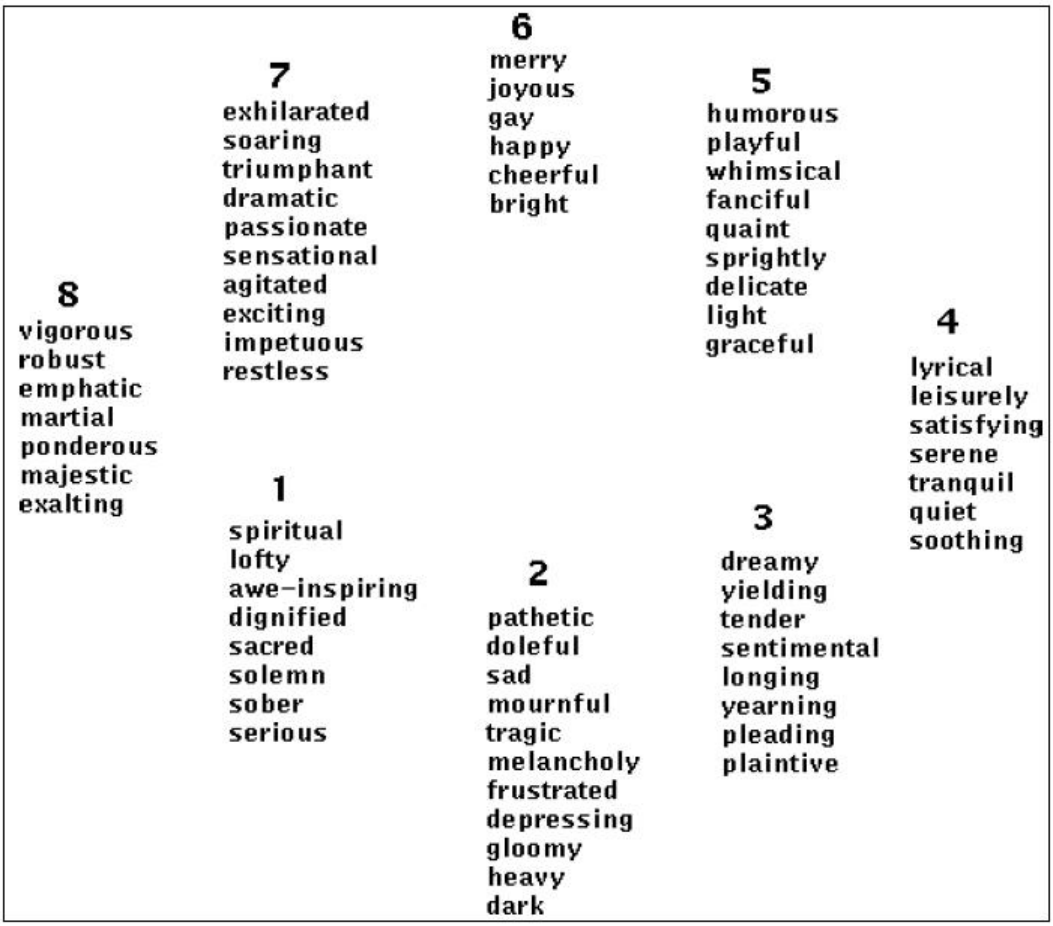

Figure 2.4: Hevner's adjective circle [19]

such integration occurs regardless of the semantic content of the lyrics (Serafine et al. [54]). Indeed, temporal contiguity of melody and lyrics has been shown to be enough to account for the integration (Crowder et al. [11]).

Some studies focusing on the effects of lyrics in popular songs on negative behaviors and attitudes have reported that lyrics do indeed make a difference (e.g. Burt [42]). For instance, Anderson [5], Arnett [25], Hansen and Hansen [8], and Harris et al. [16] have observed positive correlations between increased exposure to hard rock/heavy-metal music with lyrics and more frequent negative behaviors.

Galizio and Hendrick [37] found that a more positive emotional state, as well as an increased acceptance for the message portrayed through the lyrics, occurred when they were coupled with melodies. Additionally, Gfeller and Coffman found that, in trained musicians, music paired with spoken text increased perceived complexity and lowered both liking and affective ratings from baseline compared to just the melody or just the spoken text, leading to the speculation that perhaps training made musicians respond analytically to the music, diminishing their emotional response to it [30]. Thus, in both studies, the presence of lyrics influenced the perception of emotion in songs. The two studies that have explored the effects of melody with and without lyrics on mood more systematically have yielded mixed results. Stratton and Zalanowski have found that the melody of a sad song was perceived as pleasant and had positive effects on mood [62]. However, only lyrics of the sad song and lyric coupled with the melody of the sad song, were perceived as unpleasant and had negative effects on mood. Furthermore, the melody of the sad song played in an up-beat tempo was perceived as pleasant and had positive effects on mood, whereas the lyrics coupled with the melody of a sad song and played in an up-beat tempo was perceived as unpleasant and had negative effects on mood. Finally, the melody of the sad song coupled with 'positive' lyrics and played in the original (i.e. slow) tempo was perceived as pleasant and had positive effects on mood. Thus, Stratton and Zalanowski concluded that the lyrics of a song have a greater ability to affect mood than does the melody.

Sousou also examined the effects of music with and without lyrics on mood [52]. In this study, participants read either happy or sad lyrics while listening to happy, sad, or no music, and rated their mood and psychological arousal before and after the presentation of the stimuli. Participants who listened to the sad melody rated their mood as sad, and participants who listened to the happy melody rated their mood as happy, regardless of the type of lyrics that they read. Participants who read the lyrics while listening to no music rated their mood as neutral. Furthermore, participants reported increased arousal from their baseline ratings while listening to the happy melody, regardless of the type of lyrics that they read, compared to those participants who listened to the sad melody or no music. Hence, even though the lyrics were presented visually and the melodies auditory, which may have inflated the importance of the melody, Sousou concluded that, unlike Stratton and Zalanowski's findings [62], the melody of a song has a greater ability

to affect mood than its lyrics.

Omar et al. explored whether lyrics and melodies of songs were equal partners in their effectiveness in conveying emotions, and how they affected each other [4]. Although intuitively one might have predicted that lyrics conveying the same emotion as the melodies would enhance the overall emotion, they found that this happened only for negative emotions. Lyrics indeed bolstered the emotion conveyed by sad or angry music. However, unexpectedly, lyrics detracted from the emotion elicited by happy or calm music, that is, positive emotions. Thus, it appears that lyrics can indeed influence the overall emotional valence of music, allowing music to more easily convey negative emotions when they are present, and allowing music to more easily convey positive emotions when they are absent. In sum, the melody of music was more dominant than the lyrics in eliciting emotional responses.

Do lyrics hold the key to improving the performance of emotion-based music information retrieval (EMIR)? While mainstream EMIR research focuses on analyzing a song's melody, Daniel et al. aims to explore the influence of lyrics vs. melody [24]. They conducted a user study to gather subjects emotion ratings on lyrics and melody, and applied statistical analysis to show how each contributes to the song's overall Valance-Arousal (V-A) emotion level. Their results show that lyrics are not only a valid measure for emotion estimation of a song, but they also provide supplementary information that can improve a melody-centric EMIR system. In addition, their data suggest that the correlation between lyrics and melody depends on the V-A quadrant in which the song resides.

2.3 Mood Detection in Music

2.3.1 Applications of Emotion Detection in Music

Huron gives some examples of different applications of emotion detection methods in music [22]. The following list is based upon the Huron's list, but extra possible applications are and more explanation are given. To give an idea of how broad the applications could be [60]:

- The owner of a trendy shop who wants to seek music that attracts a certain clientele.
- An aerobics instructor who seeks a certain tempo for his or her workout.
- A film director who seeks music evoking a certain mood which matches the images on screen. In this way the person watching the movie will be totally grasped by the scene.
- An advertiser seeking a tune that is highly memorable or that evokes a positive emotion towards a certain product. Presumably happy and positively charged music.

- A physiotherapist who seeks music that will motivate a patient while doing recovery exercises. (e.g. Survivor - Eye of the Tiger)
- A call center that receives inbound calls that has to put callers on hold will want to give their users happy music. Often very old and typical music is used, this can be improved by an application that searches for happy music in recent music.
- Working personnel who seek music that will keep them alert. This will be mostly cheerful or arousing music.
- A DJ who seeks music that will has the same key as the current song or approximately the same beat so that the people on the dance floor notice as little as possible from mixing two songs.

There are a number of issues emotion detection methods [60]. There always will be a certain compromise between various issues. These issues differentiate the various emotion detection methods.

Precision One of the most obvious criteria for a good emotion or mood detection algorithm is the precision or accuracy that is achieved. An algorithm gives some sort of output. This output depends on the algorithm used. Some give just one emotion, while others give two or more emotions with their reliability, e.g. 70% emotion1 and 30% emotion2. Such output will be compared with the annotated emotions evoked by a human subject. The percentage of 'right answers' determines the accuracy.

Granularity In a strong relationship with the above, granularity of mood classes has a very big influence on the precision achieved. This is a quite logical phenomenon because when one has to choose between more options, there is a larger chance that a wrong one is chosen. Low granularity (e.g. 4 emotions) can be useful but that will depend on the application.

Diversity Some papers only use a limited number of songs or just one or two genres of music. This will have its influence on how much the algorithm can be optimized for that particular genre. However, this gives an unfair advantage when comparing the accuracy with other methods whose algorithm will be usable for more or even all sorts of music. These optimizations can be useful in certain applications.

Mobile use Implementation of emotion detection on mobile devices gives a couple of restrictions to the methods. On a mobile device the resources are typically more limited than on a PC. Hard disk or flash space, computing power and total amount of RAM memory are limited because of cost and size. None of the reviewed methods mentions how much computing power for instance is needed to work trough their test corpus. Adversely mobile devices are getting more advanced every few months. So if the resources on mobile devices aren't sufficient today, they probably will be within a few years. Another facet

to this issue is the nature of use of a mobile device. People will want to use the device and its "emotion drive selection search" right away and probably do not want train it for multiple hours on the music they just put on it.

Learning The learning implemented in the algorithms is essential for getting good results. The algorithm has to learn which features linking to a certain emotion are. These features can be different for persons from different cultural backgrounds. Also other genres of music than tested can be used by a user. When a method is implemented into software that can be used on a PC or a mobile device one has to think at the trade off between giving the user software with a standard database and letting the user train the algorithm so that its accuracy will improve. Not all users will be patient to train a program with hours of music. On the other hand, software that is too inaccurate will not be used.

2.3.2 Acoustical Approaches

Automatic emotion detection and extraction in music is growing rapidly with the advancement of digital signal processing, audio analysis and feature extraction tools. As a fledgling field, the feature extraction methods and emotional models used by its proponents are varied and difficult to compare; however, these first small steps are important in forming a basis for future research. Moreover, mood detection in music is beginning to be seen as a relevant field of music information retrieval and promises to be an effective means of classifying songs.

One of the first publications on emotion detection in music is credited to Feng, Zhuang, and Pan. They employ Computational Media Aesthetics to detect mood for music information retrieval tasks [67]. The two dimensions of tempo and articulation are extracted from the audio signal and are mapped to one of four emotional categories; happiness, sadness, anger, and fear. This categorization is based on both Thayer's model [58] and Juslin's theory [29], where the two elements of slow or fast tempo and staccato or legato articulation adequately convey emotional information from the performer to the audience.

