Knowledge Extraction and Representation Learning for Music Recommendation and Classification

Sergio Oramas Martín

TESI DOCTORAL UPF / 2017

Director de la tesi:

Dr. Xavier Serra Casals

Dept. of Information and Communication Technologies

Universitat Pompeu Fabra, Barcelona, Spain

Dissertation submitted to the Department of Information and Communication Technologies of Universitat Pompeu Fabra in partial fulfillment of the requirements for the degree of

DOCTOR PER LA UNIVERSITAT POMPEU FABRA

Music Technology Group (http://mtg.upf.edu), Department of Information and Communication Technologies (http://www.upf.edu/dtic), Universitat Pompeu Fabra (http://www.upf.edu), Barcelona, Spain.

The doctoral defense was held on .... at the Universitat Pompeu Fabra and scored as ....

Dr. Xavier Serra Casals
(Thesis Supervisor)
Universitat Pompeu Fabra (UPF), Barcelona, Spain

Dr. Markus Schedl
(Thesis Committee Member)
Johannes Kepler University, Linz, Austria

Dr. Emilia Gómez
(Thesis Committee Member)
Universitat Pompeu Fabra (UPF), Barcelona, Spain

Dr. Brian Whitman
(Thesis Committee Member)
Spotify, New York, USA

a Olivia y Chiara, la luz en el camino...

Acknowledgements

First of all, I would like to thank my supervisor, Dr. Xavier Serra, for giving me the opportunity to work in this fantastic environment, the Music Technology Group, and for his wise advises. Also, I want to give special thanks to Paco Gomez for teaching me how to be a researcher. This thesis does not have a specific co-supervisor, but along this journey I have met three great researchers and better persons who have helped me through my PhD and without whom this work would have not been possible, Mohamed Sordo, Vito Claudio Ostuni, and Oriol Nieto. Special thanks also to Frederic Font for all these years sharing office, research, music, and friendship. A very important element of this thesis has been my collaboration with the TALN group. Everything started sharing pizza and water and now we have a lot of papers together and thousands of ideas. It was great collaborating and partying with Luis Espinosa-Anke and Francesco Barbieri. Also special thanks to Horacio Saggion for his wise advises, and Tommaso Di Noia, Aonghus Lawlor, and Michael O'Mahony for hosting me during my research stays at Politecnico de Bari and University College of Dublin. A special mention to Ichiro Fujinaga and Susan Weiss, who strongly believed in my research line. Also to my COFLA mates Emilia Gómez, Joaquin Mora, and José Miguel Díaz-Báñez, thanks for considering me a researcher from the first day. To my Pandora managers Andreas Ehmann, Oscar Celma, and Fabien Gouyon, who trusted me and gave me the opportunity to apply my research in the real world.

I also want to thank other amazing MTG people who shared with me knowledge and laughs throughout these years. In no specific order, thanks to the SIC-refugio team Sebastian Mealla, Álvaro Sarasúa, Panos Papiotis. To Rafael Caro, my musicologist mate. To Oriol Romaní and Juanjo Bosh for so much fun. To Alastair Porter and Dmitry Bogdanov for their experience and friendship. To Andrés Ferraro who shared my vision. To Gopala Koduri and our conversations about semantics. To Jordi Pons for initiating me in the deep learning cult. To my new roommates Eduardo Fonseca, Xavier Favory, and former ones Gerard, Dara, Hector, Giuseppe, and Albin. To more great people I met in the MTG, Cárthach, Ángel, Dani, Marius, Alfonso, Sergio, Zacharias, Olga, Perfe, Sergi, Rong, and Georgi. To former members, Sankalp, Sertan, Sercan, Ajay, Swapnil, Joan, Justin, Jordi, Julian, Julio, Carles, Nadine, and Martí. To Joan and Miguel (TALN), Humberto, and Manuel. To my Pandora mates Massimo, Theo, Chun, and Andreu. To the Supertropical doctoral consortium. To all my friends. Sorry if I forgot to mention anyone.

Also thanks to Aurelio, Lydia, Sonia, Cristina, Magda, Jana, and Alba for

helping me with the administrative work. Special thanks to Obra Social "La Caixa" and their fellowship program, to believe in me and support this research.

Last but not least, this work would never have been possible without the support of my wife Chiara, who followed me to Barcelona and helps me to pursue my dreams. Also very special thanks to my lovely Olivia, I started my PhD and my paternity at the same time, and I am pretty sure these have been the happiest years of my life, until now... To my little one on the way :) Finally, many thanks to my parents, brothers, and sisters, who helped me so much along my whole life, I am here thanks to them.

This thesis has been carried out at the Music Technology Group of Universitat Pompeu Fabra (UPF) in Barcelona, Spain, from October 2013 to September 2017, including a stay at Politecnico di Bari, Italy, from September 2014 to October 2014, at the Insight Centre for Data Analytics of University College of Dublin (UCD), Ireland, from May 2015 to July 2015, and at Pandora Media Inc., USA, from June 2016 to September 2016. It has been supervised by Dr. Xavier Serra. The work described in Chapter 4 is a joint effort by the author of this dissertation, and Luis Espinosa-Anke, both researchers in the Department of Information and Comunication Technologies of the Universitat Pompeu Fabra, Barcelona. Parts of the work described in Chapter 7 have been conducted in collaboration with Vito Claudio Ostuni, during the research stay at Politecnico di Bari. Work in some parts of this thesis has also been carried out in collaboration with other researchers, including Mohamed Sordo, Oriol Nieto, Francesco Barbieri, Horacio Saggion, Tommaso Di Noia, Francisco Gómez, and Aonghus Lawlor. The work in this thesis has been mainly supported by Obra Social "La Caixa" under their doctoral fellowship program, and by the Spanish Ministry of Economy and Competitiveness under the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502). The research stay at UCD has been also funded by the Keystone COST Action IC1302 under grant number SFI/12/RC/2289. Parts of the work presented in this thesis were partly supported by the European Research Council under the European Union's Seventh Framework Program, as part of the CompMusic project (ERC grant agreement 267583), and by the COFLA2 research project (Proyectos de Excelencia de la Junta de Andalucia, FEDER P12-TIC-1362).

Abstract

Music content creation, publication and dissemination has changed dramatically in the last few decades. Huge amounts of information about music are being published daily in online repositories such as web pages, forums, wikis, and social media. However, most of this content is still unusable by machines due to the fact that it is mostly created by humans and for humans. Furthermore, online music services currently offer ever-growing collections with tens of millions of music tracks. This vast availability has posed two serious challenges. First, how can a musical item be properly annotated and classified within a large collection? Second, how can a user explore or discover preferred music from all of the available content? In this thesis, we address these two questions by focusing on the semantic enrichment of descriptions associated to musical items (e.g., artists biographies, album reviews, metadata), and the exploitation of the heterogeneous data in large music collections (e.g., text, audio, images). To this end, we first focus on the problem of linking music-related texts with online knowledge repositories via entity linking, and on the automated construction of music knowledge bases via relation extraction. Then, we investigate how extracted knowledge may impact recommender systems, classification approaches, and musicological studies. We show how modeling semantic information helps to outperform purely text-based approaches in artist similarity and music genre classification, and achieves significant improvements with respect to state-of-the-art collaborative algorithms in music recommendation, while promoting long-tail recommendations. Next, we focus on learning new data representations from multimodal content using deep learning architectures. Following this approach, we address the problem of cold-start music recommendation by combining audio and text. We show how the semantic enrichment of texts and the combination of learned data representations improve the quality of recommendations. Moreover, we tackle the problem of multi-label music genre classification from audio, text, and images. Experiments show that learning and combining data representations yields superior results. As an outcome of this thesis, we have collected and released six different datasets and two knowledge bases. Our findings can be directly applied to design new algorithms for tasks such as music recommendation, and more specifically the recommendation of music from novel and unknown artists, which can potentially have an impact in the music industry. Although our research is motivated by particularities of the music domain, we believe that the proposed approaches can be easily generalized to other domains.

Resumen

La creación, publicación y diseminación de contenido musical ha cambiado radicalmente en las últimas décadas. Por un lado, grandes cantidades de información son publicadas diariamente en páginas web, fórums, wikis y redes sociales. Sin embargo, la mayor parte de estos contenidos son aún incomprensibles computacionalmente, ya que son creados por y para humanos. Por otro lado, los servicios de música online ofrecen inagotables catálogos con millones de canciones. Esta disponibilidad presenta dos desafíos. Primero, ¿cómo clasificar adecuadamente un ítem musical en una gran colección? Segundo, ¿cómo puede un usuario explorar o descubrir música de su agrado entre todo el contenido disponible? En esta tesis, abordamos estas cuestiones centrándonos en el enriquecimiento semántico de descripciones de ítems musicales (biografías de artistas, reseñas musicales, metadatos, etc.), y en el aprovechamiento de datos heterogéneos presentes en grandes colecciones de música (textos, audios e imágenes). Para ello, primero nos centramos en el problema de enlazar textos musicales con bases de conocimiento online y en la construcción automatizada de bases de conocimiento musical. Luego investigamos cómo el conocimiento extraído puede impactar en sistemas de recomendación y clasificación, además de en estudios musicológicos. Mostramos cómo el modelado de información semántica contribuye a mejorar los resultados con respecto a métodos basados solo en texto, tanto en similitud de artistas como en clasificación de géneros musicales, y a conseguir mejoras significativas en recomendación de música con respecto a algoritmos de referencia, mientras a su vez se promueven recomendaciones de ítems menos populares. A continuación, investigamos el aprendizaje de nuevas representaciones de datos a partir de contenidos multimodales utilizando redes neuronales, y lo aplicamos a los problemas de recomendar música nueva y clasificar géneros musicales con múltiples etiquetas, mostrando que el enriquecimiento semántico y la combinación de representaciones aprendidas produce mejores resultados. Uno de los frutos de esta tesis es la publicación de seis datasets y dos bases de conocimiento. Además, nuestros descubrimientos pueden ser directamente aplicados al diseño de nuevos algoritmos de recomendación de música, y más concretamente, de artistas nuevos y desconocidos, lo cual tiene potencial impacto en la industria musical. Aunque nuestra investigación está motivada por las particularidades del dominio de la música, creemos que las metodologías propuestas pueden ser fácilmente generalizables a otros dominios.

