NCCCS-2012

# PROBABILITY BASED PLAYLIST GENERATION BASED ON MUSIC SIMILARITY AND USER CUSTOMIZATION

*Karan Kumar Budhraja, Ashutosh Singh, Gaurav Dubey, Arun Khosla*

# The Authors

- *List of Authors*
  - *Karan Kumar Budhraja, Ashutosh Singh, Gaurav Dubey, Arun Khosla*
- *Affiliation*
  - *National Institute of Technology, Jalandhar, India*

# About the Presenter

- Name
  - Karan Kumar Budhraja
- Qualifications
  - B.Tech. from National Institute of Technology, Jalandhar (2010)
- Other Contributions
  - “User Customized Playlist Generation Based on Music Similarity” (NCCCS 2012)

# Introduction

- Large music databases
  - Playlist generation helpful
- Songs (N) considered as nodes of a graph
  - Similarity between two songs related to weight of edge
  - Exhaustive approach has $O(N^2)$ calculations
- Probability based candidate solutions
  - Calculations greatly reduced

# Musical Attributes

- Classification of attributes
  - Temporal, spectral, perceptual, harmonic, statistical
- Mel Frequency Cepstrum Coefficients (MFCC) used
  - Perceptual feature
  - Coefficient values translated using Gaussian Mixture Model (GMM)

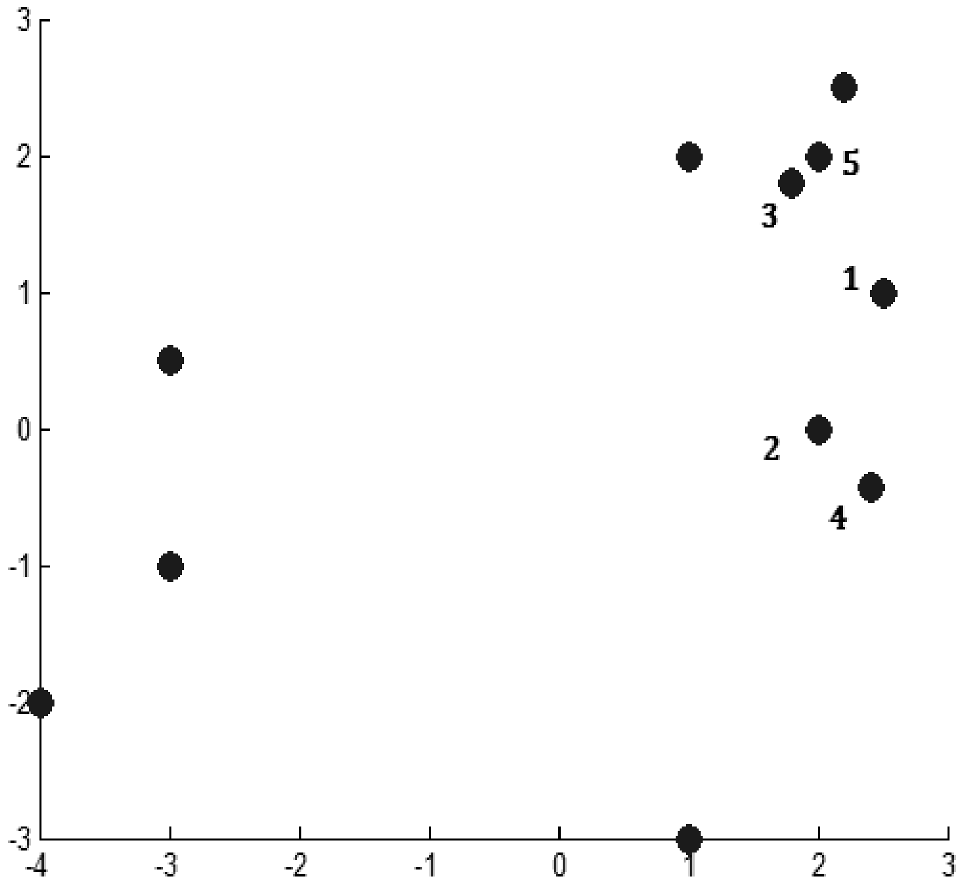
# Song Selection Methodology

- Song Similarity
  - Each song represented in Euclidian space
  - Considered as vectors
  - Cosine similarity based on angle between two vectors
    - Does not distinguish magnitudes
    - Distance based term added to check this

$$Similarity = \cos\theta - \frac{\tan^{-1}(d)}{p}$$

# Song Selection Methodology

Even though the items mapped to points 2 and 3 are close to the item mapped to point 1, if considered as a pair, they are both individually closer to other items (mapped to points 4 and 5 respectively).