Another integral emotion detection project is Li and Ogihara's content-based music similarity search [35]. Their original work in emotion detection in music [34] utilized Farnsworth's ten adjective groups [13]. Li and Ogihara's system extracts relevant audio descriptors using MARSYAS [59] and then classifies them using Support Vector Machines (SVM). Their 2004 research [35] utilized Hevner's eight adjective groups to address the problem of music similarity search and emotion detection in music. Daubechies Wavelet Coefficient Histograms are combined with timbral features, again extracted with MARSYAS, and SVMs were trained on these features to classify their music database.

Implementing Tellegen, Watson, and Clark's three-layer dimensional model of emotion [2], Yang and Lee developed a system to disambiguate music emotion using software agents [68]. This platform makes use of acoustical audio features and lyrics, as well as cultural

metadata to classify music by mood. The emotional model focuses on negative affect, and includes the axes of high/low positive affect and high/low negative affect. Tempo is estimated through the autocorrelation of energy extracted from different frequency bands. Timbral features such as spectral centroid, spectral roll off, spectral flux, and kurtosis are also used to measure emotional intensity. The textual lyrics and cultural metadata helped to distinguish between closely related emotions.

Alternatively, Leman, Vermeulen, De Voogdt, and Moelants employ three levels of analysis, from subjective judgments to manual-based musical analysis to acoustical-based feature analysis, to model the affective response to music [38]. Their three-dimensional mood model consists of valence (gay-sad), activity (tender-calm), and interest (exciting-boring). The features of prominence, loudness, and brightness are present along the activity axis, while tempo and articulation contribute to varying degrees of valence. The axis of interest is not clearly defined by any features.

Wang, Zhang, and Zhu's system differs slightly from the aforementioned models in that it analyzes symbolic musical data rather than an acoustic audio signal [40]. However, the techniques used are still relevant with respect to emotion detection in music. The user adaptive music emotion recognition system addresses the issue of subjectivity within mood classification of music. This model employs Thayer's two-dimensional model of emotion with some modifications. Both statistical and perceptual features are extracted from MIDI song files, including pitch, intervals, tempo, loudness, note density, timbre, meter, tonality (key and mode), stability, perceptual pitch height, and the perceptual distance between two consecutive notes. SVMs were then trained to provide personally adapted mood-classified music based on the users opinions.

Another implementation of Thayer's dimensional model of emotion is Tolos, Tato, and Kemp's mood-based navigation system for large collections of musical data [39]. In this system a user can select the mood of a song from a two-dimensional mood plane and automatically extract the mood from the song. Tolos, Tato, and Kemp use Thayer's model of mood, which comprises the axes of quality (x-axis) and activation (y-axis). This results in four mood classes, aggressive, happy, calm, and melancholic. A twenty-seven-dimension feature vector is used for the classification of the audio data. This vector contains cepstral features, power spectrum information, and the signal's spectral centroid, rolloff, and flux. The authors conclude from the results of their studies that there are strong inter-human variances in the perception of mood and different perceptions of mood between cultures. They also deduce that the two-dimensional model is well suited to small portable devices as only two one-dimensional inputs are required.

Building on the work of Li and Ogihara, Wieczorkowska, Synak, Lewis, and Ras conducted research to automatically recognize emotions in music through the parameterization of audio data [3]. They implemented a k-NN classification algorithm to determine the mood of a song. Timbre and chords are used as the primary features for parameterization.

Their system implements single labeling of classes by a single subject with the idea of expanding their research to multiple labeling and multi-subject assessments in the future. This labeling resulted in six classes: happy and fanciful; graceful and dreamy; pathetic and passionate; dramatic, agitated, and frustrated; sacred and spooky; and dark and blue.

A third system to employ the psychological findings of Thayer is that of Lu, Liu and Zhang, who introduced a method for automatically detecting and tracking mood in a music audio signal [32]. They created a hierarchical framework to extract features from the acoustic music data based on Thayer's psychological studies [58]. This model classifies an emotion on a two-dimensional space as either content, depressed, exuberant, or anxious/frantic. Intensity, timbre, and rhythm are extracted and used to represent the piece of music. The first feature, intensity, is measured by the audio signal's energy in each sub-band. Timbre is represented by the spectral shape and contrast of the signal, and rhythmic information is gauged by its regularity, intensity, and tempo. This model also implements mood tracking, which accounts for changing moods throughout a piece of music. In the hierarchical model, intensity is first used as a rough measure of mood in each section of a piece of music, and timbre and rhythm are later applied to more accurately define each section's mood space.

Less focused on the issue of the actual emotion detection in music, Skowronek, McKinney, and van de Par focused on discovering a ground truth for automatic music mood classification [27], which classified musical excerpts based on structure, loudness, timbre, and tempo. Russell's circumplex model of affect was used in conjunction with nine bipolar mood scales. They found that "easy to judge" excerpts were difficult to determine, even by an experienced listener. In terms of affective vocabulary, they surmised that the best labels were tender/soft, powerful/strong, loving/romantic, carefree/lighthearted, emotional/passionate, touching/moving, angry/furious/aggressive, and sad.

Lastly, an emerging source of information relating to emotion detection in music is the Music Information Retrieval Evaluation eXchange's (MIREX) annual competition, which will for the first time include an audio music mood classification category. This MIR community has recognized the importance of mood as a relevant and salient category for music classification. They believe that this contest will help to solidify the area of mood classification and provide valuable ground truth data. At the moment, two approaches to the music mood taxonomy are being considered. The first is based on music perception, such as Thayer's two-dimensional model. It has been found that fewer categories result in more accurate classifications. The second model comes from music information practices, such as All Music Guide and MoodLogic, which use mood labels to classify their music databases. Social tagging of music, such as Last.FM, is also being considered as a valuable resource for music information retrieval and music classification.

Although above researches showed that acoustical features are effective in mood detection, this approach still has some limitations. The first one is audio data of each song

takes much time to process. Thus, for a big system like our search engine, applying this approach is not practical. The second one is copyright problem. In order to build a trust able mood classification system, we need a big enough ground-truth dataset. Such kind of dataset is not easy to collect because of copyright problem, whereas lyrical data, which is comparative to acoustical data in mood detection and especially available free on the Internet, is not concerned much. That why we decided to build a mood classification system based on lyrical data.

2.3.3 Textual Approaches

Various methods have been applied to music mood detection(MMD). Most mainstream research focused on melody-based methods as introduced in the section 2.3.2. Meanwhile, lyrics have seldom played a significant role in MMD probably because of its subtleness described in [6]. Nevertheless, in recent decades, technology advancement in computational linguistic supported textual mood detection can be retrieved with less effort and higher precisions. [41, 21] demonstrate some approaches using lyrics to estimate the emotional level of songs.

In [21], emotion model of a given song is analyzed by textual analysis with commonsense [36] on lyrics and metadata. In this approach, lyrics can be viewed as a concrete, implicit expression of an abstract concept, which often makes listeners catch the feeling of a song even without listening to it actually. Given that lyrics are written in natural language by people, its content can be analyzed with the help of some commonsense knowledge base. Commonsense knowledge spans a huge portion of human experience, encompassing knowledge about the spatial, physical, social, temporal, and psychological aspects of typical everyday life.

Meyer thesis [41] used the guess mood function, included in ConceptNet's natural language tools to extract salient emotional concepts and words from a song's lyrics. Its output has been modified to reflect Russell's dimensional emotional model [50] rather than guess mood's original categorical model. The affective value of the lyrics is then used in conjunction with the audio features to classify the song by mood.

The two above researches have just tried some first steps using ConceptNet [36] toolkit to extract mood from lyric. This is a unsupervised learning approach. They just use available affective sensing ability of ConceptNet, but have not investigated thoroughly lyric's features to detect mood. Our research applies the supervised learning approach and considers in every aspect of lyric for mood classification.

2.4 Natural Language Processing

2.4.1 Information Retrieval for Music

There are two main groups of MIR systems for content-based searching can be distinguished, systems for searching audio data and systems for searching notated music. There are also hybrid systems that first convert audio signal into a symbolic description of notes and then search a database of notated music [46].

Several methods for retrieving music from large databases have been proposed. Most of these approaches use query-by-example methods. Thus, the query must consist of a piece of information that has the same representation as the records in the database. For example, in Query-by-Humming/Singing (QBHS) systems [1] the user has to sing or hum a part of the searched piece into a microphone. In most cases, these systems operate on a symbolic representation of music (i.e. MIDI).

A more challenging task is to design systems that enable cross-media retrieval. These systems allow queries consisting of arbitrary natural language text, e.g. descriptions of sound, mood, or cultural events, and return music pieces that are semantically related to this query, are of interest. Unfortunately, the number of systems enabling its users to perform such queries is very little. The most elaborate approach so far has been presented by Baumann et al. [51]. Their system is supported by a semantic ontology which integrates information about artist, genre, year, lyrics, and automatically extracted acoustic properties like loudness, tempo, and instrumentation and defines relations between these concepts. Beside they also do the mapping of the query to the concepts in the ontology. In the end, the system allows for semantic queries like "something fast from..." or "something new from...".

Knees et al. [56] proposed an approach to automatically build a search engine for large-scale music collections that can be queried through natural language. While existing approaches depend on explicit manual annotations and meta-data assigned to the individual audio pieces, they automatically derive descriptions by making use of methods from Web Retrieval and Music Information Retrieval. They use ID3 of mp3 files to retrieve relevant Web pages via Google queries and use the contents of these pages to characterize the music pieces and represent them by term vectors. By incorporating complementary information about acoustic similarity they are able to both reduce the dimensionality of the vector space and improve the performance of retrieval.

In [43], Celma et al. present the music search engine Search Sounds (www.searchsounds.net). The system uses a special crawler that focuses on a set of manually defined "audio blogs", which can be accessed via RSS links. In these blogs, the authors explain and describe music pieces and make them available for download. Thus, the available textual information that refers to the music, together with the meta-data of the files, can be used to match text queries to actual music pieces. Furthermore, acoustically similar pieces can

be discovered by means of content-based audio similarity for all returned results.

Another system that opts to enhance music search with additional semantic information is Squiggle [23]. In this system, queries are matched against meta-data and also further evaluated by a word sense disambiguation component that proposes related queries. For example, a query for "rhcp" results in zero hits, but suggests to search for the band "Red Hot Chili Peppers". The underlying semantic relations are taken from the freely available community databases MusicMoz (www.musicmoz.org) and MusicBrainz (www.musicbrainz.org). The system depends on explicit knowledge which is in fact a more extensive set of manually annotated meta-data.

A system that is not restricted to a pre-defined set of meta-data is Last.fm. Last.fm integrates into music player software and keeps track of each user's listening habits. Based on the collected data, similar artists or tracks can be recommended. Additionally, users can assign tags to the tracks in their collection. These tags provide a valuable source of information on how people perceive and describe music. A drawback of the system is that most tags are highly inconsistent and noisy.

Beside music information systems that deal solely with popular music, there exist a number of search engines that use specialized crawlers to find all types of sounds on the Web. The traced audio files are indexed using contextual information extracted from the text surrounding the links to the files. Some examples of such systems are Arooga [31] and FindSounds (www.findsounds.com).

2.4.2 Text Categorization

Text categorization is the problem of automatically assigning one or more predefined categories to free text documents. Since more and more textual information is available online, effective retrieval of documents is difficult. Document categorization is one solution to this problem. A growing number of statistical classification methods and machine learning techniques have been applied to text categorization in recent years. In this subsection, we introduce three machine learning methods that are applied in our system: Support Vector Machine (SVM), Naive Bayes and Graph-based Classification.

2.4.2.1 Support Vector Machine

Support Vector Machines (SVMs) have shown to yield good generalisation performance on a wide variety of classification problems. The SVM integrates dimension reduction and classication. It is only applicable for binary classication tasks, meaning that, using this method text categorization has to be treated as a series of dichotomous classification problems.