Resum

La creació, publicació i disseminació de contingut musical ha canviat radicalment en les últimes decades. Per una banda, una gran quantitat d'informació es publica diàriament a pàgines web, fòrums, wikis i xarxes socials. Tot i això, la majoria d'aquest contingut és encara incomprensible computacionalment degut a que es crea per i per als humans. Per una altra banda, els serveis de música online ofereixen inagotables catàlegs de milions de cançons. Aquest àmplia disponibilitat ofereix dos reptes; com es pot anotar i classificar adequadament un ítem musical en una col·lecció molt gran? I en segon lloc; com pot un usuari descobrir música del seu gust entre tot el contingut disponible? En esta tesi abordem aquestes qüestions centrant-nos en l'enriquiment semàntic de descripcions d'ítems musicals (biografies d'artistes, ressenyes musical, metadades, etc.) i en la exploració de dades heterogènies en grans col·leccions de música (textos, àudio i imatges). Ens centrem en primer lloc en el problema d'enllaçar textos musicals amb bases de coneixement i en la construcció automatitzada d'aquestes bases de coneixement. Tot seguit investiguem quin impacte pot tenir el coneixement extret anteriorment en sistemes de recomanació y classificació, a més de en estudis musicològics. Mostrem a continuació com el modelat de la informació semàntica contribueix a millorar els resultats obtinguts amb métodes basats només en text, tant pel que fa a la similitud d'artistes com a la classificació de gèneres musicals. Aquest modelat també aconsegueix millores significatives en recomanació de música, en comparació a algorismes de referència, alhora que es promouen recomanacions d'ítems menys populars. A continuació investiguem l'aprenentatge de noves representacions de les dades a partir de diverses modalitats de contingut fent servir xarxes neuronals. Així encarem el problema de recomanar nova música combinant text i àudio. Mostrem com l'enriquiment semàntic dels textos i la combinació de representacions apreses millora la qualitat de les recomanacions. A més, abordem el problema de classificació de gèneres musicals amb múltiples etiquetes utilitzant text, àudio i imatges. Els experiments mostren que l'aprenentatge i la combinació de representacions de dades produeixen millors resultats. Un dels fruits d'aquesta tesi es la publicació de sis datasets i dues bases de coneixement. A més, els nostres descobriments es poden aplicar directament al disseny de nous algorismes de recomanació de música i, més concretament, a la recomanació d'artistes nous o desconeguts, de tal manera que té potencial per generar impacte a la indústria. Encara que la motivació d'aquesta investigació són les particularitats del domini de la música, creiem que les metodologies proposades poden ser fàcilment generalitzables a altres dominis.

Contents

Abstract ix

Contents xv

List of figures xxi

List of tables xxiii

1 Introduction 1

1.1. Motivation . . . 1

1.2. Methodologies . . . 2

1.2.1. Why knowledge extraction? . . . 3

1.2.2. Why representation learning? . . . 4

1.3. Challenges . . . 5

1.3.1. Music classification . . . 5

1.3.2. Music recommendation . . . 6

1.4. Objectives and outline of the thesis . . . 7

2 Background 11

2.1. Introduction . . . 11

2.2. Natural Language Processing . . . 11

2.2.1. Knowledge base construction . . . 12

2.2.2. Music knowledge bases . . . 13

2.2.3. Entity linking . . . 14

2.2.4. Relation extraction . . . 15

2.3. Recommender Systems . . . 16

2.3.1. Knowledge-based approaches . . . 17

2.3.2. Deep learning approaches . . . 18

2.4. Music Information Retrieval . . . 18

2.4.1. Text-based approaches . . . 18

2.4.2. Knowledge-based approaches . . . 19

2.4.3. Music classification . . . 20

2.4.4. Artist similarity . . . 21

2.4.5. Music recommendation . . . 22

I Knowledge Extraction from Text 23

3 Linking Music-related Texts to Knowledge Bases 25
3.1. Introduction 25
3.2. Music entity linking 26
3.3. ELVIS 27
3.3.1. Argumentum ad populum in entity linking 27
3.3.2. 'Translating' entity linking formats 28
3.4. From Last.fm to ELMD 29
3.4.1. Data enrichment 30
3.5. Evaluation 32
3.6. Extending ELMD 34
3.7. Gold standard dataset 35
3.8. Conclusion 36

4 Automated Construction of Music Knowledge Bases 37
4.1. Introduction 37
4.2. Method 38
4.2.1. Notation 38
4.2.2. Morphosyntactic preprocessing 39
4.2.3. Semantic processing: entity linking 40
4.2.4. Syntactic semantic integration 41
4.2.5. Relation extraction and filtering 42
4.2.6. Dependency-based loose clustering 43
4.2.7. Scoring 44
4.3. Experimental setup 46
4.3.1. Source dataset 46
4.3.2. Extracted knowledge bases 47
4.4. Experiments 48
4.4.1. Quality of entity linking 48
4.4.2. Quality of relations 50
4.4.3. Coverage of the extracted knowledge base 52
4.4.4. Interpretation of music recommendations 53
4.5. Conclusion 54

5 Applications in Musicology 57
5.1. Introduction 57
5.2. Building culture-specific knowledge bases: the flamenco case . . 58
5.2.1. Flamenco music 58
5.2.2. FlaBase 59
5.2.3. Content curation 60
5.2.4. Knowledge extraction 63
5.2.5. Looking at the data 65

5.3. Diachronic study of music criticism 67
5.3.1. Dataset 68
5.3.2. Sentiment analysis 69
5.3.3. Experiments 70
5.4. Conclusions 74

6 Semantic Enrichment for Similarity and Classification 75
6.1. Introduction 75
6.2. Artist similarity 76
6.2.1. Entity linking 76
6.2.2. Knowledge representation 77
6.2.3. Similarity approaches 79
6.2.4. Experiments 80
6.2.5. Results and discussion 82
6.3. Music genre classification 84
6.3.1. Dataset description 85
6.3.2. Linguistic processing 85
6.3.3. Features 85
6.3.4. Baseline approaches 86
6.3.5. Experiments 87
6.3.6. Results and discussion 87
6.4. Conclusion 88

7 Sound and Music Recommendation with Knowledge Graphs 91
7.1. Introduction 91
7.2. Knowledge enrichment via entity linking 93
7.3. Recommendation approach 94
7.3.1. Explicit feature mappings for graph-based item representations 97
7.3.2. Feature combination 99
7.4. Experimental evaluation 100
7.4.1. Datasets description 101
7.4.2. Experiment settings 102
7.4.3. Sound recommendation experiment 103
7.4.4. Music recommendation experiment 106
7.5. Conclusion 109

II Representation Learning from Multimodal Data 111

8 Cold-start Music Recommendation 113
8.1. Introduction 113
8.2. Recommendation approach 114

8.3. Learning artist representations from text 116
8.3.1. Semantic enrichment 116
8.3.2. Word embeddings 117
8.4. Learning track representations from audio 117
8.5. Multimodal fusion 118
8.6. Experiments 119
8.6.1. Dataset 119
8.6.2. Artist recommendation 119
8.6.3. Song recommendation 121
8.7. Conclusions 122

9 Multi-label Music Genre Classification 125
9.1. Introduction 125
9.2. Multimodal dataset 126
9.2.1. Genre labels 127
9.3. Multi-label classification 127
9.3.1. Labels factorization 128
9.3.2. Evaluation metrics 129
9.4. Album genre classification 129
9.4.1. Audio-based approach 130
9.4.2. Text-based approach 130
9.4.3. Image-based approach 131
9.4.4. Multimodal approach 131
9.5. Experiments 132
9.5.1. Audio classification 133
9.5.2. Text classification 134
9.5.3. Image classification 135
9.5.4. Multimodal classification 136
9.6. Conclusions 136

10 Summary and future perspectives 139
10.1. Introduction 139
10.2. Summary of contributions 140
10.2.1. Scientific contributions 140
10.2.2. Datasets 142
10.2.3. Knowledge bases 142
10.2.4. Software 143
10.2.5. Publications 143
10.3. Directions for future research 143

Bibliography 149

Appendix A: Publications by the author 171

Appendix B: Datasets, Knowledge Bases, and Software 175

Datasets . . . 175

Knowledge bases . . . 175

Software . . . 176

List of figures

1.1. Long-tail distribution. 6
1.2. Thesis overview. 8

3.1. ELVIS Workflow. 28
3.2. Number of entities and precision of the manual evaluation. 31
3.3. ELMD Overview. Number of annotations, confidence score, and precision values at different confidence score thresholds. 34

4.1. Example sentence with dependency parsing tree. 39
4.2. Semantic integration on syntactic dependencies. 42
4.3. Example of a parsed relation pattern $p \in \mathcal{P}$ and a valid cluster pattern (bold). 44
4.4. Percentage of triples and relation patterns from KBSF-ft. 47
4.5. F-measure of the entity linking systems at different thresholds. . . 49
4.6. Precision of relations. 52
4.7. User interface for the music recommendation experiment. 55

5.1. Ontology schema 60
5.2. Selected data sources 61
5.3. F-measure for different values of θ 63
5.4. FlaBase distributions. 67
5.5. Artists by decade of birth 67
5.6. Overview of the opinion mining and sentiment analysis framework. 70
5.7. A sentence from a sample review annotated with opinion and aspect pairs. 70
5.8. Sentiment (a, c, and d) and rating (b) averages by review publication year; Kernel density estimation of the distribution of reviews by year (e); GDP trend in USA from 2000 to 2014 (f) 72
5.9. Sentiment (a), rating (b), and sentiment by genres (c) averages by album publication year. 73

6.1. Workflow of the proposed method. 76
6.2. Knowledge graphs. 78
6.3. Percentage of accuracy of the different approaches. 88
6.4. Confusion matrices. 89

7.1. Portion of the final knowledge graph enriched with WordNet and DBpedia 95

7.2. An example of 3-hop item neighborhood graph for the item i. 97
7.3. Precision-Recall, Novelty, and Aggregate Diversity plots in Freesound dataset 105
7.4. Precision-Recall, Novelty, and Aggregate Diversity plots in Last.fm dataset 108

8.1. Model architecture. 115

9.1. t-SNE of album factors. 133
9.2. Particular of the t-SNE of randomly selected image vectors from five of the most frequent genres. 135

List of tables

3.1. Equivalence of types between Last.fm and DBpedia. Yago, Schema, and DBpedia refer to the correspondent ontologies. 30
3.2. Agreement examples of ELVIS. 31
3.3. Precision and coverage of ELMD mapping to DBpedia. 32
3.4. Statistics of the linked entities in ELMD. 32
3.5. Statistics of the extended ELMD corpus. 35
3.6. Percentage of linked entities in the extended ELMD corpus. 35
3.7. Statistics of the ELMD gold standard corpus. 36

4.1. Type mapping. 41
4.2. Complete set of patterns used in the filtering heuristics 43
4.3. Regular expressions for dependency paths in cluster patterns. . . . 44
4.4. Example of a relation cluster. 45
4.5. Statistics of all the extracted KBs 48
4.6. Precision and recall of the entity linking systems considered. . . . 49
4.7. Precision and recall of the entity linking systems considered. . . . 50
4.8. Top-5 most frequent entities by type and tool. 51
4.9. Number of triples with labeled relations in the different KBs. . . . 53

5.1. Precision, Recall and F-measure of entity linking approaches. . . . 65
5.2. PageRank Top-5 artists by category. 66
5.3. Precision values of artist relevance ranking. 66
5.4. Number of albums by genre with information from the different sources in MARD. 69

6.1. Precision and nDCG for Top-N artist similarity in the MIREX dataset. 83
6.2. Precision and nDCG for Top-N artist similarity in the Last.fm dataset. 83
6.3. Average genre distribution of the top-10 similar artists using the MIREX dataset. 84
6.4. Accuracy of the different classifiers. 87

7.1. Number of tags and keywords. 101
7.2. Accuracy, Novelty, and Aggregate Diversity results for different versions of the Freesound dataset. 103
7.3. Accuracy, Novelty, and Aggregate Diversity results for different versions of the Last.fm dataset. 107

8.1. DBpedia properties selected for each entity class. 116
8.2. Artist Recommendation Results. 120
8.3. Song Recommendation Results. 121

9.1. Top-10 most and least represented genres. 127
9.2. Results for Multi-label Music Genre Classification of Albums. . . . 132

CHAPTER 1

Introduction

1.1. Motivation

We are witnessing an unprecedented information explosion thanks to the dramatic technological advancement brought by the Information Age (Smith, 2009). This technological (r)evolution has enabled the release and publication of huge amounts of data into online repositories such as web sites, forums, wikis, and social media. Art and culture have benefited dramatically from this context, which allows potentially anyone with an available Internet connection to access, produce, publish, comment, or interact with any form of media.