# Song Selection Methodology

- Exhaustive Suggestion Algorithm
  - Each song considered for similarity calculation
  - N-1 computations per iteration

# Song Selection Methodology

- Proposed Suggestion Algorithm
  - Select only k (<<N) candidates
  - Selected on probability basis
    - Uniform at start of program
  - *Pheromone* used to track history
    - Less repetition

# Song Selection Methodology

- Feedback
  - May be positive or negative
  - Range from +1 to -1
  - If feedback is 1, we use suggested song as next reference
    - Songs similar to a song not always similar to each other

# Song Selection Methodology

- System Modification
  - Probability Modification
    - Positive feedback
      - Selection  probability for current song increased
      - Probability for other candidates decreased
    - Negative feedback
      - Selection  probability for current song decreased
      - Probability for other candidates increased

# Song Selection Methodology

- System Modification
  - Weight Modification
    - Specific to each attribute
    - Positive feedback
      - Attributes with more difference in values given less weight
      - These are less important attributes for the user
    - Negative feedback
      - Attributes with more difference in values given more weight
      - These are more important attributes for the user

# Song Selection Methodology

- System Modification
  - Pheromone Modification
    - Evaporation after each iteration
    - Exponential decay

# Evaluation

- Evaluation using precision
  - Fraction of songs suggested that suit given context
  - Songs grouped using fuzzy sets
  - Precision calculated per iteration
    - +1 if suggestion has certain membership value in fuzzy set corresponding to base song
    - -1 otherwise

# Evaluation

- Results

  Precision values for song suggestion over 10 runs of the algorithm. It should be noted that the systems were taken from their initial state where nothing has been learned

| Run | Exhaustive Algorithm Precision | Probabilistic Algorithm Precision |
|---:|---:|---:|
| 1 | 0.7 | 0.2 |
| 2 | 0.8 | 0.1 |
| 3 | 0.7 | 0.1 |
| 4 | 0.8 | 0.3 |
| 5 | 0.7 | 0.1 |
| 6 | 0.8 | 0.1 |
| 7 | 0.7 | 0.1 |
| 8 | 0.8 | 0.1 |
| 9 | 0.7 | 0.1 |
| 10 | 0.8 | 0.2 |

# Evaluation

- Discussion
  - Fluctuations attributed to system modifications
  - Probabilistic system degrades precision
    - Due to simplistic nature of modifications

- Future Work
  - Number of candidate songs may vary
  - Variants of ACO may be considered for adaptation
  - Modifications may be made complex

# Related Work

- Alternative version uses deterministic method
  - Songs divided into groups
  - Calculation reduced
  - Produced better precision values than current system

# Conclusion

- Algorithm for playlist generation proposed
- User specific customization
  - Sensitivity to attributes
- Much reduction in computation
  - Loss in precision
- System must be enhanced for better performance

# Bibliography (selected)

- S. Gunasekaran, K. Revathy: Automatic Recognition and Retrieval of Wild Animal Vocalizations. In Int. Journal of Computer Theory and Engineering, Vol.3, No.1, pp. 1793-8201 (2011)
- A. Rauber; E. Pampalk; D. Merkl: The SOM-enhanced JukeBox: Organization and Visualization of Music Collections Based on Perceptual Models. In Journal of New Music Research, Vol.32, No.2, pp. 193-210 (2003)
- M. Alghoniemy, A. Tewfik: A Network Flow Model for Playlist Generation In Proc. IEEE Int. Conf. on Multimedia and Expo (2001)
- E. Pampalk, A. Flexer, G. Widmer: Improvements of audio based music similarity and genre classification. In Proc. of the 6th Int. Symposium on Music Information Retrieval (2005)
- S. M. Lloyd, M. Sandler: Automatic Playlist Generation and Music Library Visualisation with Timbral Similarity Measures. In Queen Mary University of London (2009)
- E. Pampalk, G. Widmer: Computational Models of Music Similarity and their Application in Music Information Retrieval. PhD thesis. In Vienna University of Technology (2006)
- M. Gasser, E. Pampalk, M. Tomitsch: A Content-Based User-Feedback Driven Playlist Generator and its Evaluation in a Real-World Scenario, In Proc. of the Audio Mostly 2007 Conference on Interaction with Sound, Ilmenau, Germany (2007)

# Acknowledgement

The concept of evolutionary computing was introduced to us by our professor, Dr. Arun Khosla along with many newly emerging nature-inspired paradigms of soft computing. We thank him for familiarizing us about many intriguing domains and inspiring us with his ideas, knowledge and guidance. We hope to grow more on the platform provided by him. We would also like to thank the anonymous reviewers for their constructive comments.