The SVM classifies a vector d to either -1 or 1 using

$$s = w^T \phi(d) + b = \Sigma_{i=1}^{N} \alpha_i y_i K(d, d_i) + b \tag{2.1}$$

and

$$y = \begin{cases} 1 & if s > s_0 \\ -1 & if otherwise \end{cases} \tag{2.2}$$

where d_i is the set of training vectors and y_i are the corresponding classes ($y_i \in -1, 1$). $K(d, d_j)$ is denoted a kernel and is often chosen as a polynomial of degree d, i.e.

$$K(d_i, d) = (d^T d_i + 1)^d \tag{2.3}$$

The training of the SVM consists of determining the w that maximizes the distance between the training samples from the two classes.

2.4.2.2 Naive Bayes

The Naive Bayes classier [28] is constructed by using the training data to estimate the probability of each class given the document feature values of a new instance. The Bayes theorem is applied to estimate the probabilities:

$$P(c_i|d) = \frac{P(d|c_i) \times P(c_i)}{P(d)} \tag{2.4}$$

The denominator in the above equation does not differ between categories and can be ignored. So we have:

$$c_{select}(d) = \arg\max_{c_i \in C} P(c_i) \times P(d|c_i) \tag{2.5}$$

Here $P(c_i)$ is estimated from training data as follow:

$$P(c_i) = \frac{N_i}{N} \tag{2.6}$$

where N_i is the number of training documents assigned with c_i label, N is the total number of training documents.

Each document is a bag-of-words in which we assume that a word's occurrence is only dependent on the class the document comes from, but that it occurs independently of the other words in the document. So $P(d|c_i)$ can be estimated as follows:

$$P(d|c_i) = \Pi_{j=1}^{|F|} P(w_j|c_i)^{TF(w_j,d)} \tag{2.7}$$

where F is the feature vector of the document d. $TF(w_j, d)$ is the frequency of word w in the document d. $P(w_j|c_i)$ can be estimated by using addictive smoothing technique as follows:

$$P(w_j|c_i) = \frac{1 + O(w_j, c_i)}{|F| + \Sigma_{w \in F} O(w, c_i)} \tag{2.8}$$

where $O(w_j, c_i)$ is the occurrence frequency of the word w_j and the category c_i.

Despite the fact that the independent assumption is generally not true for word appearance in documents, the Naive Bayes classifier is surprisingly effective.

2.4.2.3 Graph-based Classification

In many settings, the "context-free" approach (i.e. SVM, Naive Bayes, ..) does not exploit the available information about relationships between data items. For example, if we want to classify a song into a topic, we can consider some additional information like the topic of other songs composed by the same artist. Using the relationship information, we can construct a graph G in which each data item (e.g., Web page) is a node and each relationship instance (e.g., a hyperlink) forms an edge between the corresponding nodes. Then the classification problem can be formulated as a graph labeling or coloring problem on such a graph. In the following, we introduce some approaches solving this problem.

Kleinberg and Tardos [26] views the classification problem for nodes in an undirected graph as a metric labeling problem where we aim to optimize a combinatorial function consisting of assignment costs and separation costs. The assignment costs are based on the individual choice of label we make for each object while the separation costs are based on the pair of label choices we make for two neighboring objects. The combination of the assignment and the separation costs gives the total cost. We need a labeling that minimizes the total cost.

S. Chakrabarti et al. [9, 10] propose to start with a greedy labeling of the graph instead of seeking a global optimization, paying attention only to the node-labeling (assignment) cost, and then iteratively "correct" the neighborhood labeling where the presence of edges leads to a very high penalty in terms of the separation costs.

In Oh et al. [55], authors present a "single step" approach in which the label of each node d in the graph is influenced by the popularity of this label among all immediate neighbors of d and the level of confidence in the labels of the documents in the neighborhood. The term weight w_t for any document is adjusted using the term frequencies in the neighboring documents and a parameter that controls the degree of influence.

Lu and Getoor [44] proposed a regression model which combines the text features for every given test document with the label assignments of its neighbors and iteratively tries to improve the classification result. Authors proposes three ways to construct a feature vector for each document. In the Binary mode, a document feature vector includes its "text" features and their weights, and additional new features derived from the underlying link structure. These features are every class label $c_i \in C$ that appears at least once in the neighborhood of d, $N(d)$. In the Count mode, the document feature vector is enhanced with the neighbors' labels c_i weighted by their frequencies in $N(d)$. In the Single mode, only the most popular class label is included into the feature vector of document d, with weight equal to its frequency in $N(d)$.

Angelova and Weikum [45] presents a new method for graph-based classification, with particular emphasis on hyperlinked text documents but broader applicability. Their approach based on iterative relaxation labeling and can be combined with either Bayesian

or SVM classifiers on the feature spaces of the given data items. The graph neighborhood is taken into consideration to exploit locality patterns while at the same time avoiding overfitting. Their techniques considerably improved the robustness and accuracy of the classification outcome compare to previously published methods.

Chapter 3

Proposed System

In this chapter, we describe how to build automatically a natural language music search engine from a music collection using Peter Knees et al. approach [56]. Then, in order to explore the music collection by mood we apply some methods for detecting moods of songs from their lyrics and some metadata information.

3.1 Natural Language Search Engine for Music

In this section, the overview of our system is described. Figure 3.1 showed the architecture of our system. There are four major components in my system: Information Collector, Mood Detection, Music Index Builder and Interactive Agent. At first, Information Collector uses metadata information of each song to collect related Web pages from Internet. Next, Mood Detection detects mood of each song in the song collection. After that, Music Index Builder extracts music-related information from collected Web pages and also use songs tagged with mood to create the music database. Now, users can interact with the system by Interactive Agent module. This module tasks are parsing user's queries semantically, retrieving corresponding songs and ranking result. For example, when you feel sad and want to hear some songs of Eric Clapton, you can use our system to find your desiring songs. From the system's interface, you just select a sad mood from a predefined mood categories and input a query "Eric Clapton". Or, one day you are walking along the maple road in JAIST in autumn. This beautiful sight might make you want to listen to autumn songs. You can turn on your iPod and search with the query "Japan autumn".

3.1.1 Information Collector

The task of this module is collecting related Web pages of each song from Internet by metadata information. First, we collect about 1,792 songs from our personal music libraries. After removing all songs which lack one of the three metadata fields: artist, title and album, our collection has about 1,300 songs. We use the Yahoo search engine API for finding information relating to each song from Internet. There are three types of information that we want to retrieve: artist, album and title of song. Finally, we collected

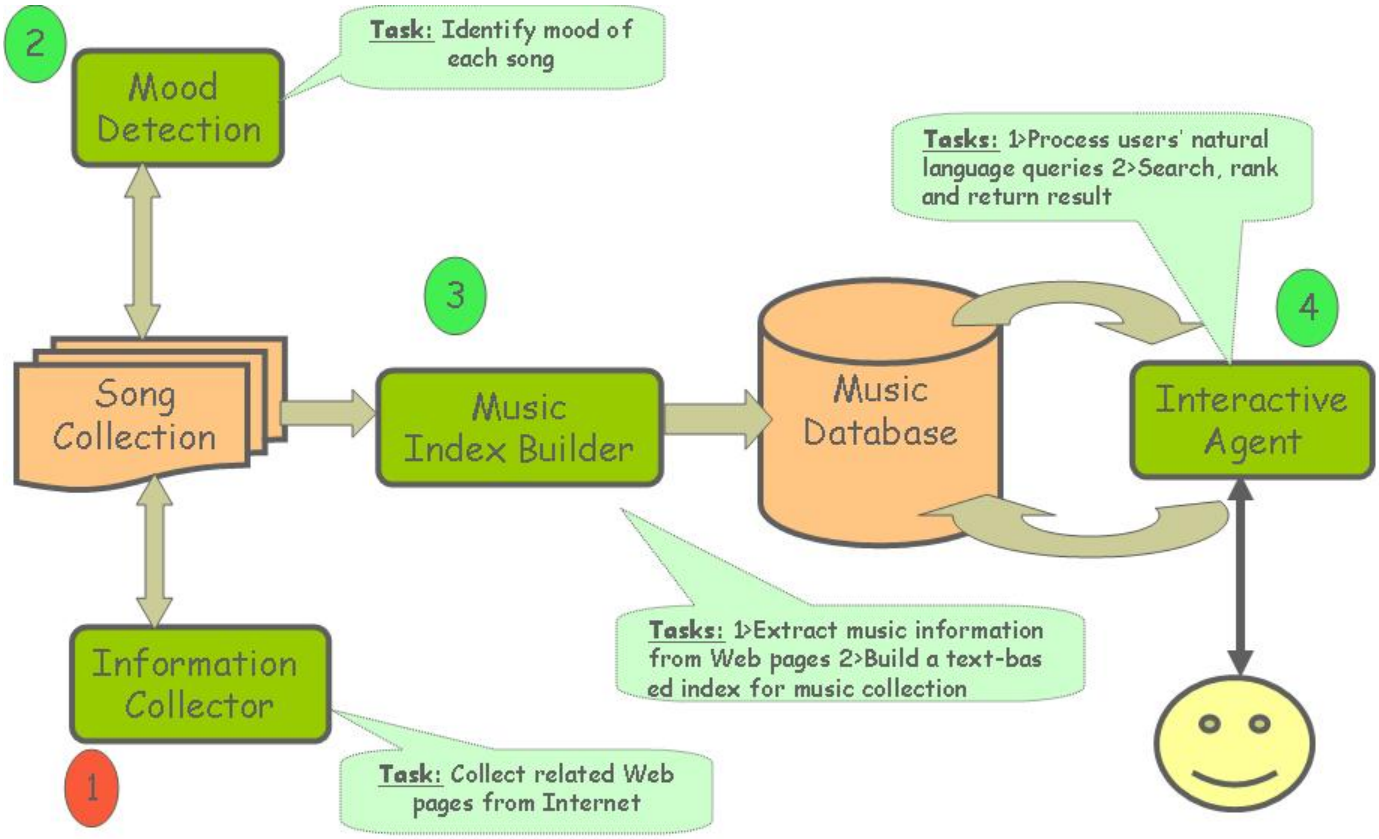

Figure 3.1: Music Search Engine System

about 58,000 Web pages for 1,300 songs in Song Collection.

3.1.2 Mood Detection

Currently, using metadata as queries are the major way to search music. Almost existing music search engine systems make use of manually assigned subjective meta-information to index the underlying music collections. There are three major types of metadata for a song that users can use to explore music stores: artist, genre and album. However moods or emotions of songs are not concerned much in current systems. We can see that human often wants to listen to music that fits best his current emotion. A grasp of emotions in songs might be a great help for us to effectively discover music. In this module, moods of songs are automatically identified based on lyrics and metadata. In this thesis, the development of this module is the main goal, and we proposed several methods for supervised learning of classifiers. Then, we use automatically identified moods of songs as metadata in our music search engine presented in Figure 3.1.

3.1.3 Music Index Builder

This module (MIB) is our future work. As mentioning before, this module's task is extracting music information from Web pages and than mapping this information to each song. As you know, there are many types of information written on each Web page. The task of MIB is to detect all words, sentences or even phrases relating to music. Namely, this kind of information can be music metadata fields like artist name, title of song or album name. We are intending to apply mining methods and the Probabilistic Latent Semantic Indexing technique to reveal underlying meaning in Web pages.

3.1.4 Interactive Agent

This module (IA) is also our future work. There are two kinds of problem we have to solve with IA. The first one is how to understand queries semantically in the music domain. Two types of semantic that we want to concentrate here are mood of query and music information. The second one is how to rank the song result in which song similarity (based on both acoustical and textual) and song popularity are top priorities.