In this context, music content creation, publication, and dissemination has changed dramatically. Online music services, such as Pandora, Spotify, Apple Music, Google Play, Deezer, Tidal, Amazon Music, or Soundcloud, benefit from this situation and currently offer ever-growing catalogs with dozens of millions of music tracks, which are in turn just one click away from millions of users. This vast availability of music poses two serious challenges: First, *how can a musical item be properly annotated and classified within a large collection?* Since manually managing these large libraries is not feasible due to size constraints, automatic methods for the annotation and classification of large-scale music collections have been an active area of research in recent years (Schedl et al., 2014). Second, *how can a user explore or discover preferred music from all the available content?* Traditionally, users have relied on their friends, their favorite music radio host, a music expert in their local retail store, etc. to obtain recommendations on artists or albums they might like. Although this traditional approach is still valid and used by many people, its ability to cover the vast amount of available music nowadays is seriously hindered. Therefore, automatic approaches to music recommendation have become necessary (Celma & Herrera, 2008).

Large music collections combine information from multiple data modalities, such as audio, images, text or videos. In addition, music collections can be enriched with user-generated content published online. However, as stated 17 years ago by Cohen & Fan (2000) “The main problem, of course, is that the bulk of information on the Web is designed to be read by humans, not by machines”. Nevertheless, this problem is far from being totally solved. In this context, Natural Language Processing is playing a key role, as one of its main lines of research is precisely to transform human readable content into machine readable data (Cowie & Lehnert, 1996).

The way human readable content and multimodal data present in large music collections is represented and combined in computational models poses numerous challenges. Artificial intelligence methods, such as machine learning, heavily rely on the choice of data representation. Therefore, finding representations that maximize the different explanatory factors of variation behind the data is a fundamental task. Traditional approaches rely on handcrafted features to represent the variability of the data, whereas more recently, and thanks to the raise of deep learning techniques, representation learning approaches have demonstrated their superiority in multiple domains (Bengio et al., 2013).

In this thesis, we address the aforementioned challenges of classification and recommendation of musical items in large music collections from two different standpoints: (1) extracting structured knowledge from music-related texts and further enriching this knowledge with semantic information present in online knowledge repositories, (2) learning new data representations from heterogeneous data using deep learning architectures and further combining these representations in multimodal networks. We hypothesize that such enriched knowledge and learned multimodal data representations are crucial for improving recommendation and classification algorithms.

1.2. Methodologies

This thesis is framed within the Music Information Retrieval (MIR) research tradition. MIR is a multidisciplinary field of research concerned with the extraction, analysis, and usage of information about any kind of music entity (e.g., song, artist, album) on any representation level (e.g., audio signal, symbolic MIDI, metadata) (Schedl, 2008). According to Schedl et al. (2013), the musical factors that influence human music perception can be categorized into *music content* and *music context.* Music context relates to all musical aspects that are not encoded in the audio signal, such as song lyrics, artist’s biography, album cover artwork, or music video clips, whereas music content is defined as human perceptual aspects that can be extracted from the audio signal. Following this distinction, research methodologies within the MIR community that deal with data modalities different from audio (e.g., text, images) are often called context-based approaches, whereas methodologies based on the analysis of the audio signal are called content-based approaches. Although we agree with this classification criteria, in this thesis we follow the nomenclature used in the Recommender Systems community (Ostuni et al., 2013) and consider *audio signal*, *text* (e.g., metadata, artist's biographies, song lyrics), and *images* (e.g., album cover artwork, artist's photographies) as different modalities of content.

1.2.1. Why knowledge extraction?

As pointed out by Humphrey et al. (2012), MIR approaches are typically based on a two-stage architecture of feature extraction and semantic interpretation (e.g., classification, regression, clustering, similarity ranking, etc.). Traditionally, MIR has been mainly focused on the use of features extracted from audio, and has not paid much attention to other data modalities. However, in recent years several studies have shown the benefits of using context-based and multimodal approaches (Schedl et al., 2014).

Audio features are often classified into low, mid, and high-level representations (Bello & Pickens, 2005). Low-level representations (e.g., *spectral flux, cepstrum, MFCCs*) are measured directly on the audio signal. Mid-level representations (e.g., chords, onsets) represent musical attributes extracted from the audio combining machine learning and musical knowledge. High-level representations (e.g., mood, form, genre) are related to human interpretations of the data, and are typically built on top of low and mid-level representations. The extraction and exploitation of features from these three representation levels have been widely studied in the MIR field (Casey et al., 2008).

Following this feature hierarchy, when dealing with textual data, we can also differentiate between low, mid, and high-level representations. Low-level representations (e.g., word frequencies, word co-occurrences, n-grams) are measured directly on text. Mid-level representations (e.g., part-of-speech tags, noun phrases) combine linguistic knowledge and statistical analysis of text corpora. High-level representations (e.g., syntactic dependencies, disambiguated named entities, semantic relations) involves a semantic understanding of text. In the context of MIR, most of the literature is focused on low-level and mid-level representations (Celma et al., 2006; Lamere, 2008; Whitman & Lawrence, 2002; Knees & Schedl, 2013), and very few in high-level ones (Tata & Di Eugenio, 2010; Knees & Schedl, 2011; Sordo et al., 2012). Little attention has been paid to the semantic of words or the context they are being used. Thus, the research in the MIR field, has not yet exploited the epistemic potential of text.

In the first part of this thesis, we focus on knowledge extraction and knowledge-based approaches. On the one hand, we work on new methodologies for the extraction of high-level semantic representations from music-related unstructured texts. On the other hand, we put the emphasis on the development of approaches that exploit these semantic representations in music recommendation and classification. In addition, we also study how semantic information may impact musicological studies.

1.2.2. Why representation learning?

As stated before, MIR approaches are commonly based on a two-stage architecture of feature extraction and semantic interpretation. In this context, data representations are generally obtained following a traditional feature extraction process, which involves a combination of music domain-knowledge, psychoacoustics, and audio engineering (Humphrey et al., 2012). This is known as feature engineering, and compensates the inability of traditional machine learning algorithms to extract the discriminative information of the data. However, it involves a labor-intensive human effort, and also all the different explanatory factors of variation behind the data are not represented (Bengio et al., 2013). Huge efforts have been put in the last two decades in the definition and extraction of audio features, which has given rise to comprehensive software libraries that assemble many of these feature extraction techniques (Bogdanov et al., 2013b; Mcfee et al., 2015).

Contrarily to feature engineering, representation learning (or feature learning) is a technique that allows a learning system to automatically discover the variation behind the data directly from raw signals. As shown by Humphrey et al. (2012), MIR approaches can benefit from the use of these learning approaches using deep neural networks. In a neural network, each hidden layer maps its input data to an inner representation that tends to capture a higher level of abstraction. This methodology has two main advantages. First, blurring the boundaries between the two-stage architecture, which implies fully automated optimization of both stages at once. Second, it results in general-purpose architectures that can be applied to different MIR problems and data modalities. In the last years, several works have been published where audio-based deep learning architectures have been applied to MIR tasks such as music recommendation (Van den Oord et al., 2013) and music classification (Choi et al., 2016a), among others. However, to the best of our knowledge, there is no multimodal system that makes use of deep learning approaches for music recommendation or music classification.

While in the first part of the thesis we use data representations based on traditional feature engineering approaches, in the second part of this thesis we focus on representation learning approaches using deep neural networks. We apply this methodology to different data modalities (audio, text, and images) and their combination, and in the context of music recommendation and classification.

1.3. Challenges

In this section we describe in detail the two main challenges addressed in this thesis: music classification and music recommendation.

1.3.1. Music classification

The advent of large music collections has posed the challenge of how to access the information in terms of retrieval, browsing, and recommendation. One way to ease the access of large music collections is to keep annotations of all music resources (Sordo, 2012). Annotations can be added either manually or automatically. However, due to the high human effort required for manual annotations, the implementation of automatic annotations processes has become a necessity.

We distinguish two ways of automatically enhancing annotations: (i) gathering annotations from external sources, and (ii) learning annotations from the collection's data. To address (i), information can be obtained from online knowledge repositories (e.g., Wikipedia, MusicBrainz), or extracted from collections of unstructured documents. This imposes the challenge of how to properly map collection's items with external entities. To address (ii), machine learning techniques can be applied over the collection's data. When annotations are learned from audio, this classification task is often called auto-tagging. However, annotations can be learned from different data modalities, such as album cover artworks, tags, editorial metadata, video clips, etc.

Among the different categories of annotations used in music collections, the most prototypical are: music genres, instruments, and moods. Music genre labels are useful categories to organize and classify songs, albums, and artists into broader groups that share similar musical characteristics. Music genres have been widely used for music classification, from physical music stores to streaming services. Automatic music genre classification thus is a widely explored topic (Sturm, 2012). However, almost all related work is concentrated in the classification of music items into broad genres (e.g., Pop, Rock), assigning a single label per item (Bogdanov et al., 2016). This is problematic since there may be hundreds of more specific music genres (Pachet & Cazaly, 2000), and these may not necessarily be mutually exclusive (i.e., a song could be Pop, and at the same time have elements from Deep House and a Reggae groove). As pointed out by McKay & Fujinaga (2006), annotations should allow more than one genre per recording, and information from several modalities should be combined for classification.

In this thesis, we focus on the problem of enriching annotations in music collections from the two above defined standpoints, i.e., gathering and learning. We study how semantic technologies may be useful to improve the annotations

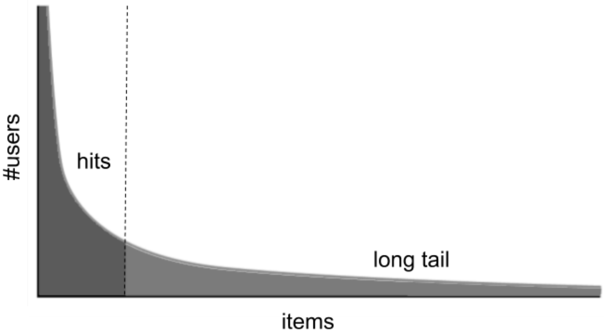


Figure 1.1: Long-tail distribution.

of musical items. In addition, we tackle the problem of multi-label music genre classification from different data modalities (i.e., audio, text, and images) and their combination.

1.3.2. Music recommendation

Information overload in modern Web applications challenges users in their decision-making tasks. Recommender systems have emerged in the last years as fundamental tools in assisting users to find, in a personalized manner, what is relevant for them in overflowing knowledge spaces (Ricci et al., 2011).

Music recommendation is a relatively young but continuously growing research topic, in both MIR and Recommender Systems communities. Although several research approaches and commercial systems have been proposed in the last decade, many of them are adaptations from other domains (Celma, 2010). Music has its own specificities with respect to other domains. For instance, a user may consume a musical item several times, or very different items according to the user context (e.g., working, dinning, exercising). Therefore, music recommendation is a challenging and still unsolved problem.

Although music online services make available almost all existing music, only a small percentage of these catalogs is actually consumed by the vast majority of users. Music consumption follows what is called a *long-tail* distribution (Celma, 2010) (see Figure 1.1). Therefore, one of the main challenges in music recommendation is how to make this niche of unknown musical items profitable. Moreover, as new music is continuously being created, new artists and releases appear every day. Hence, another important challenge in music recommender systems is how to deal with these new items, which is often called the *cold-start* problem.

To tackle these problems, it is crucial to have good item descriptions and to exploit all the available multimodal data (e.g., images, audio, texts, and videos). The web is full of user-generated content with relevant information about music, which has the potential to impact in the performance of music recommender systems. However, as stated before, this content is mostly unstructured and requires the application of knowledge extraction techniques that exploit the semantic of texts. In addition, up to now, audio content has been barely exploited in commercial recommender systems. However, thanks to the advent of novel deep learning approaches that raised the accuracy of audio-based recommendations (Van den Oord et al., 2013), audio may turn into a key factor in order to provide accurate long-tail recommendations.

Most research in music recommendation has been dedicated to developing algorithms that provide *good* and *useful* recommendations (Celma, 2010), neglecting the importance of the novelty and diversity of recommendations (Adomavicius & Kwon, 2012; Bellogín et al., 2010). In addition, very few approaches are able to provide explanations of the recommendations to the users (Passant & Raimond, 2008; Passant, 2010). According to Celma & Herrera (2008), giving explanations of the recommendations provides transparency to the recommendation process and increases the confidence of the user in the system.

In this thesis we further explore the music recommendation problem from three different perspectives. First, we investigate how information extracted from large collections of music-related documents may be useful to provide explanations of recommendations to users. Second, we tackle the problem of recommending long-tail items by enriching item descriptions with semantic information combined with user feedback data. Finally, we address the problem of cold-start music recommendations by combining different data modalities using deep neural networks.

1.4. Objectives and outline of the thesis

In the previous sections we have explained the motivations and context of our thesis. According to that, the main objective of this thesis is to improve the classification and recommendation of musical items in large music collections, with special emphasis on the promotion of novel and less popular items. To do so, we have addressed two different methodologies, one related to the extraction of structured knowledge from unstructured text sources and its further enrichment using information present in online knowledge repositories, and the other related to the learning of new data representations from multimodal data using deep learning architectures. The former approach may be also applied to other music related problems such as the creation of music knowledge repositories or the discovery of musicological knowledge. Therefore, in addition to the classification and recommendation problems, we have also addressed these linguistic and musicological challenges. In Figure 1.2, a diagram of this thesis is shown, organized according to the different approaches, methods, and ap-

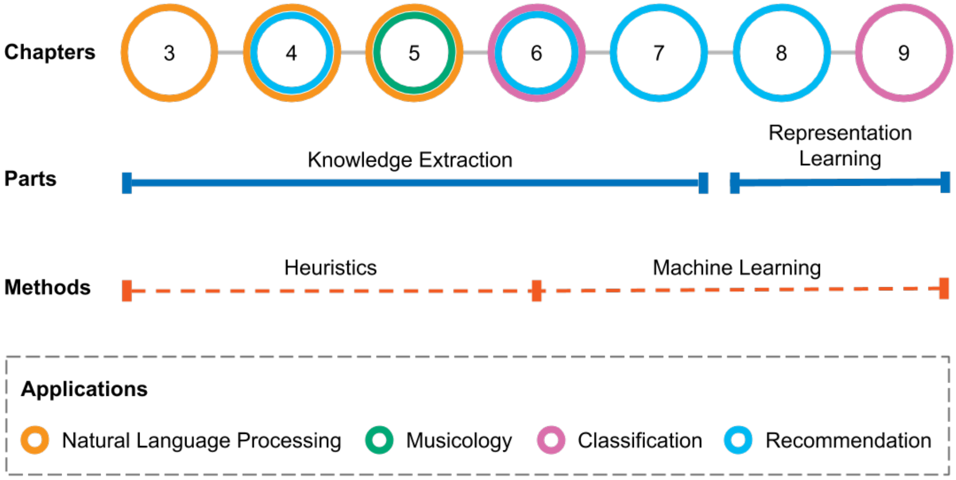


Figure 1.2: Thesis overview.

plications present in each chapter. Although this thesis is focused on the music domain, we strongly believe that the work we present can be easily adapted to other multimedia domains.

This thesis is structured as follows: Chapter 2 presents some background knowledge and related work on Natural Language Processing, Recommender Systems, and Music Information Retrieval. Hereafter, the work in this thesis is divided in two Parts: In Part I we explore different techniques and approaches to extract semantic information from unstructured music-related text sources. Then, these semantic representations are exploited in music classification, similarity, and recommendation problems. Within this part, Chapter 3 illustrates the problem of linking musical texts and knowledge repositories. In Chapter 4 we address the automatic generation of music knowledge bases from unstructured text sources. This chapter encloses with an experiment on explanations of music recommendations based on extracted knowledge. In Chapter 5, two experiments study the potential impact of knowledge extraction techniques in musicological studies. Chapter 6 presents the application of a semantic-based approach to music similarity and classification, whereas Chapter 7 addresses the problem of long-tail sound and music recommendations by enriching item descriptions with semantic information. Then, in Part II an approach to learn data representations from different data modalities using deep neural networks is applied to music recommendation and classification problems. In Chapter 8 we address the problem of cold-start music recommendations using audio and text. Next, in Chapter 9 we apply a similar representation learning approach for multi-label classification of music genres using audio, text, and images. At the end of each chapter, we include a focused discussion about the relevant results and conclusions. We conclude this thesis in Chapter 10 with a summary of our work, our main conclusions, and a discussion about open issues and future perspectives.

CHAPTER 2

Background

2.1. Introduction

The literature review presented in this chapter is divided in three parts. Each part is focused on one of the main fields of research where this thesis is framed: Natural Language Processing (NLP), Recommender Systems (RS), and Music Information Retrieval (MIR). First, we summarize existing work on several areas of Natural Language Processing, with special focus on its application to the music domain (Section 2.2). We define what a knowledge base (KB) is and the different existing types. In addition, we enumerate the available knowledge bases that contain music information. Then, we explain what entity linking is and briefly describe some state-of-the-art systems. Additionally, we outline different existing approaches for relation extraction. Next, we briefly explain the main methodologies used in Recommender Systems, and illustrate the available work on knowledge-based approaches (Section 2.3). Finally, we dig into the existing literature on Music Information Retrieval (Section 2.4). More specifically, we first review the state of the art in text-based and semantic-based approaches. Then, we focus on three specific MIR tasks: music genre classification, artist similarity, and music recommendation.

2.2. Natural Language Processing

Natural Language Processing (NLP) is a field of study that focuses on the interactions between human language and computers. One of the main subtopics of NLP is Natural Language Understanding (NLU), which deals with machine reading comprehension. Knowledge Representation and Reasoning is a key enabler of Intelligent Systems (Suchanek et al., 2007), and plays an important role in NLU (Baral & De Giacomo, 2015). In this thesis, we focus on an important aspect of NLU, which is *how to make sense* of the data that is generated and published online on a daily basis. This data is mostly produced in human-readable format, which makes it unsuitable for automatic processing. Considering that deep understanding of natural language by machines seems to be very far off (Cambria & White, 2014), there is great interest in formalizing unstructured data, and knowledge bases are a paradigmatic example of large-scale content processed to make it machine readable.

Information extraction is the task of automatically extracting structured information from unstructured or semi-structured text sources. It is a widely studied topic within the NLP research community (Cowie & Lehnert, 1996). A major step towards understanding language is the extraction of meaningful terms (entities) from text as well as relationships between those entities. This statement involves two different tasks. First, the identification and categorization of entity mentions. This task is called named entity recognition. However, when this task involves a latter step of disambiguation of entities against a knowledge base it is called named entity disambiguation or entity linking. Second, the identification of relevant semantic relations between entities. This task is called relation extraction.

The work described in this thesis strongly focuses on the exploitation of linguistic and semantic properties of text collections. For this reason, we deem relevant to cover related work in the following areas: (1) knowledge base construction and curation; (2) music knowledge bases; (3) entity linking, and (4) relation extraction.

2.2.1. Knowledge base construction

We may define a knowledge base as a repository of knowledge organized in a predefined taxonomic or ontologic structure, potentially compatible with other knowledge bases, thus contributing to the Linked Open Data initiative[1]. These knowledge bases may be designed to represent unconstrained knowledge, or a single domain of interest. This representation is formalized either manually, automatically, or with a combination of both.

We understand language by making sense of the connections between words, concepts, phrases, and thoughts (Havasi et al., 2007). Knowledge bases constitute a resource for encapsulating this knowledge. Previous efforts on knowledge base construction may be characterized as: (1) handcrafted knowledge bases; (2) integrative projects (automatic in design, but reliant on manually validated data); and (3) fully automatic, also in the relation extraction process.

Among the first group, the best known is probably `WordNet` (Miller, 1995), a lexical database which groups concepts in “synonym sets”, and encodes predefined relations among them such as *hyponymy/hypernymy*, *meronymy*, *holonymy*, or *instantiation*. Manually constructed knowledge bases, however,

[1] `http://linkeddata.org/`

are mostly developed in specific domains, where the degree of ambiguity is lower and there is more availability of trained knowledge engineers.

Next, integrative projects are probably the most productive, as they are the most ambitious attempts in terms of content coverage and community involvement, not only users, but also contributors. Examples of these include Yago (Suchanek et al., 2007), an automatically created knowledge base derived from integrating Wikipedia and WordNet; DBpedia (Lehmann et al., 2014), a collaboratively maintained project aimed at exploiting information present in Wikipedia, both structured and in free text; Freebase (Bollacker et al., 2008), also a collaborative effort mainly based on extracting structured knowledge from Wikipedia; or BabelNet (Navigli & Ponzetto, 2012), a semantic network which started as a seamless integration of Wikipedia and WordNet, and today constitutes the largest multilingual repository of words and senses.