3.2 Mood Detection

In this section, we will present about three works:

- How to define mood categories (in Subsection 3.2.1)
- How to construct a training dataset for mood detection (in Subsection 3.2.2)
- How to decide the mood of a song (in Subsection 3.2.3)

3.2.1 Mood Model

As presented in Section 2.2.2, there are two approaches to model emotion, categorical and dimensional. The categorical model often uses a list of basic adjective words as emotion categories. The dimensional model uses some dimensions each one of which relates to some specific acoustical features. Our mood classification system does not base on acoustical features, so dimensional model is not suitable.

In 2007, there is a famous contest about Audio Music Mood Classification (MIREX 07) [64]. The contest concentrated on using audio information to detect mood of music. They use 5 mood clusters in which each mood cluster is represented by a group of words which have close meaning. These mood clusters reduce the diverse mood space into a tangible set of categories, yet root in the social-cultural context of music. Therefore, we use these mood categories for classification. Table 3.1 shows more details about each mood cluster, a group of words which define each cluster. We can see that cluster 1 shows an exciting mood; cluster 2 a joyful, gentle mood; cluster 3 a sad and gentle mood; cluster 4 a funny mood and cluster 5 an aggressive mood.

Table 3.1: Mood Clusters

Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**
Rowdy	Amiable/	Literate	Witty	Volatile
Rousing	Good natured	Wistful	Humorous	Fiery
Confident	Sweet	Bittersweet	Whimsical	Visceral
Boisterous	Fun	Autumnal	Wry	Aggressive
Passionate	Rollicking	Brooding	Campy	Tense/anxious
	Cheerful	Poignant	Quirky	Intense
			Silly	

3.2.2 Construction of Training Data

In this subsection, we will describe how to prepare our training data, the collection of songs tagged with their moods. Figure 3.2 shows the way we create it. We use a big blog site LiveJournal (www.livejournal.com) which has more than 9,000 users and each blog entry is tagged with mood and music. Users can choose a mood tag from 132 predefined moods of LiveJournal or input freely their current mood. Figure 3.3 is a LiveJournal post example, and Table 3.2 shows 25 most popular mood keywords of LiveJournal. Thanks to Gilly Leshed and Joseph 'Jofish' Kaye who collected LiveJournal blogs for their research [33], we used their dataset. It consists of 16.6 millions posts which contain both music and mood information.

Music tag is inputted arbitrarily even if it doesn't contain any music information. Almost users post music tag in the following formats "artist SEP title" or "title SEP artist"

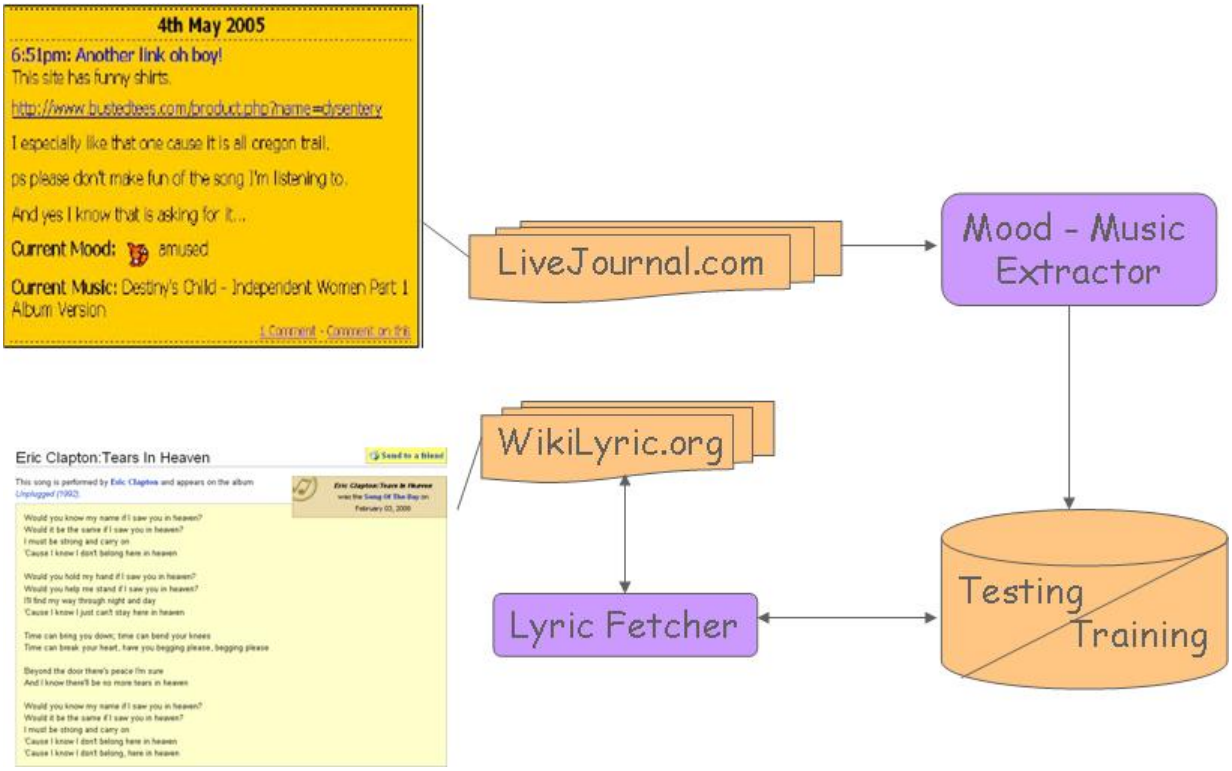

Figure 3.2: Data Construction Overview

Table 3.2: 25 Most Popular Mood Keywords in LiveJournal

tired	happy	bored	amused	blah
content	cheerful	contemplative	calm	depressed
sleepy	bouncy	excited	accomplished	confused
awake	chipper	drained	anxious	crazy
sick	annoyed	sad	blank	exhausted

in which SEP is one of separating symbols: $\{-, --, \sim, by\}$. Table 3.3 gives some examples of pairs of music and mood keyword extracted from LiveJournal blog posts. After removing all posts in which music tags are not written in those formats, the collection remains 9.7 millions posts.

By using SEP symbols we can separate artist and title information in a music tag, but it is still uncertain which one is an artist or title. To deal with this problem, we collect the most 6,820 popular artist/band names from an open music website www.musicmoz.org. After removing posts, songs of which don't contain any artist/band name in the above artist/band collection, we have 6.3 millions posts. Using only 50 most frequent mood keywords, finally we extracted about 665,000 songs of 6,800 artists/bands which are tagged with moods.

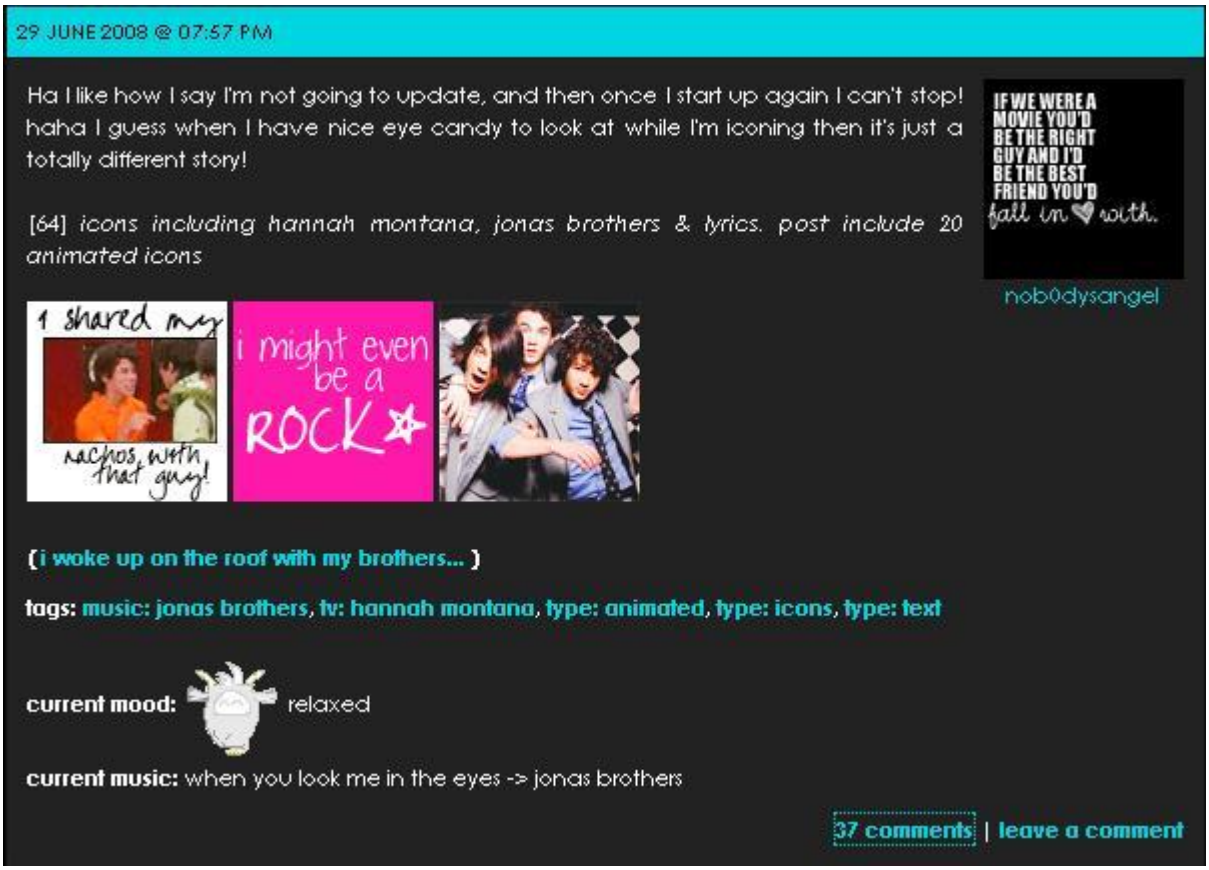

Figure 3.3: A LiveJournal Blog Post

Table 3.3: Some examples of music and mood information in LiveJournal

Music	**Mood Keyword**
Deep Purple - Knocking At Your Back Door	happy
Unwritten Law –– Cailin	indescribable
KISS ∼ Into the Void	full
The Beatles - Lucy in the Sky With Diamonds	lethargic
AC/DC - Thunderstruck	In the hole $5.00
none	tired
"You've Got a Friend" -James Taylor	stressed out to the max
"Ob-la-de Ob-la-da" by The Beatles	boring
"Ob-la-de Ob-la-da" -The Beatles... WHAT ELSE?!?!?	FAAAAAAAAANTASTIC

The next task is how to map mood keywords posted in LiveJournal to our mood categories. Note that LiveJournal mood keywords are mood of users, not mood of music. Thus, we must map LiveJournal mood keywords to our mood categories which are music moods. In almost cases, music mood and user mood are the same, but they are different in some cases, i.e. when a user feels happy, he tends to listen to happy songs, but if he feels tired, he will listen to relax songs. We manually design rules to map a mood keyword to mood categories. The mapping rules can be a 1-to-1 or 1-to-many mapping. For example, "sad" mood is mapped to only cluster 3, but "tired" is mapped to both clusters 3 and 5. Table 3.4 shows some examples of mapping rules used in our system. All rules are listed in Appendix A. There are four principles to design the mapping rules:

- Feel positive → listen positive music (Songs belong to mood categories 1, 2 and 4)
- Feel negative → listen negative music (Songs belong to mood categories 3 and 5)
- Feel a high energy or high stress mood → listen high energy music (Songs belong to mood categories 1 and 5)
- Feel a low energy or low stress mood → listen low energy music (Songs belong to mood categories 2, 3 and 4)

In LiveJournal, a song can be tagged with many mood keywords. Thus, how to choose the correct mood of a song is a problem. We design four methods:

Method 1 Choose the most frequent keyword mood as the keyword mood candidate(kmc), then decide the category mood of kmc using mapping rules. In ambiguous cases, the category mood candidate (cmc) of a song is determined according to sum of frequencies of category moods associated with that song. To avoid the cases in which frequency of kmc is too low (low confidence), we use the threshold value KMT. The condition of choosing kmc is shown in 3.1.

$$O(kmc, s) > KMT \tag{3.1}$$

where $O(kmc, s)$ is the co-occurrence frequency of kmc with song s.