With regard to the third group we refer to approaches where knowledge is obtained automatically. Endeavours in this area include TextRunner (Banko et al., 2007), widely regarded as the first Open Information Extraction (OIE) system; ReVerb (Fader et al., 2011), particularly designed to reduce noise while keeping a wide coverage, thanks in part to a set of syntactic and lexical constraints; NELL (Carlson et al., 2010b), which incorporates semantic knowledge in the form of a handcrafted taxonomy of entities and relations; PATTY (Nakashole et al., 2012) and WiseNet (Moro & Navigli, 2012, 2013), in which a shared vision to integrate semantics is applied both at the entity and relation level; DefIE (Bovi et al., 2015b), a recent development in OIE tested on the whole set of BabelNet glosses; and KB-Unify (Bovi et al., 2015a), not an actual information extraction implementation, but rather a unification framework for knowledge bases.

2.2.2. Music knowledge bases

MusicBrainz and Discogs are two paramount examples of manually curated music knowledge bases. They are not strictly knowledge bases, but open music encyclopedias of music metadata, which are built collaboratively and are openly available. MusicBrainz, in addition, has been published as Linked Data by the LinkedBrainz project[2].

As for generic knowledge bases based on Wikipedia, such as the ones described earlier, these include a remarkable amount of music data, such as artist, album, and song biographies; definitions of musical concepts and genres; and articles about music institutions and venues. However, their coverage is biased towards the best-known artists, and towards products from Western culture. Finally, let us refer to the notable case of Grove Music Online[3], a music encyclopedia

[2] http://linkedbrainz.org/

[3] http://www.oxfordmusiconline.com

containing over 60k articles written by music scholars. However, it has the drawback of not being freely open, as it runs by subscription. Other than the aforementioned curated repositories, to the best of our knowledge, there is not a single automatically learned open music knowledge base. A first step in this direction is taken in this thesis.

2.2.3. Entity linking

The advent of large knowledge repositories and collaborative resources has contributed to the emergence of entity linking, i.e., the task of discovering mentions of entities in text and link them to a suitable knowledge repository (Moro et al., 2014b). It encompasses similar subtasks such as named entity disambiguation (Bunescu & Pasca, 2006), which is precisely linking mentions of entities to a knowledge base, or wikification (Mihalcea & Csomai, 2007), specifically using Wikipedia as knowledge base. Another highly related technique is word sense disambiguation (Stevenson & Wilks, 2003). Its main task is to identify which sense of a word (i.e., meaning) is used in a sentence, when the word has multiple meanings.

There has been a great development of entity linking systems for unconstrained domains. Among these systems we focus on three of them in this thesis:

DBpedia Spotlight (Mendes et al., 2011) is a system for automatically annotating text documents with DBpedia URIs, finding and disambiguating natural language mentions of DBpedia resources. DBpedia Spotlight is shared as open source and deployed as a Web service freely available for public use[4].

TagMe (Ferragina & Scaiella, 2012) is an entity linking system that matches terms with Wikipedia link texts and disambiguates them using the Wikipedia in-link graph. Then, it performs a pruning process by looking at the entity context. TagMe is available as a web service[5].

Babelfy (Moro et al., 2014a) is an entity linking and word sense disambiguation system based on non-strict identification of candidate meanings (i.e., not necessarily exact string matching), together with a graph-based algorithm that traverses the BabelNet graph and selects the most appropriate semantic interpretation for each candidate[6].

In the context of Open Data[7], the need for benchmarking datasets and evaluation frameworks for entity linking is clear. However, while general-purpose datasets and benchmarks exist (Usbeck et al., 2015), dealing with highly specific domains (e.g., chemistry) or ever-evolving areas (e.g., video games or

[4] `https://github.com/dbpedia-spotlight/dbpedia-spotlight/`

[5] `https://tagme.d4science.org/tagme/`

[6] `http://babelfy.org/`

[7] http://linkeddata.org/

music) poses a greater challenge due to linguistic idiosyncrasies or underrepresentation in general-purpose knowledge bases. This is true in the music domain as well, where available data is scarce (Pereira, 2014). Among the few works on entity linking for the music domain, let us refer to Gruhl et al. (2009), who describe an approach for detecting musical entities from MusicBrainz in informal text. In addition, Zhang et al. (2009) describe a system for musical entity linking in the Chinese language based on Hidden Markov Models.

There is a number of evaluation benchmarks for general-purpose entity linking systems. Cornolti et al. (2013) and Usbeck et al. (2015) put forward benchmarking frameworks for comparing entity linking systems, defining a hierarchy of entity linking problems together with a set of novel measures. Rizzo et al. (2014) and Gangemi (2013) provide evaluation reports on the performance of different state-of-the-art named entity recognition and entity linking systems.

2.2.4. Relation extraction

Extracting semantic relations between entities is an important step to acquire and formalize the knowledge contained in unstructured natural language text (Wang, 2008). Relation Extraction (RE) is an established task in Natural Language Processing (Bach & Badaskar, 2007). It has been defined as the process of identifying and annotating relevant semantic relations between entities in text (Jiang & Zhai, 2007).

Relation Extraction approaches are often classified according to the level of supervision involved. Supervised learning is a core component of a vast number of relation extraction systems, as they offer high precision and recall. However, the need of hand-labeled training sets makes these methods not scalable to the thousands of relations found on the Web (Hoffmann et al., 2011). More promising approaches, called semi-supervised approaches, bootstrapping approaches, or distant supervision approaches do not need big hand-labeled corpus, and often rely on existent knowledge bases to heuristically label a text corpus (Carlson et al., 2010b; Hoffmann et al., 2011).

Open information extraction methods neither require an annotated corpus nor a pre-specified vocabulary, as they aim to discover all possible relations in the text (Banko et al., 2007). However, these unsupervised methods have to deal with uninformative and incoherent extractions. In Fader et al. (2011) part-of-speech regular expressions are introduced to reduce the number of these incoherent extractions. Less restrictive pattern templates based on dependency paths are learned in Mausam et al. (2012) to increase the number of possible extracted relations.

Dependency parsing is an NLP technique that provides a tree-like syntactic structure of a sentence based on the linguistic theory of Dependency Grammar (Tesnière, 1959). One of the outstanding features of Dependency Grammar is that it represents binary relations between words (Ballesteros & Nivre, 2013). Dependency relations have been successfully incorporated to relation extraction systems. For example, Bunescu & Mooney (2005) describe and evaluate a relation extraction system based on shortest paths among named entities. Culotta & Sorensen (2004) focus on the smallest dependency subtree in the sentence that captures the entities involved in a relation, and Gamallo et al. (2012) propose a rule-based dependency-parsing open information extraction system. Moreover, in Nakashole et al. (2012); Moro & Navigli (2012); Bovi et al. (2015b) syntactic and semantic information is exploited to reduce inconsistent relations, by means of the combination of dependency parsing and entity linking techniques.

2.3. Recommender Systems

The main objective of a recommender system is to predict the *rating* or *preference* that a user would give to an item. By doing so, its mission is to provide suggestions for items to be of use to a user (Ricci et al., 2011). *Item* is the general term used to denote what the system recommends to users (e.g., song, album, video, book).

Within the recommender systems arena, there are two main methods for computing recommendations: collaborative filtering and contend-based ones. The most popular is collaborative filtering, which provides recommendations to a user by considering the preferences of other users with similar tastes. The two primary areas in collaborative filtering are neighborhood methods and latent factors models. Neighborhood methods are based on computed similarity between items or between users (Sarwar et al., 2001). By contrast, latent factor models are based on the decomposition of the sparse user-item interactions matrix to a set of user and item d-dimensional vectors using matrix factorization techniques (Koren et al., 2009). The recommendation problem is then treated as a matrix completion problem, where missing entries are filled by taking the dot product of the corresponding user and item latent factors.

Since collaborative filtering methods rely only on user feedback information, they may suffer from the so-called *cold-start* problem (Saveski & Mantrach, 2014). That is, when new items are introduced in the system, they cannot be initially recommended as there is no feedback information related to them. In addition, most popular items tend to attract most of the recommendations in collaborative filtering approaches. The relation between items and popularity is typically represented as a *long-tail* distribution (Anderson, 2006) (see Figure 1.1). A large number of items in the catalog normally receive few interactions from users, so they are hardly recommended. These less popular items may be promoted by priming other evaluation measures rather than prediction accuracy, such as novelty or diversity (Abdollahpouri et al., 2017).

However, another larger set of items in large catalogs receive almost no interactions. These items in the *extreme long tail* are in practice cold-start items, as the number of interactions is smaller than the minimum value required for collaborative filtering methods.

Contrary to collaborative filtering, content-based methods (Mooney & Roy, 1999) rely only on item features, and recommendations are based on similarity between such features. Content-based methods do not suffer from the cold-start problem and are not biased towards popular items. However, collaborative filtering methods tend to perform better when measuring the overall accuracy of the predictions (Pilászy & Tikk, 2009). Finally, hybrid methods (Burke, 2002) try to combine the best of both worlds, combining both item-content and item-user feedback. When available, the usage of side information about items has proven to boost the performances of pure collaborative filtering techniques (Ning & Karypis, 2012).

2.3.1. Knowledge-based approaches

The usage of structured knowledge to improve recommendation systems has been proposed in many works in the past. For instance, in Mobasher et al. (2004) structured semantic knowledge about items is used in conjunction with user-item ratings to create a combined similarity measure for item comparisons. In Ziegler et al. (2004) taxonomic information is used to represents the user's interest in categories of products. In Anand et al. (2007) the authors use an ontology of items to infer user preferences from rating data. In Cantador et al. (2008), user preferences and item features are described by semantic concepts to obtain users' clusters corresponding to implicit *Communities of Interest*. In Middleton et al. (2009) an ontological recommender system makes use of semantic user profiles with the effect of mitigating cold-start and improving overall recommendation accuracy. In all of these works, the experiments prove an accuracy improvement over traditional memory-based collaborative approaches especially in presence of sparse datasets.

With the rise of the Semantic Web, a new class of recommender systems has emerged, taking advantage of the availability of large Linked Open Data (LOD) datasets. One of the first approaches in this sense is Heitmann & Hayes (2010), where LOD is used to mitigate cold-start and sparsity problems. In Fernández-Tobías et al. (2011) DBpedia is leveraged for computing cross-domain recommendations. In Di Noia et al. (2012a,b) a model-based approach and a memory-based one are presented leveraging LOD datasets. In Ostuni et al. (2013) the authors show how to compute top-N recommendations from implicit feedback using linked data sources. In Ostuni et al. (2014) a LOD content-based method is presented, where a neighborhood-based graph kernel is defined for matching graph-based item representations. Finally, another interesting direction about the usage of LOD is explored in Musto et al. (2014), where the authors present a content-based context-aware recommendation framework that adopts a semantic representation based on distributional semantics and entity linking techniques.