Table 3.4: Some examples of mapping rules

Mood Keyword	**Mood Category**	**Mood Keyword**	**Mood Category**
tired	3;5	depressed	3
happy	2	bouncy	2
bored	3;5	excited	1
amused	4	accomplished	1
blah	3;5	awake	1
content	2	confused	3
cheerful	2	chipper	1
calm	2	anxious	5
contemplative	3	drained	3;5
sleepy	2;3	crazy	5

Method 2 Map all keyword moods of a song to category moods, then choose *cmc* with the ratio condition:

$$\begin{cases} O(cmc, s) > CMT \\ R(cmc, s) = \frac{O(cmc,s)}{\sum_{m \in M_s} O(m,s)} > RT \end{cases} \tag{3.2}$$

where $O(cmc, s)$ is co-occurrence frequency of the mood candidate *cmc* with the song s; M_s is the set of category moods relating to the song s.

Method 3 Map all keyword moods of a song to category moods, then choose *cmc* with the maximum difference condition: the frequency difference between the first and second most frequent category mood gains maximum value.

$$\begin{cases} O(cmc, s) > CMT \\ FD_s(m_1, m_2) > FD_s(m_i, m_{i+1}) \forall i \neq 1 \end{cases} \tag{3.3}$$

$$FD_s(m_i, m_{i+1}) = O(m_i, s) - O(m_{i+1}, s) \tag{3.4}$$

where m_i is the i^{th} most frequent mood category of the song s, $O(m_i, s)$ is co-occurence frequency of the song s and the mood category m_i.

Method 4 Map all keyword moods of a song to category moods, then choose *cmc* if occurrence of both *cmc* and *kmc* are great enough.

$$\begin{cases} O(kmc, s) > KMT \\ O(cmc, s) > CMT \end{cases} \tag{3.5}$$

The datasets constructed by these methods are checked using closed test. The best method is chosen for creating our dataset. The details of how to select the best method will be described in Section 4.1.

3.2.3 Proposed Methods

There are two approaches to solve mood classification problems: using acoustical and verbal information. In this paper, we will concentrate on the latter, especially using lyrics of songs. We present three methods: SVM classifier, Naive Bayes classifier and graph-based method.

3.2.3.1 SVM Classifier

SVM [61] classifiers are trained for mood categorization of songs. In this model, each song is represented as a vector with following features and weights.

Words Features All words in a lyric of a song are used as features. Weight of each feature in a vector is defined as (3.6):

$$weight(w) = tfidf(w) \tag{3.6}$$

where $tfidf$ is a product of TF (Term-Frequency) and IDF (Inverse-Document Frequency) of word w in our song collection.

Sentiment Words Feature Although a lyric expresses personal feelings, it contains a lot of short and incomplete sentences. Because of this characteristic, it's hard to detect the mood of a song if we simply apply a traditional method of text categorization problem.

In lyric, sentiment words show feelings clearly, so they are important features in deciding the mood of a song. Figure 3.4 gives us an example of sentiment words in a lyric. The underline words are sentiment words which contain almost emotion of that song. We can see that emotional polarity of a sentiment word (SW) can be changed by a negation word and be made stronger or weaker by a modifier words (MOD). For example, there's a positive meaning in "I love you", but if we use a negation as: "I don't love you", the emotional polarity is changed. We also see that the sentence "I love you very much" is stronger than "I love you". In brief, there are three types of sentiment events by which we can capture the mood of a song: occurrence of sentiment word (SW), occurrence of sentiment word with negation ($NEG\text{-}SW$) and occurrence of sentiment word with modifier ($MOD\text{-}SW$). In the first attempts, we just consider SW and $NEG\text{-}SW$.

Sentiment words are collected from SentiWordNet (115,448 words) which is a lexical resource for opinion mining. SentiWordNet assigns to each synset of WordNet three sentiment scores: positive $fsp(SW)$, negative $fsn(SW)$ and objective $fso(SW)$. These factors show the level of positivity, negativity and objectivity of each word. For example, the "ill" sentiment word has 0% positive, 75% negative and 25% objective as shown in Figure 3.5. We consider three features $\{SW_p, SW_n, SW_o\}$ for each sentiment word SW where SW_p, SW_n and SW_o is a feature representing the positive, negative and objective aspect of SW, respectively. The weighting model is designed as below:

Michael Learns To Rock - 25 Minutes

"*After some time I've finally made up my mind*
She is the girl and I really want to make her mine
I'm searching everywhere to find her again
To tell her I love her
And I'm sorry 'bout the things I've done

I find her standing in front of the church
The only place in town where I didn't search
She looks so happy in her wedding dress
But she's crying while she's saying this

Chorus:
Boy I missed your kisses all the time but this is
Twenty five minutes too late
Though you traveled so far boy I'm sorry you are
Twenty five minutes too late

Against the wind I'm going home again
Wishing be back to the time when we were more than
friends

But still I see her in front of the church
The only place in town where I didn't search
She looks so happy in her wedding dress
But she's cried while she's saying this

Chorus

Out in the streets
Places where hungry hearts have nothing to eat
Inside my head
Still I can hear the words she said

I can still hear what she said"

Figure 3.4: An example of sentiment words in lyric

$$weight(SW_p) = (tf(SW) \times fsp(SW) + tf(NEG\text{-}SW) \times fsn(SW)) \times idf(SW) \quad (3.7)$$
$$weight(SW_n) = (tf(SW) \times fsn(SW) + tf(NEG\text{-}SW) \times fsp(SW)) \times idf(SW) \quad (3.8)$$
$$weight(SW_o) = tfidf(SW) \times fso(SW) \quad (3.9)$$

In (3.7) and (3.8), $tf(NEG\text{-}SW)$ is multiplied by $fsn(SW)$ for SW_p and $fsp(SW)$ for SW_n because $NEG\text{-}SW$ has an opposite polarity.

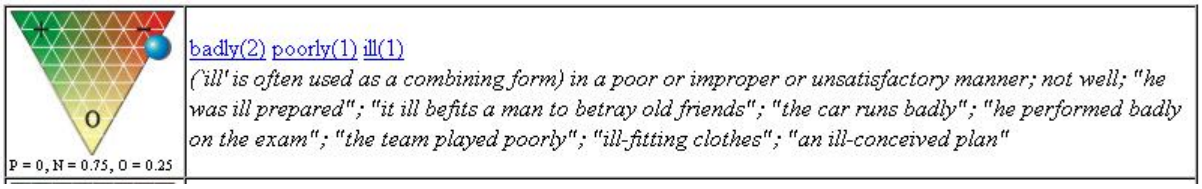

Figure 3.5: Sentiment scores of "ill"

Artist Feature We can observe that each artist/band often sings songs about a specific topic with a specific mood. For example, Eric Clapton often sings sad songs but Bob Marley likes singing happy songs. Table 3.5 give us some particular examples about this phenomenon with artist/bands extracted from our dataset.

Table 3.5: Mood Distribution on Artists

Artist	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
The Beatles	4	24	25	1	2
Metallica	1	4	32	4	10
Green Day	0	9	27	4	2
Evanescence	0	3	30	1	1
Bob Marley	0	13	5	0	1

Table 3.5 indicates that artists/bands are good features for identifying mood of a song. In the SVM classifier, we assign artist feature's weight of each song with $tfidf$ model:

$$weight(a) = tfidf(a) \quad (3.10)$$

We assume $tf(a) = 1$, so we have:

$$weight(a) = idf(a) = \log \frac{N}{N(a)} \quad (3.11)$$

where N is the number of songs in training data, $N(a)$ is the number of songs of artist a.

Weighting by Entropy In this method, entropy of the distribution probability of moods is integrated into the weighting model as (3.12):

$$weight(w) = \frac{tf(w)}{H(w)} \tag{3.12}$$

$$H(w) = -\sum_{m \in M} P(m|w) \log P(m|w) \tag{3.13}$$

In (3.13), $P(m|w)$ is the probability that mood category of a song is m when its lyric contains the word w. If the probabilistic distribution $P(m|w)$ is nearly uniform, i.e. $H(w)$ is great, w would be ineffective to classify moods of songs. Thus we put lower weights if $H(w)$ is great as indicated in (3.12).

3.2.3.2 Naive Bayes Classifier

Basic Model In this model, the mood m for a song s is chosen such that $P(m|L)$ is the greatest, where L stands for a lyric. According to Naive Bayes assumptions:

$$\begin{aligned} m_{select}(s) &= \arg\max_m P(m|L) \\ &= \arg\max_m \tfrac{P(m)P(L|m)}{P(L)} \\ &= \arg\max_m P(m)P(L|m) \\ &\simeq \arg\max_m P(m)\Pi_{w \in L} P(w|m)^{TF(w,L)} \end{aligned} \tag{3.14}$$

where $TF(w, L)$ is the frequency of term w in lyric L. $P(w_i|m)$ is estimated with the following formula:

$$P(w|m) = \frac{1 + O(w, m)}{|F| + \sum_{w \in F} O(w, m)} \tag{3.15}$$

where $O(w, m)$ is co-occurrence frequency of word w and mood m, F is a set of all features (words) in lyrics in the training data.

Weighting for Chorus and Title Part Each song has a specific structure in which each part plays a different role. A lyric is similarly divided into several parts:

Title: It describes theme of song shortly.

Introduction (Intro): It is usually one verse composed by three or four phrases used to introduce the main theme or to give a context to the listener.

Verse: Verse roughly corresponds with a poetic stanza. Lyrics in verses tend to repeat less than they do in choruses.

Chorus: The refrain of a song. A verse repeats at least twice with none or little differences between repetitions. It is the most repetitive part of a lyric. It is also where the main theme is more explicit. As well as what happens with music, it is also the part

which listeners tend to remember.

Bridge: In song writing, a bridge is an interlude that connects two parts of that song. As verses repeat at least twice, the bridge may then replace the 3rd verse or follow it thus delaying the chorus. In both cases it leads into the chorus.

Outro: It is not always present, this part is located at the end of a lyric and tends to be a conclusion about the main theme.

In a given song, the most important part is chorus which is repeated many times and contains almost meaning and emotion of a song. This is the part which listeners tend to remember. Thus words in the chorus part will be effective features to predict the mood of a song. Beside chorus, the title of a song which gives us the main topic of it also contains some useful mood information.

We consider modifying Naive Bayes classifier to put more weights for words in the CHORUS or TITLE part. In this model, $P(w|m)$ is estimated as follows:

$$P(w|m) = \frac{1 + O(w,m) \times (1 + vote_w)}{|F| + \sum_{w \in F} O(w,m) \times (1 + vote_w)} \tag{3.16}$$

$$vote_w = \begin{cases} 0 \text{ if } w \notin CHORUS \wedge w \notin TITLE \\ V_c \text{ if } w \in CHORUS \wedge w \notin TITLE \\ V_t \text{ if } w \notin CHORUS \wedge w \in TITLE \end{cases} \tag{3.17}$$

where V_c, V_t are parameters to put more weights for terms in CHORUS and TITLE parts. In this paper, these parameters are decided empirically.