2.3.2. Deep learning approaches

Recently several researchers have tried to apply deep learning techniques to recommender systems. Here we list some of these works. Restricted Boltzmann Machines (RBM) was one of the first neural networks methods used in recommender systems, modeling user-item interaction (Salakhutdinov et al., 2007). Neural networks have been also applied to learn latent factors from item content-features, which has been revealed as a useful technique for cold-start recommendations (Wang et al., 2015). This technique has been successfully applied to audio (Van den Oord et al., 2013) and text (Bansal et al., 2016). Neural networks have been also applied to other recommendation problems, such as session-based recommendation (Hidasi et al., 2015) or playlist generation (Vall et al., 2017).

2.4. Music Information Retrieval

As stated in Section 1.2.1, Music Information Retrieval (MIR) is a multidisciplinary field of research that is concerned with the extraction, analysis, and usage of information about music. Traditionally, MIR has been more focused on the use of audio content, underestimating other sources of information. However, in recent years several studies have showed the benefits of using other modalities, as well as their combination in multimodal approaches (Schedl et al., 2014). Although MIR approaches have been traditionally focused on audio content, there has been a growing interest in text-based and multimodal approaches along the past decade. However, most of these text-based approaches are focused on low and mid-level text representations, ignoring the full epistemic potential expressed in texts. In addition, most audio-based approaches have traditionally relied on handcrafted features, underexploring factors of variability behind the data. In this section, we first focus on existing literature related to text and knowledge-based approaches applied to MIR in general. Then, we further explore the related work of three specific MIR tasks: music classification, artist similarity, and music recommendation.

2.4.1. Text-based approaches

Early work on NLP in the context of MIR is related to the extraction of music artist information from artist-related web pages by parsing their HTML-trees (Cohen & Fan, 2000), using weighted term profiles (Ellis et al., 2002;

Whitman & Lawrence, 2002), or counting co-occurrence of artist names in results provided by search engines (Schedl et al., 2005). Other text sources, such as song lyrics (Laurier et al., 2008; Corona & O'Mahony, 2015) and tweets (Hauger et al., 2013; Schedl & Hauger, 2012) have been also studied. More detailed information about text-based approaches applied to MIR problems can be found in Knees & Schedl (2013); Schedl et al. (2014).

In recent years, there have been some initial attempts to work with high-level text representations in the context of MIR. In Sordo et al. (2012), a methodology for extracting semantic information from music-related forums is proposed, inferring semantic relations from the co-occurrence of musical concepts in forum posts. In Knees & Schedl (2011) a methodology to automatically extract semantic information and relations about musical entities from arbitrary textual sources is proposed. In Tata & Di Eugenio (2010) a method to extract information about individual songs from album reviews is proposed, combining syntactic, semantic and sentiment analysis. Finally, the C@merata task (Sutcliffe et al., 2016, 2015), part of the MediaEval evaluation campaigns from 2013 to 2017, is focused on music Question & Answering (Q&A) systems. In this task the input is a natural language phrase, combined with a music score in MusicXML format, and the required output is one or more matching passages in the score.

There have also been some interesting works trying to understand the semantics behind the audio signal using natural language text. The earliest work, by Whitman & Ellis (2004), combines text analysis with acoustic descriptors in order to automatically generate music reviews from the audio signal. In Koduri (2014), culturally relevant and musically meaningful information about melodic intervals extracted from audio and text are structured in a formal knowledge representation and exploited to compute similarity measures for the discovery of musical entities.

2.4.2. Knowledge-based approaches

Knowledge representations have also been studied in the context of MIR, but instead of being extracted from text, they are typically retrieved from online knowledge repositories. For instance, in Sordo et al. (2013) a set of semantic facets is automatically obtained and anchored upon the structure of Wikipedia, and tags from the folkosonomy of Last.fm are then categorized with respect to the obtained facets. In Celma (2006), information from different sources is gathered in a central knowledge repository following the Semantic Web principles and using the Friend of a Friend (FOAF) ontology. This semantic information is then exploited for music recommendation. In Passant & Decker (2010), the DBpedia graph is used to provide explanations of music recommendations, whereas in Ostuni et al. (2013), the same is combined with user feedback data coming from Last.fm to compute music recommendations.

A key aspect in the development of knowledge-based music retrieval systems have been the definition of the Music Ontology (Raimond et al., 2007), a formal framework for dealing with music-related information on the Semantic Web, including editorial, cultural and acoustic information. This development has facilitated the interlinking between music-related datasets on the Web (Raimond et al., 2008). In this direction, Gracy et al. (2013) reviews current efforts to connect music data already available within the Semantic Web. The authors collected, analyzed, and mapped properties used by music Linked Data knowledge bases, library catalogs, and various digital collections.

In this context of Linked Open Data, semantic information has also been exploited for Computational Musicology. It is worth mentioning Crawford et al. (2014), where a method helps the musicologist to create a linked and extensible knowledge structure over a collection of Early Music metadata and facsimile images. In Rose & Tuppen (2014) seven big datasets of musical-biographical metadata are aligned. The authors show how analysis and visualization of the data might transform musicological understanding. In Pattuelli et al. (2013), Linked Data technology is applied to enhance discovery and visibility of jazz music.

2.4.3. Music classification

Most published music genre classification approaches rely on audio sources (for an extensive review on the topic, please refer to Sturm (2012); Bogdanov et al. (2016)). Traditional techniques typically use handcrafted audio features, such as Mel Frequency Cepstral Coefficients (MFCCs) (Logan & Others, 2000), as input of a machine learning classifier (e.g., SVM, k-NN) (Tzanetakis & Cook, 2002; Seyerlehner et al., 2010a). More recent deep learning approaches take advantage of visual representations of the audio signal in form of spectrograms. These visual representations of audio are used as input to Convolutional Neural Networks (CNNs) (Dieleman et al., 2011; Dieleman & Schrauwen, 2014; Pons et al., 2016; Choi et al., 2016a,b), following approaches similar to those used for image classification.

Text-based approaches have also been explored for this task. For instance, one of the earliest attempts on classification of music reviews is described in (Hu et al., 2005), where experiments on multi-class genre classification and star rating prediction are described. Similarly, (Hu & Downie, 2006) extend these experiments with a novel approach for predicting usages of music via agglomerative clustering, and conclude that bigram features are more informative than unigram features. Moreover, part-of-speech (POS) tags along pattern mining techniques are applied in Downie & Hu (2006) to extract descriptive patterns for distinguishing negative from positive reviews. Additional textual evidence is leveraged in Choi et al. (2014), who consider lyrics as well as texts referring to the meaning of the song, and used for training a kNN classifier for predicting song subjects (e.g., war, love, or drugs).

There are a few papers dealing with image-based music genre classification (Libeks & Turnbull, 2011). Regarding multimodal approaches found in the literature, most of them combine audio and song lyrics (Laurier et al., 2008; Neumayer & Rauber, 2007). Other modalities such as audio and video have been explored (Schindler & Rauber, 2015). In McKay & Fujinaga (2008) cultural and audio features are combined for music classification.

Multi-label classification is a widely studied problem in other domains (Tsoumakas & Katakis, 2006; Jain et al., 2016). In the context of MIR, tag classification from audio (or auto-tagging) has been studied from a multi-label perspective using traditional machine learning approaches (Sordo, 2012; Wang et al., 2009; Turnbull et al., 2008b; Bertin-Mahieux et al., 2008; Seyerlehner et al., 2010b), and more recently using deep learning approaches (Choi et al., 2016a; Dieleman & Schrauwen, 2014). However, there are not many approaches for multi-label classification of music genres (Sanden & Zhang, 2011; Wang et al., 2009).

2.4.4. Artist similarity

Artist similarity may be seen as a music recommendation problem without the personalization component. However, we decided to address its literature review separately, given that it has become a proper task in the context of MIR. Music artist similarity has been studied from the score level, the acoustic level, and the cultural level (Ellis et al., 2002). In this thesis, we focus on the latter approach, and more specifically on text-based approaches.

The task of identifying similar text instances, either at sentence or document level, has applications in many areas of Artificial Intelligence and Natural Language Processing (Liu & Wang, 2014). In general, document similarity can be computed according to the following approaches: surface-level representation such as keywords or n-grams (Chim & Deng, 2008); corpus representation using counts (Rorvig, 1999), e.g., word-level correlation, Jaccard or cosine models; Latent factor models, such as Latent Semantic Analysis (Deerwester et al., 1990); or methods exploiting external knowledge bases like ontologies or encyclopedias (Hu et al., 2009).

The use of text-based approaches for artist and music similarity was first applied in (Cohen & Fan, 2000), by computing co-occurrences of artist names in web page texts and building term vector representations. By contrast, in (Schedl et al., 2005) term weights are extracted from search engine's result counts. In Whitman & Lawrence (2002) n-grams, part-of-speech tagging and noun phrases are used to build a term profile for artists, weighted by employing tf-idf. Term profiles are then compared and the sum of common terms weights gives the similarity measure. In Logan & Ellis (2003) Latent Semantic Analysis is used to measure artist similarity from song lyrics.

Multimodal approaches have been also applied to this problem. For instance, in McFee & Lanckriet (2009), social tags, biography summaries, and spectral features are combined and embedded into low dimensional vectors. In Kim et al. (2009), user preference data, social tags, web documents, and audio content are used to compute similarity between artists. Finally, in Fields et al. (2008), audio-based artist similarity is compared to similarity measures based on social connectivity.

2.4.5. Music recommendation

An extensive description of the music recommendation problem and a comprehensive summarization of the initial attempts to tackle it is presented in Celma (2010). An overview about techniques for music recommendation and similarity based on music contextual data is given in Knees & Schedl (2013). In Kaminskas & Ricci (2012) the authors provide a description of various tools and techniques that can be used for addressing the research challenges posed by context-aware music retrieval and recommendation. A survey about techniques for the generation of music playlists is given in Bonnin & Jannach (2014). In particular, the authors provide a review of the literature on automated playlist generation and a categorization of the existing approaches.

Content-based methods have shown to be useful when user feedback information is scarce, as in cold-start scenarios. Social tags have been extensively used as a source of content features to recommend music (Knees & Schedl, 2013). In addition, features extracted from audio signals have also been used as content features. Traditional audio-based approaches rely on handcrafted features obtained from audio signals (Bogdanov et al., 2013a). However, as in many other disciplines and MIR tasks, the application of deep learning approaches has supposed a boost in the performance of audio-based music recommendation (Van den Oord et al., 2013).

Multimodal approaches for content-based music recommendation typically combine audio and textual data, which most commonly consists of web documents, lyrics and social tags (Liem et al., 2011). In Bogdanov & Herrera (2011), for instance, the authors evaluate how much metadata is necessary to improve the quality of audio-based recommendations. In Eck et al. (2008), tags are first learned from audio separately and then combined with the audio in a recommendation system.