Artist Feature As mentioned in 3.2.3.1, artist is an important feature for mood detection problem. Each artist tends to compose or sing a kind of music. Assume that an artist art and a lyric L are independent, we have:

$$\begin{aligned} m_{select}(s) &= \arg\max_{m \in M} P(m|L, art) && (3.18) \\ &= \arg\max_{m \in M} \frac{P(L, art|m) \times P(m)}{P(L, art)} && (3.19) \\ &\simeq \arg\max_{m \in M} P(L|m) \times P(art|m) \times P(m) && (3.20) \\ &= \arg\max_{m \in M} P(L|m) \times \frac{P(m|art) \times P(art)}{P(m)} \times P(m) && (3.21) \\ &= \arg\max_{m \in M} P(L|m) \times P(m|art) && (3.22) \end{aligned}$$

$P(L|m)$ is the same of the second term of (3.14), estimated by the following formula:

$$P(L|m) = \Pi_{w \in L} P(w|m)^{TF(w,L)} \tag{3.23}$$

while $P(m|art)$ is estimated as follows:

$$P(m|art) = \begin{cases} \left(\frac{N(art,m)}{N(art)}\right)^{wc} if N(art) \geq X \\ \left(\frac{N(gnr,m)}{N(gnr)}\right)^{wc} if N(art) < X \end{cases} \tag{3.24}$$

In (3.24), $N(art)$ is the number of songs of the artist art, while $N(art, m)$ is the number of songs of art tagged with mood m. In case that $N(art)$ is not great, estimated $P(m|art)$ might be unreliable. For smoothing, we estimated $P(m|gnr)$ instead of $P(m|art)$ when $N(art)$ is less than a certain threshold X, where gnr stands for the genre of a song. We set $X = 5$. Finally, wc is a parameter defining the weight for $P(m|art)$ compared with $P(L|m)$.

Finally, we have:

$$m_{select}(s) = \begin{cases} \arg\max_{m \in M} \Pi_{w \in L} P(w|m)^{TF(w,L)} \times \left(\frac{N(art,m)}{N(art)}\right)^{wc} if N(art) \geq X \\ \arg\max_{m \in M} \Pi_{w \in L} P(w|m)^{TF(w,L)} \times \left(\frac{N(gnr,m)}{N(gnr)}\right)^{wc} if N(art) < X \end{cases} \tag{3.25}$$

3.2.3.3 Graph-Based Method

In many settings, the usual approach for classification problem does not exploit the available information about relationships between data items. Using the relationship information, we can construct a graph G in which each data item is a node and each relationship forms an edge between the corresponding nodes. Then the classification problem can be formulated as a graph labeling or coloring problem on such a graph.

In our problem, each node is a song and links are created by artist relationship (two songs are connected if they belong to the same artist). In order to try the first attempt on evaluating affection of graph-based approaches to our mood classification problem, we apply the simplest method of Oh et al. [55]. Firstly, a graph of test data is built based on links of data items. Then, they use Naive Bayes model to classify test data. Finally, in order to consider neighbors' affection to a data item, they classify test data again using the test graph. The classification model for the final phase as follow:

$$C_{select} = \arg\max_{C} P(C|G, T) \tag{3.26}$$

$$= \arg\max_{C} P(T|C)P(C|G) \tag{3.27}$$

$$= \arg\max_{C} \Pi_{i=1}^{|T|} P(t_i|C)^{N(t_i,d)} \times Neighbor_d(C) \tag{3.28}$$

where C stands for a category (mood), T a document (lyric), G a graph and $N(t_i, d)$ frequency of the term t_i in the node d.

Neighbor function is calculated from neighbor data items, not from training data. It basically computes the degree to which the current class c is supported by the neighbors

of a node d.

$$Neighbor_d(C) = \frac{l_d(C)}{l_d} \times w_L \tag{3.29}$$

Here l_d and $l_d(C)$ represent the number of all links from or to d and the number of neighbors having class label c, respectively. In addition, w_L represents the average weight for all the links and indicates to what extent the categories of the neighbor nodes are confident.

Then we propose the new model which is an extension of Oh's method. First, we construct two graphs, G_a and G_g. Songs of the same artist are connected in G_a, while songs in the same genre are connected in G_g. We primary use G_a. However, if the number of neighbors is small, the neighbor function would be unreliable. In such cases, we use G_g where the number of neighbors will be much greater than in G_a. Neighbor function in our new model is summarized as follows:

$$Neighbor_d(C) = \begin{cases} \left(\frac{l_d(C)}{l_d}\right)^{wc} \text{in } G_a & \text{if } l_d \geq X \\ \left(\frac{l_d(C)}{l_d}\right)^{wc} \text{in } G_g & \text{if } l_d < X \end{cases} \tag{3.30}$$

Here we set X is 5. Note that the parameter w_L in (3.29) is replaced with wc in (3.30), that is, we modified the way to decide the weight for categories in neighbors.

Chapter 4

Evaluation

In this chapter, we evaluate methods in Proposed Methods. First, the methods creating datasets are investigated to choose the best dataset for mood detection. Next, we analyze and discuss characteristics of mood detection methods using SVM, Naive Bayes and graph-based classifier.

4.1 Construction of training dataset

In this section, we evaluate all three methods of creating training dataset. First, 50 most frequent mood keywords of LiveJournal are chosen as the basic mood keywords. Mapping rules are designed manually to map these 50 mood keywords to mood categories. Then, four datasets are created with the same size by the methods presented in Section 3.2.2. Closed tests are applied with Naive Bayes (NB) classifier for evaluating the quality of these datasets. In a close test, the training dataset is also test dataset. We assume that the training data is more consistent if the accuracy of NB classifier is greater.

Table 4.1: Accuracies with Closed Tests

Methods	**Parameters**	**Training Size**	**Accuracy (%)**
Method 1	KMT=9	2,610	78.82
Method 2	CMT=20,RT=0.7	2,468	79.89
Method 3	CMT=20	2,507	79.66
Method 4	CMT=15,KMT=9	2,504	80.02

From Table 4.1, we can see that the best method is 4 in which confidence of keyword mood is taken into account beside confidence of category mood. Method 1 uses only keyword mood candidate to decide the category mood of a song. It does not consider votes of other keyword moods even they are mapped to the same category mood with keyword mood candidate. It makes this method has a lowest quality. The three methods 2, 3 and 4 would overcome the problem of method 1 since they consider the occurrence of mood categories. As a result, they achieve higher accuracies. The highest accuracy of the method 4 shows that confidence of keyword mood candidate is more important than

complicated conditions described in method 2 and 3.

In Tables 4.2, 4.3, 4.4 and 4.5, we want to investigate how parameters of each method affect to the size and quality of datasets through NB classifier. With each method, we change parameters to make datasets become more consistent in compensation for the data size. Then these datasets are divided into five parts, one for testing data and the remaining for training data. We use NB classifier to test on these datasets. According to the tables, as we expected, accuracies of systems become great when we set the threshold higher so that only confident moods of songs are tagged.

Table 4.2: Datasets created by method 1

KMT	**Size**	**Baseline (%)**	**NB (%)**
10	2275	55.34	57.68
15	1283	58.53	62.16
20	843	60.74	64.91

Table 4.3: Datasets created by method 2 (CMT=20)

RT	**Size**	**Baseline (%)**	**NB (%)**
0.7	2468	47.29	59.48
0.8	2457	47.50	59.31
0.9	2455	47.54	59.92

Table 4.4: Datasets created by method 3

CMT	**Size**	**Baseline (%)**	**NB (%)**
10	5325	43.68	53.98
15	3423	45.17	58.37
20	2507	46.55	59.13
25	1970	48.22	62.88
30	1585	48.01	66.77

Table 4.5: Datasets created by method 4

Dataset	**Size**	**Baseline (%)**	**NB (%)**
10;5	5580	55.34	56.88
10;10	2275	55.34	58.11
15;10	2214	56.86	59.91
15;15	1283	58.53	60.23

Finally, we chose the method 4 with parameters ($CMT = 10; KMT = 5$) to create our final dataset. This dataset consists of 5,580 songs. To check the quality of this dataset,

we chose 50 songs randomly and checked mood of each song manually. The mood tags of 92% songs are correct. Table 4.6 shows us some examples of songs and their mood categories created by the method 4. Furthermore, lyrics of songs in our collection are obtained from LyricWiki website (www.lyricwiki.org).

Table 4.6: Some songs with their moods

Song	**Category Mood**
toby keith - who's your daddy?	1
12 stones - fade away	3
polaris - hey sandy	3
five iron frenzy - on distant shores	3
saves the day - as your ghost takes flight	5
the libertines - death on the stairs	2
dream theater - caught in a web	5
the cure - bloodflowers	3
tantric - astounded	5
avril lavigne - my happy ending	3
quarashi - stick 'em up	5
travis - writing to reach you	2
our lady peace - 4 am	3
the cure - labyrinth	3
black sabbath - sweet leaf	2
nelly - heart of a champion	2
the doors - five to one	2
arch enemy - we will rise	3
meredith brooks - im a bitch	5
david bowie - young americans	2
placebo - 20th century boy	1

4.2 Mood Detection

In this section, we will report the results of experiments to evaluate our methods. First, we define the baseline as the naive system which always selects the most frequent mood (cluster 3 in our training data).

4.2.1 SVM Classifier

We trained the following four SVM classifiers:

- SVM-AW: use all words as features
- SVM-SW: use all words and sentiment words as features

- SVM-AA: use all words and artist as features
- SVM-AE: use all words and entropy weighting model

Table 4.7 reveals the accuracies of SVM classifiers evaluated by five-fold cross validation as well as the baseline (BL).

Table 4.7: Accuracies of SVM Classifiers

BL	**SVM-AW**	**SVM-SW**	**SVM-AA**	**SVM-AE**
54.12%	50.58%	52.73%	52.41%	52.11%

Table 4.7 shows that artist and sentiment word features are good for mood classification. Accuracy is improved 2% in average on three methods compared with SVM-AW, although they do not outperform the baseline. However, sentiment word features are not as good as we expect. The reason is that number of sentiment words is very few, so this method can not capture moods of songs.

4.2.2 Naive Bayes Classifier

We evaluated following four Naive Bayes classifiers:

- NB: basic model
- NB-C: Naive Bayes with weighting chorus part
- NB-T: Naive Bayes with weighting title part
- NB-A: Naive Bayes with artist features

We divide the dataset into 6 parts, one for testing data, one for validation data and the remaining for training data. Then parameters V_c, V_t and wc are optimized in the validation data, which is mutual exclusive with both testing and training data. Table 4.8 shows the accuracies of NB classifiers as well as optimized parameters.

Table 4.8: Accuracies of Naive Bayes Classifiers

BL	**NB**	**NB-C**	**NB-T**	**NB-A**
54.12%	53.40%	56.44% (V_c=7.2)	54.92% (V_t=1.6)	57.44% (wc=10.5)

This result shows that NB classifier works better than SVM classifier on mood classification problem. More weighting for words in title and chorus can improve the system. The highest improvement is the method using artist feature NB-A.

Below we will analyze more detail about NB-C, NB-T and NB-A methods. In Table 4.9 and 4.10 we analyze affections of parameters V_c and V_t to accuracies of the system on

the test data. We expected that features occurring in chorus and title parts are effective, and we found that it is true. When we set V_c and V_t to values in the range from 1 to 10, our system always outperforms the basic model. At the value of $V_c = 7$ and $V_t = 2$, our systems are even better than the basic model 3% and 1.7% respectively.

Table 4.9: Chorus Voting

V_c	**Accuracy (%)**	V_c	**Accuracy (%)**
1	54.91	6	55.81
2	55.46	7	56.35
3	55.46	8	55.90
4	55.55	9	55.72
5	55.72	10	55.90

Table 4.10: Title Voting

V_t	**Accuracy (%)**	V_t	**Accuracy (%)**
1	54.56	6	54.29
2	55.10	7	54.47
3	54.56	8	54.47
4	54.47	9	54.29
5	54.03	10	54.38

The model of method NB-A described in Equation (3.25) has two parameters: artist smoothing threshold X and confident parameter wc. In Figure 4.1, we changed wc from 1 to 27 and investigated its affections to accuracy. Here we set the parameter $X = 5$. According to this graph, NB-A always outperforms the basic model when wc is positive. However, when we set wc negative, -3, -2 and -1, accuracy of NB-A is 33.18%, 40.25% and 48.66%, respectively, which is much worse than the basic model. This means that $P(L|m)$ is more important than $P(m|art)$ in the Equation (3.25). The maximum accuracy is gained at $wc = 9$ and $wc = 11$. The system is stable with wc values from 13 to 27 with average accuracy 56.59% which is 3.19% better than the NB system.

Table 4.11 shows affection of smoothing threshold X to accuracy in the range from 1 to 9. Without smoothing step, the system achieve 56.88% accuracy. Accuracies of the systems almost are almost same in this range but always better than the system without smoothing.

4.2.3 Graph-Based Method

We evaluated following three graph-based methods:

- NB: content-based Naive Bayes classifier
- GC-Oh: graph-based method by Oh et al.

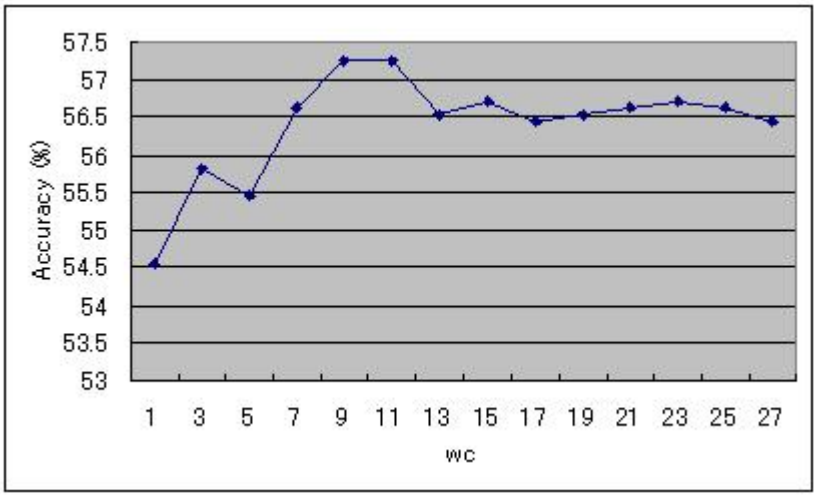

Figure 4.1: Affection of wc to NB-A

Table 4.11: Smoothing of NB-A. No smoothing = 56.88%

X	**Accuracy (%)**
1	57.51
3	57.42
5	57.51
7	57.51
9	57.51

- GC-New: graph-based method with our extension of neighbor function ($wc = 7$)

Table 4.12: Accuracies of Graph-based Method

BL	**NB**	**GC-Oh**	**GC-New**
54.00%	53.40%	53.75%	57.00%

This result shows that relationship information among songs is useful for mood classification. Our result is improved 3.6% to Naive Bayes and 3.25% to GC-Oh in which only artist relationship is used to build graph. If other types of relationship are used, we believe that the result will be much better. Applying more sophisticated graph-based classification methods such as [45, 44] will also improve accuracy.

Now we consider some characteristics of this method: training size/test size ratio, wc parameter, role of genre graph and performance on noise data.

In Table 4.13, we performed some experiments to investigate affection of train/test size ratio settings on performance of the GC-New system. First, our dataset is divided into five parts. Then with the first setting, we use four parts for training data and one part for

testing data. On the contrary, one part is for training data and four parts are for testing data in the second setting. GC-New(Test) and GC-New(All) is GC-New classifier that uses only testing data and both testing and training data to build the graph, respectively. We can see that the classifier GC-New(Test) gains almost same accuracies in the two settings. This is very meaningful in practical applications because with the second setting, the system needs a human labour 1/4 less than the first one. Comparing GC-New(Test) and GC-New(All) in the first setting, we can see that when the more labeled data items in our graph, the greater accuracy of the system is. However, accuracies of both systems are nearly equal in the second setting. The reason would be that labeled data items are too smaller than unlabeled data items.

Table 4.13: Accuracies with various train/test size ratios

Setting	**Train/Test**	**NB**	**GC-New(Test)**	**GC-New(All)**
(1)	4/1	53.40%	55.28%	56.62%
(2)	1/4	52.25%	54.25%	54.25%

Figure 4.2 shows affection of wc parameter to the performance of GC-New classifier. The maximum accuracy(57%) is gained at $wc = 7$. Then, the system is stable in the range from 15 to 27 with average accuracy of 56.53% which is 3.13% better than the NB system.

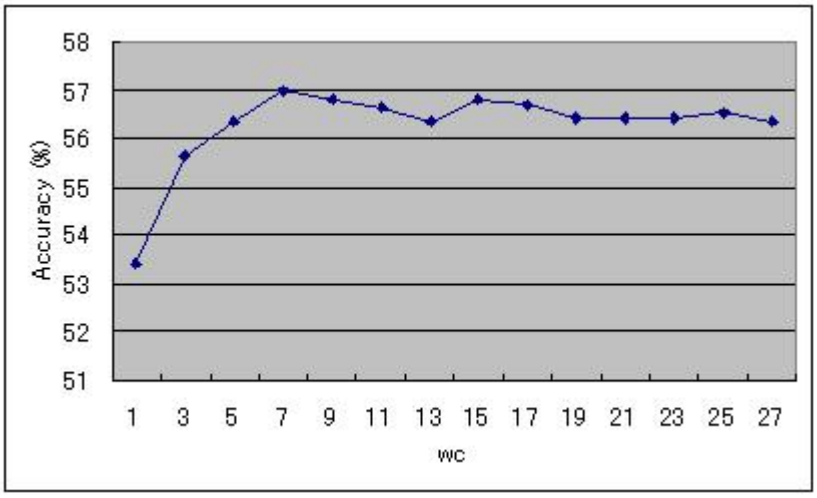

Figure 4.2: Affection of wc to GC-New

In Table 4.14, we evaluate the contribution of genre graph in supporting for artist graph. There are three experiments: do not use genre graph, use genre graph with $X = 2$ and $X = 5$. The results show that genre graph can improves the system, namely about 0.63% with $X = 2$ and 0.45% with $X = 5$.

Table 4.14: Affections of Genre Graph

Without genre graph	**Use genre graph (X=2)**	**Use genre graph(X=5)**
56.53%	57.16%	56.98%

In Table 4.15, we change the quality of datasets by increasing CMT and KMT parameters and evaluate NB and GC-New systems on them. We can see that the more noisy the dataset is, the better GC-New is compared to NB. With the most noisy dataset ($CMT = 10, KMT = 5$) in these settings, GC-New is 3.49% better than NB, however their accuracies are the same in the most confident one ($CMT = 20, KMT = 15$). This means that GC-New can overcome noise problem in a dataset.

Table 4.15: Noise Affections

CMT, KMT	**Baseline**	**NB**	**GC-New**
10,5	54.12	53.40	56.89
10,10	55.34	57.68	58.11
15,10	56.87	59.91	59.91
20,15	59.46	62.35	62.35

4.3 The Best Classifier with Various Datasets

In this section, we investigate the best classifier NB-A on various datasets created by the methods proposed in Subsection 3.2.2. Here wc parameter is set to 9.

First, we performed closed tests on these datasets with NB-A. Table 4.16 shows our experiments. According to the results, datasets created by method 2 and 3 are better than the dataset created by method 4. This conflicts with results shown in Table 4.1. Method 2 and 3 are worse than the method 4 in Table 4.1, now they become better in Table 4.16. We guess the reason as follows. Many songs have same artists/bands in the dataset 2 and 3, so the estimated probability $P(m|art)$ in Equation (3.24) is reliable enough. While only a few songs have same artists in the dataset 4, causing the contribution of artist feature to be less sensitive. This means that if we use artist feature, method 2 and 3 datasets should be used to achieve the best performance.

Table 4.16: Closed tests with NB-A

Dataset	**NB-A (%)**
Method 1 (KMT=9)	77.43
Method 2 (RT = 0.7; CMT=20)	79.69
Method 3 (CMT=20)	79.51
Method 4 (CMT=15, KMT=9)	77.78

Next, we perform an analysis of NB-A method on the above datasets with the *wc* parameter changing in the range from 1 to 27 in open tests. The graph for the method 1 and 4 has the same shape. This phenomenon is also the same to method 2 and 3. According the graph, we know that NB-A method works best on method 2 dataset and worst on the method 1 dataset. Surprisingly, although the method 4 dataset has highest accuracy with NB classifier (59.84%), it is not better than the method 2, 3 datasets with NB-A classifier on almost values of *wc* (from 3 to 19). The reason is the same to that in closed test experiments mentioned above.

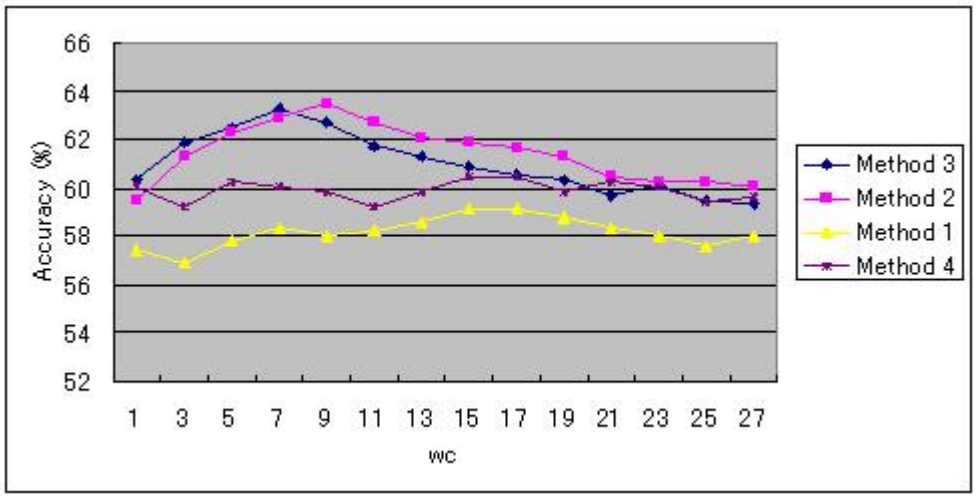

Figure 4.3: Affections of *wc* to NB-A on various datasets

Chapter 5

Conclusion

In this research, we built a music search engine system using natural language queries to explore cultural information of music. This search engine then is supported with a mood classification system. Our first contribution is building the first great enough ground-truth dataset for mood detection problem. The second one is that we proposed some effective methods that use both lyric and metadata to classify moods. The accuracy of mood classification methods is not really high and good enough to apply for a real music search engine system. However acoustical methods also can not achieve high accuracy. In the 2007 MIREX audio mood classification contest [66], the highest result is 61.50%. If we use the dataset with 2468 songs created by method 2 ($RT = 0.7$, $CMT = 20$), the best classifier NB-A ($wc = 9$) can achieve an accuracy 63.25%, a little better than the MIREX contest's result. There are two main reasons for low accuracy: mood is a subjective metadata; lyric is short and contains many metaphors which only human can understand. However the experiments showed that artist, sentiment words, putting more weight for words in chorus and title parts are effective for mood classification. Graph-based method promises a good improvement if we have rich relationship information among songs.