Part I

Knowledge Extraction from Text

CHAPTER 3

Linking Music-related Texts to Knowledge Bases

3.1. Introduction

In this chapter we focus on the problem of linking music-related texts, such as artist biographies or music reviews, to knowledge repositories, such as Wikipedia, DBpedia, or MusicBrainz. The language used to describe music and its context is specially ideosyncratic, and Natural Language Processing (NLP) tools and techniques may not be specifically tuned to it. A first step towards the creation of domain-specific NLP tools is the creation of large-scale corpus of annotated documents. However, there is a lack of these music specific datasets for tasks such as named entity recognition or entity linking. Aiming at bridging this gap, we propose ELMD, an automatically constructed corpus where named entities are classified as any of four predefined *musical categories*, namely *Song*, *Album*, *Artist*, and *Record Label.* It was created by leveraging the hyperlinks present in a set of artist biographies gathered from `Last.fm`[8]. Then, we further enrich ELMD by performing entity linking and automatically annotating a large portion of the entities with their DBpedia URI. ELVIS (Entity Linking Voting and Integration System), a voting-based algorithm for entity linking is applied, which considers, for each entity mention in text, the degree of agreement across three state-of-the-art entity linking systems. Manual evaluation shows that entity linking Precision is at least 94% in the resulting dataset. Then, a process to propagate the annotations in ELMD is presented, and annotations are further enriched with MusicBrainz URLs. Finally, a subset of 200 documents is manually annotated with named entities and MusicBrainz URLs to provide a comprehensive gold standard dataset.

In the remainder of this chapter, we first introduce ELVIS, our entity linking integration and agreement approach (Section 3.3). Then, we describe the

[8] `http://www.last.fm`

text corpus we compiled from the `Last.fm` website and how it is combined with ELVIS (Section 3.4). In the next step, the obtained dataset is evaluated (Section 3.5). Then, a further process of automatic expansion of ELMD is described (Section 3.6). Finally, the manual annotation of a subset of ELMD is presented (Section 3.7), and some conclusions are drawn (Section 3.8).

3.2. Music entity linking

Named entity recognition is the task to identify mentions to entities belonging to a set of predefined categories (Zhou & Su, 2002). Traditionally, the most widely covered types of entities are *Person*, *Location* and *Organization*, as well as numeric expressions or time-spans. While named entity recognition is a widely studied topic, and has been at the core of well-known shared tasks and conferences (Nadeau & Sekine, 2007) such as MUC, ACE or CoNLL, the advent of large knowledge repositories and collaborative resources has contributed to the emergence of another discipline: entity linking, i.e., to discover mentions of entities in text and link them to a suitable knowledge repository (Moro et al., 2014b).

In many circumstances, it may be useful to obtain annotations for music entity mentions in text, either simply as music types (e.g., tagging 'Yellow Submarine' as `Song`) or performing entity linking, e.g., tagging 'Yellow Submarine' as `dbpedia.org/page/Yellow_Submarine_(song)`. However, this is not a trivial task as mentions to music entities show language and register idiosyncrasies (Tata & Di Eugenio, 2010; Gruhl et al., 2009), and therefore a certain degree of tailoring is required in order to account for them. Let us consider, for instance, multiword music entities, which usually are those that pose greatest challenges for entity linking. As Tata & Di Eugenio (2010) point out, they are difficult to discover because they may not be restricted to a single Noun Phrase or may be abbreviated (by means of acronyms, dropping entire words or even full rephrasing). Additionally, a specific trait of music texts is the fact that one song may have many covers by many different artists. According to our evaluation, it may be difficult even for a human to identify what *version* of the song the writer is referring to. Furthermore, availability of entity linking testbeds in general (Usbeck et al., 2015), and in the music domain in particular (Gruhl et al., 2009), is scarce, making it very difficult to evaluate novel systems and approaches. Hence, it is difficult to know how well a certain method, which may work well for generic texts, will perform on music data.

Despite the current context of scarcity of both entity linking systems and evaluation benchmarks in the music domain, there are some exceptional cases in which these issues were addressed, such as: (1) Detecting music entities (e.g., songs or bands) on informal text (Gruhl et al., 2009); (2) Applying Hidden Markov Models for discovering music entity mentions in Chinese corpora

(Zhang et al., 2009).

A large number of entity linking systems not bound to any domain or discipline have emerged in the last years. However, we have observed that the number of identification errors in musical entities produced by these systems is still high. We argue that this problem of precision may be tackled by leveraging a combination of several of these generic entity linking off-the-shelf systems. Simply put, we hypothesize that if two or more generic systems annotate with the same URI an entity mention, the probability of this annotation to be correct increases. To the best of our knowledge, very little effort has been put in exploiting this *agreement* feature. One of the reasons may be that, as of now, most entity linking systems *speak their own language*, partially due to the fact that each of them points back to different KBs, and hence their output is heterogeneous and cannot be directly compared, let alone combine. This has motivated research towards unification frameworks for evaluation of entity linking. For instance, Cornolti et al. (2013) put forward a benchmarking framework for comparing entity linking systems. Moreover, Rizzo et al. (2014) describe a system aimed at combining the output of the different named entity recognition systems. Finally Usbeck et al. (2015) present GERBIL, an evaluation framework for semantic entity linking based on Cornolti et al. (2013).

3.3. ELVIS

In this section we describe ELVIS, the generic integration framework for entity linking, which is leveraged for the construction of ELMD. First, we describe our entity linking research problem and provide an intuition on how this may be surmounted via an agreement scheme. Then, we provide details on the main modules integrating ELVIS, highlighting the possible cases of agreement and disagreement over the entity linking systems that are integrated in our framework.

3.3.1. Argumentum ad populum in entity linking

Our method relies on the *argumentum ad populum* intuition, i.e., if two or more different entity linking systems perform the same prediction in linking a named entity mention to its entry in a reference KB, the more likely this prediction is to be correct. We put this intuition into practice by combining the output of three well-known systems, namely DBpedia Spotlight (Mendes et al., 2011), Tagme (Ferragina & Scaiella, 2012), and Babelfy (Moro et al., 2014a), whose agreement (or disagreement) when disambiguating an in-text entity mention is taken as an agreement-driven *confidence score*. These specific tools were chosen for being considered state-of-the-art entity linking systems and for being well known in the NLP community. However, ELVIS can easily incorporate any

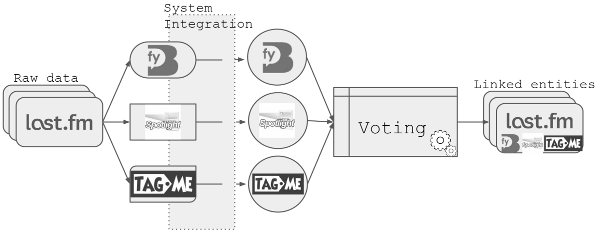


Figure 3.1: ELVIS Workflow.

additional system. We also selected these tools because entities identified by all of them can be easily referenced to DBpedia URIs. While these tools have proven highly competitive on their own, in this chapter we explore the gain in performance obtained by combining them together, and apply global agreement-driven decisions on the `Last.fm` corpus.

3.3.2. 'Translating' entity linking formats

In order to have each entity linking system *speak the same language* for measuring agreement in their predictions, output homogenization is required. This is not a trivial task, as each entity linking approach may be based on a different reference KB, the offsets may be computed differently, and so on. For instance, DBpedia Spotlight links entity mentions via DBpedia URIs, whereas Tagme provides Wikipedia page IDs, and Babelfy disambiguates against BabelNet (Navigli & Ponzetto, 2012) and its corresponding BabelNet synsets. We attempt to surmount this heterogeneity as follows: First, we retrieve DBpedia URIs of every named entity. There are some considerations to be taken into account, however: (1) Character encoding differs from system to system, which we address by converting the character encoding of the retrieved URI to UTF-8; (2) Several URIs may refer to the same DBpedia resource. We solve this specific issue thanks to the transitive redirections provided by DBpedia. If a URI has a transitive redirection, it is replaced by the redirected URI. (3) Note that, in the case of Tagme, only Wikipedia page IDs are provided, which we can straightforwardly exploit to map entity mentions to their DBpedia equivalent. Finally, and after surmounting compatibility issues among systems, we retrieve DBpedia types (`rdf:type` property) and Wikipedia categories (`dcterms:subject` property) for all entities. This *type* information is further used in the creation of ELMD, and throughout this thesis.

After successfully providing a process that harmonizes the output of entity linking systems, it is possible to compute the degree of agreement among them, which will become our system's confidence score. We define the following set of *agreement heuristics* to set such score for each linking prediction (an overview of the workflow of ELVIS is provided in Figure 3.1).

- **Full Agreement** (++) When all systems detect an entity with the same URI and offset.

- **Partial Agreement** (+) When more than one but less than all systems detect an entity with the same URI and offset. Outliers (i.e., systems performing a different prediction) may detect a different entity or may not detect anything.

- **Singleton Decision** (−) When only one system detects an entity for a given text offset.

- **Disagreement** (−−) When more than one system performs a linking over the same text offset, but all of their predictions are different.

3.4. From Last.fm to ELMD

In what follows, we describe the original data gathered from `Last.fm`, and the process to apply the integration framework described in Section 3.3, in order to construct a highly precise benchmarking dataset for entity linking in the music domain.

In `Last.fm`, users may add relevant biographical details to any artist's main page in the form of a *wiki*. These edits are regularly moderated. Furthermore, artist biographies are often enriched with hyperlinks to other `Last.fm` Artist, Album, Song, and Record Label pages, similarly as with Wikipedia hyperlinks. Our purpose is to leverage this meta-information to automatically construct a dataset of Music-specific annotated named entities.

We crawled artist biographies from `Last.fm` in March 2015, and gathered 13,000 artist biographies with hyperlinks, which comprise 47,254 sentences with at least one hyperlink, amounting to a total of 92,930 links. These may be broken down as follows: (1) 64,873 hyperlinks referencing Artist pages; (2) 16,302 to Albums; (3) 8,275 to Song pages; and finally (4) 3,480 hyperlinks referencing Record Labels. This *type* information is extracted thanks to the structure of each link's URL, as it includes in its path the category of the annotated entity. Consider, for example, the following sentence:

> After their debut The Intelligence got signed to *In the Red Records*.

Here, we may infer that the entity *In the Red Records* is a Record Label, thanks to its `Last.fm` URL: `http://www.last.fm/label/In+the+Red+Records`. This information is extracted from the whole `Last.fm` corpus for those entities falling in one of the four *musical categories* previously defined.