5.1 Future Work

Mood Detection In future works, we have a plan to use karaoke music data that enables mapping from each word in lyric to a musical notes. Using this kind of mapping, we believe that results will be improved much. Currently, we apply just a simple graph-based method and use only artist link type to create the graph. Thus, a more complicated model for graph-based method with various types of links also promises a great improvement.

Music Index Builder As you know, there are many types of information written on Web pages. The task of Music Index Builder (MIB) is to detect all words, sentences or even phrases relating to music domain. Namely, this kind of information is music metadata fields like artist name, title of song or album name, etc. However, current MIB module, like a state of the art search engine, just build a index for all pages relating to songs in our music collection. Hence, we are intending to apply mining methods and

probabilistic Latent Semantic Indexing technique for revealing meanings in Web pages.

Interactive Agent Interactive Agent (IA) is one of our future works. There are two kind of problems we have to solve with IA. The first one is how to understand queries semantically in the music domain. Two types of semantic that we want to concentrate here are mood of query and music information. The second one is how to rank the song result in which song similarity (based on both acoustical and textual) and song popularity are the top priorities.

Music Recommendation It is imperative to provide music listeners with the proper tools with which to access their music. Many of the current recommendation systems lack a thorough understanding of the content and context of a song. We have a plan to build a recommendation system supporting for this music search system. Bayesian Set [15] is an approach to build such a kind of system.

Bibliography

[1] D. Chamberlin A. Ghias, J. Logan and B. C. Smith. Query by humming: musical information retrieval in an audio database. San Francisco, CA, USA, 1995. The 3rd ACM International Conference on Multimedia.

[2] D. Watson A. Tellegen and L.Clark. On the dimensional and hierarchical structure of affect. *Psychological Science*, 1999.

[3] R. Lewis A. Wieczorkowska, P. Synak and Z. Ras. Extracting emotions from music data. Saratoga Springs, USA, 2004. the 15th International Symposium on Methodologies for Intelligent Systems.

[4] S. Omar Ali and Zehra F. Peynircioglu. Songs and emotions: are lyrics and melodies equal partners? *Psychology of Music*, 2006.

[5] Carnagey N.L. Anderson C.A. and Eubanks. Exposure to violent media: The effects of songs with violent lyrics on aggressive thoughts and feelings. *Journal of Personality and Social Psychology*, 2003.

[6] Chengliang Zhang Bin Wei and Mitsunori Ogihara. Keyword generation for lyrics. ISMIR, 2007.

[7] Peretz I. Bonnel A.M., Faita F. and Besson M. Divided attention between lyrics and tunes of operatic songs: Evidence for independent processing. *Perception and Psychophysics*, 2001.

[8] Hansen C. and Hansen R. Rock music videos and antisocial behavior. *Basic and Applied Social Psychology*, 1990.

[9] S. Chakrabarti. Mining the web: Discovering knowledge from hypertext data. Morgan-Kauffman, 2002.

[10] S. Chakrabarti. Breaking through the syntax barrier: Searching with entities and relations. ECML, 2004.

[11] Serafine M.L. Crowder R.G. and Repp B. Physical interaction and association by contiguity in memory for the words and melodies of songs. *Memory and Cognition*, 1990.

[12] P. Ekman. An argument for basic emotions. *Cognition and Emotion*, pages 169–200, 1992.

[13] P. R. Farnsworth. *The social psychology of music*. The Dryden Press, 1958.

[14] B. Fehr and J. A. Russell. Concept of emotion viewed from a prototype perspective. *Experimental Psychology: General*, pages 464–486.

[15] Zoubin Ghahramani and Katherine A. Heller. Bayesian sets. 2006.

[16] Bradley R.J. Harris C.S. and Titus S.K. A comparison of the effects of hard rock and easy listening on the frequency of observed inappropriate behaviors: Control of environmental antecedents in a large public area. *Journal of Music Therapy*, 1992.

[17] K. Hevner. The affective character of the major and minor modes in music. *The American Journal of Psychology*, pages 103–118, 1935.

[18] K. Hevner. Expression in music: A discussion of experimental studies and theories. *Psychological Review*, pages 186–204, 1935.

[19] K. Hevner. Experimental studies of the elements of expression in music. *The American Journal of Psychology*, pages 246–268, 1936.

[20] K. Hevner. The affective value of pitch and tempo in music. *The American Journal of Psychology*, pages 621–630, 1937.

[21] David Chia-Wei Hsu. iplayr: an emotion-aware music player. Master's thesis, National Taiwan University, June 2007.

[22] D. Huron. Perceptual and cognitive applications in music information retrieval. 2000.

[23] D. Cerizza I. Celino, E. Della Valle and A. Turati. Squiggle: a semantic search engine for indexing and retrieval of multimedia content. Athens, Greece, 2006. The 1st International Workshop on Semantic-enhanced Multimedia Presentation Systems.

[24] ISMIR. *Music and Lyrics: Can Lyrics Improve Emotion Estimation for Music?*, 2008.

[25] Arnett J. Heavy metal music and reckless behavior among adolescents. *Journal of Youth and Adolescents*, 1991.

[26] E. Tardos J. Kleinberg. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. FOCS, 1999.

[27] M. F. McKinney J. Skowronek and S. van de Par. Ground truth for automatic music mood classification. Victoria, Canada, 2006. the 7th International Conference on Music Information Retrieval.

[28] T. Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. Int. Conf. Machine Learning, 1997.

[29] P. N. Juslin. Cue utilization in communication of emotion in music performance: relating performance to perception. *Experimental Psychology: Human Perception and Performance*, pages 1797–1813, 2000.

[30] Gfeller K. and Coffman D.D. An investigation of emotional response of trained musicians to verbal and music information. *Psychomusicology*, 1991.

[31] I. Knopke. Aroooga: An audio search engine for the world wide web. Miami, USA, 2004. The International Computer Music Conference.

[32] D. Liu L. Lu and H.-J. Automatic mood detection and tracking of music audio signals. pages 5–18, 2006.

[33] Gilly Leshed and Joseph 'Jofish' Kaye. Understanding how bloggers feel: Recognizing affect in blog posts. Montreal, Quebec, Canada, 2006. CHI.

[34] T. Li and M. Ogihara. Detecting emotion in music. Baltimore, USA, 2003. 4th International Conference on Music Information Retrieval.

[35] T. Li and M. Ogihara. Content-based music similarity search and emotion detection. volume 5, pages 705–708. IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004.

[36] Hugo Liu and Push Singh. Conceptnet: A practical commonsense reasoning toolkit. 22, 2004.

[37] Galizio M. and Hendrick C. Effects of musical accompaniment on attitude: The guitar as a prop for persuasion. *Journal of Applied Social Psychology*, 1972.

[38] L. D. Voogdt M. Leman, V. Vermeulen and D. Moelants. Using audio features to model the affective response to music. Nara, Japan, 2004. International Symposium on Musical Acoustics.

[39] R. Tato M. Tolos and T. Kemp. Mood-based navigation through large collections of musical data. pages 71–75, 2005.

[40] N. Zhang M. Wang and H. Zhu. User-adaptive music emotion recognition. Istanbul, Turkey, 2004. International Conference on Signal Processing.

[41] Owen Craigie Meyers. A mood-based music classification and exploration system. Master's thesis, MASSACHUSETTS INSTITUTE OF TECHNOLOGY, 2004.

[42] Burt M.R. Cultural myths and supports for rape. *Journal of Personality and Social Psychology*, 1980.

[43] P. Cano O. Celma and P. Herrera. Search sounds: An audio crawler focused on weblogs. Victoria, B.C., Canada, 2006. the 7th International Conference on Music Information Retrieval.

[44] L. Getoor Q. Lu. Link-based classification. ICML, 2003.

[45] G Weikum. R Angelova. Graph-based text classification: Learn from your neighbors. SIGIR, 2006.

[46] Remco C. Veltkamp Rainer Typke, Frans Wiering. A survey of music information retrieval systems. 2005.

[47] M. G. Rigg. What features of a musical phrase have emotional suggestiveness? 1939.

[48] M. G. Rigg. The mood effects of music: A comparison of data from four investigators. *The Journal of Psychology*, pages 427–438, 1964.

[49] Paolo Petta Robert Trappl and Sabine Payr. *Emotions in Humans and Artifacts*. 2002.

[50] J. A. Russell. A circumplex model of affect. *Personality and Social Psychology*, pages 1161–1178, 1980.

[51] A. Kluter S. Baumann and M. Norlien. Using natural language input and audio analysis for a human-oriented mir system. Germany, 2002. The 2nd International Conference on Web Delivering of Music.

[52] Sousou S.D. Effects of melody and lyrics on mood and memory. *Perceptual and Motor Skills*, 1997.

[53] Crowder R.G. Serafine M.L. and Repp B.H. Integration of melody and text in memory for songs. *Cognition*, 1984.

[54] Crowder R.G. Serafine M.L., Davidson J. and Repp B.H. On the nature of melody-text integration in memory for songs. *Journal of Memory and Language*, 1986.

[55] SIGIR. *A Practical Hypertext Categorization Method using Links and Incrementally Available Class Information*, 2000.

[56] SIGIR. *A Music Search Engine Built upon Audio-based and Web-based Similarity Measures*, 2007.

[57] J. A. Sloboda and P. N. Juslin. *Music and Emotion: Theory and Research*. Oxford University Press, 2001.

[58] R. Thayer. The biopsychology of mood and arousal. 1999.

[59] G. Tzanetakis and P. Cook. Marsyas: a framework for audio analysis. pages 169–175, 1999.

[60] Bram van de Laar. Emotion detection in music, a survey. 4th Twente Student Conference on IT, 2006.

[61] V. Vapnik. The nature of statistical learning theory. 1995.

[62] Stratton V.N. and Zalanowski. Affective impact of music vs. lyrics. *Empirical Studies of the Arts*, 1994.

[63] K. B. Watson. The nature and measurement of musical meanings. *In Psychological Monographs*, 1942.

[64] Wikipedia. http://www.music-ir.org/mirex/2007.

[65] wikipedia.org. http://en.wikipedia.org/wiki/music_theory on 26 january 2009.

[66] C Laurier M Bay X Hu, JS Downie and AF Ehmann. The 2007 mirex audio mood classification task: Lessons learned. ISMIR, 2008.

[67] Y. Zhuang Y. Feng and Y. Pan. Music information retrieval by detecting mood via computational media aesthetics. page 235, Washington, USA, 2003. IEEE/WIC International Conference on Web Intelligence.

[68] D. Yang and W. Lee. Disambiguating music emotion using software agents. Barcelona, Spain, 2004. 5th International Conference on Music Information Retrieval.

Appendix A

Mapping rules from mood keywords to mood categories

Table A.1: All rules to map mood keywords of LiveJournal site to our 5 mood categories

Mood Keyword	**Mood Category**	**Mood Keyword**	**Mood Category**
tired	3;5	depressed	3
happy	2	bouncy	2
bored	3;5	excited	1
amused	4	accomplished	1
blah	3;5	awake	1
content	2	confused	3
cheerful	2	chipper	1
calm	2	anxious	5
contemplative	3	drained	3;5
sleepy	2;3	crazy	5
sick	3	annoyed	5
blank	3	sad	3
exhausted	3;5	aggravated	5
okay	2	pissed off	5
cold	3	lonely	3
loved	2	ecstatic	1
good	2	mellow	2
hopeful	1	crushed	3;5
crappy	3;5	frustrated	3;5
stressed	5	energetic	1
relaxed	2	hyper	1
thoughtful	3	bitchy	5
giddy	4	weird	4