Last.fm type	DBpedia type
Song	DBpedia:Song, DBpedia:Single, Yago:Song
Album	DBpedia:Album, Yago:Album, Schema:MusicAlbum
Artist	DBpedia:MusicalArtist, DBpedia:Band, Schema:MusicGroup, Yago:Musician, Yago:Creator, DBpedia:Artist
Record Label	DBpedia:RecordLabel

Table 3.1: Equivalence of types between Last.fm and DBpedia. Yago, Schema, and DBpedia refer to the correspondent ontologies.

3.4.1. Data enrichment

For the creation of the ELMD dataset, the crowd-sourced annotations extracted from `Last.fm` biographies are combined with decisions made by ELVIS and its voting framework.

Every entity mention annotated in the `Last.fm` corpus is a candidate to be included in ELMD. The challenge is to assign to each entity its correct DBpedia URI. We approach this problem by leveraging (1) The DBpedia URI assigned by ELVIS, (2) The *agreement score* for that prediction, as well as (3) The *type* information derived from the entity's `Last.fm` URL. Our intuition is that the higher the *agreement score*, the more likely the prediction is to be correct. Likewise, we also hypothesize that if a linking decision made by ELVIS coincides in *type* with the original `Last.fm` annotation, it is more likely to be correct. Since there is no direct mapping between `Last.fm` and DBpedia types, we manually set the type equivalences shown in Table 3.1.

Regarding the *agreement score*, it corresponds to the number of systems that agreed in a decision (see **Score** column in Table 3.2). Note that an *agreement score* of 1 may be caused either by cases in which only one system detected an entity mention, or when there is disagreement among systems, but one and only one of them coincides in *type* with the original `Last.fm` annotation (last row in Table 3.2).

As for *type equivalence*, this is a binary value (*type-equivalent* or *type-discrepant*) based on coinciding types between `Last.fm` URLs and ELVIS decisions.

Context	**Type**	**Tagme**	**Babelfy**	**Spotlight**	**Score**	**Type Eq.**
and the academic minimalism of **Steve Reich**	Artist	Steve_Reich (type:artist)	Steve_Reich (type:artist)	Steve_Reich (type:artist)	3	yes
The new album **Hypocrisy** followed shortly thereafter	Album	—	Hypocrisy (type:band)	Hypocrisy (type:band)	2	no
The third album **Lucifer Songs**, opened new and unexpected doors	Album	—	Lucifer_Songs (type:album)	—	1	yes
The band's debut album, **Cookies**, was released on 14 May 2007	Album	HTTP_cookie (type:unk)	Cookies (type:album)	—	1	yes (only Babelfy)

Table 3.2: Agreement examples of ELVIS.

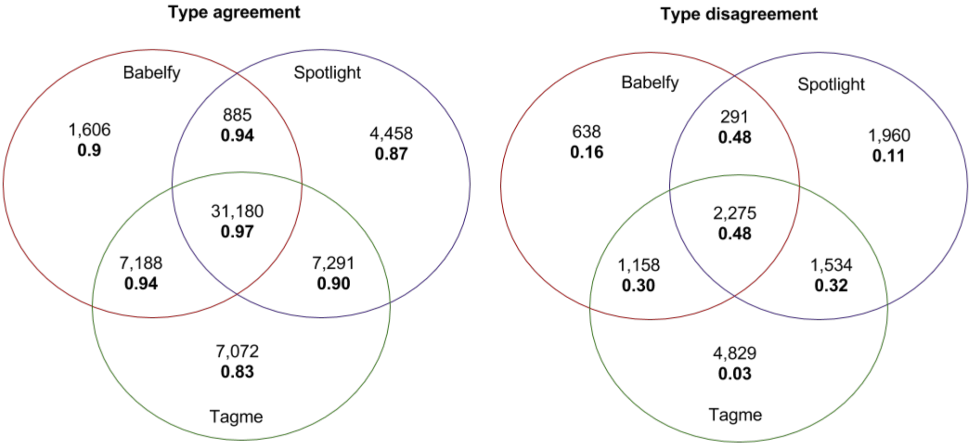


Figure 3.2: Number of entities and precision of the manual evaluation. Note the major differences in Precision between *type-equivalent* and *type-discrepant* systems.

	Agreement	Precision	No. Entities
type-equivalent	$= 3$	0.97	31,180
	≥ 2	0.96	46,544
	≥ 1	0.94	59,680
all	$= 3$	0.94	33,455
	≥ 2	0.90	51,802
	≥ 1	0.81	72.365

Table 3.3: Precision and number of entities with this value of precision of ELMD mapping to DBpedia. *Type-equivalent* implies entities from the type-equivalent configuration only, whilst *All* implies all entities regardless their type information.

Category	Annotations	Entities	Avg-words	Most frequent
Song	3,302	2,823	2.81	Shine (6)
Album	7,872	6,897	2.69	Like Drawing Blood (6)
Artist	46,337	17,535	1.88	The Beatles (160)
Label	2,169	815	1.94	Sub Pop (33)

Table 3.4: Statistics of the linked entities in ELMD. We report, for each *musical category*, the total number of annotations linked to DBpedia, number of unique entities, average number of words per entity mention, and most frequently annotated entity (along with its frequency).

3.5. Evaluation

Considering the different possibilities of agreement across the three systems integrating ELVIS, there are in total 7 possible configurations: 1 with **full agreement** (score= 3); 3 with **partial agreement** (score = 2); and 3 **singleton** configurations (score= 1). Moreover, considering also the two possible values of *type equivalence*, namely `equivalent` and `discrepant`, we have a total number of 14 configurations. Figure 3.2 provides a visual overview of these configurations, where we show both Precision scores for each configuration (in bold) in addition to the number of entities disambiguated with ELVIS in each case.

We evaluated 100 randomly selected entity samples (25 for each of the four music categories we consider) from each one of the 14 possible configurations, and asked an evaluator with computational linguistics background to manually assess the correctness of the 1,400 predictions. From scores obtained from manual evaluation, we estimated Precision for the whole ELMD dataset with different ranges of *agreement score* as well as two options *type*-wise (see Table 3.3). The precision value for all the entities is computed proportionally according to the number of entities and the precision obtained in the manual evaluation for the *type-equivalent* and *type-discrepant* settings, hence these can be seen as Micro Average Precision numbers.

We observe that the *type-equivalent* configuration yields much better Precision with only a slight tradeoff in terms of coverage. Therefore, we decided to select for the final ELMD dataset only those URIs stemming from a *type-equivalent* setting where *agreement score* is equal or greater to 1. This ensures a Precision of at least 0,94 in terms of entity linking. Moreover, a manual survey of false positives in the highest scoring setting (*agreement score*= 3 and *type-equivalent*) showed that these are cases in which even a human annotator may not find it trivial to find the correct entity to those entity mentions. One of these cases are those in which ELVIS is presented with and entity mention that on surface may refer to either an Artist or an Album named after the artist or band itself. An actual case of false positive in our evaluation dataset is the following sentence:

> Her debut album, *Kim Wilde*, (released on RAK records) came out in July 1981 and stayed in the U.K. album charts for 14 weeks, peaking at number 3 and getting much acclaim.

Here, the entity *Kim Wilde* should be disambiguated as the Album with the same name as the artist, but ELVIS incorrectly assigned the Artist's DBpedia URI: `dbpedia.org/resource/Kim_Wilde`. In ELMD there are 50 cases where the same surface text is correctly linked to an Artist entity in some sentences, and to a Song entity in others. Similar ambiguous cases involving Artist and Album (148) and Song and Album (95) are correctly resolved by our system. These particularly challenging cases may be interesting for training music-specific entity linking algorithms.

Another interesting source of false positives comes between musical entities and equally named entities (not necessarily related to Music). In cases in which the latter are more popular in a reference KB, e.g., their associated node in the graph may have higher connectivity, may become prioritized by disambiguation entity linking algorithms that consider graph connectivity as a feature. Consider the following sentence:

> He is becoming more and more in demand for his remixing skills; working for the likes of Justin Timberlake and Armand van Helden, and labels including *Ministry Of Sound*, Defected and Intec, to name a just a few.

Here, the entity *Ministry of Sound* refers to a Record Label, a spin-off of the well-known club, which is the entity that was incorrectly assigned: `dbpedia.org/resource/Ministry_of_Sound`. Cases like this would require, first, to ensure that the different entities derived from *Ministry of Sound* (such as the Record Label or a clothing brand of the same name) exist in a reference KB, and second, to exploit contextual information so that a correct decision is made.

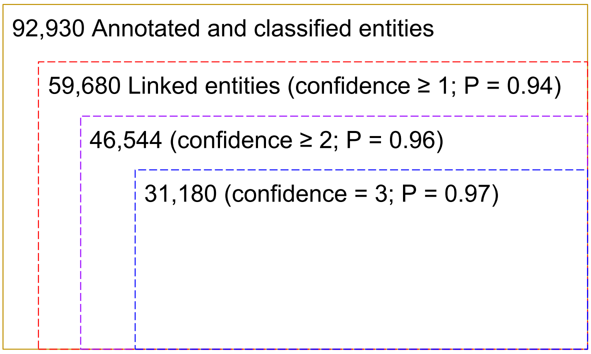

92,930 Annotated and classified entities
59,680 Linked entities (confidence ≥ 1; P = 0.94)
46,544 (confidence ≥ 2; P = 0.96)
31,180 (confidence = 3; P = 0.97)

Figure 3.3: ELMD Overview. Number of annotations, confidence score, and precision values at different confidence score thresholds.

A similar situation happens when song or album names may be confused with very common words or expressions (e.g., 'Easy', 'Stupid', 'Sad song', 'If', 'Be there'). ELMD is rich in challenging cases like these.

As shown in Figure 3.3, different subsets of ELMD can be obtained at different confidence score thresholds, with Precision ranging from 0.94 to 0.97.

3.6. Extending ELMD

The number of links present in the `Last.fm` biographies is small compared to the size of the biographies. For instance, a link may have been added only once in a specific biography, even though the same entity is mentioned several times along the text. In addition, music information represented in DBpedia is not complete, as many existing artists, albums, and songs do not have a Wikipedia page. As a consequence, there are many annotated links in the biographies to `Last.fm` pages that does not have a corresponding DBpedia resource. Therefore, to extend the coverage and the number of annotations of the ELMD dataset we applied the following processes. First, we take advantage of the fact that a large portion of `Last.fm` annotations have a direct mapping to MusicBrainz, and this information can be retrieved through the `Last.fm` API. Thus, in addition to the already available DBpedia links, MusicBrainz URLs are added to the annotations, when this information is available. Furthermore, existing annotations in every document are propagated, assuming they appear in a one-sense-per-discourse fashion (Gale et al., 1992). For example, if the text span *The Beatles* is marked as an annotation in the first sentence of a document, and it appears again in the second sentence, but there is no annotation associated, an annotation is added. Finally, we look for mentions of the entity that constitutes the main theme of the biography, and annotate all its mentions within the biography, assuming unambiguity. The number of annotations and

	Annotations	Entities
All	144,593	63,902
Artist	112,524	39,131
Album	18,701	15,064
Song	9,203	7,832
Label	4,165	1,875

Table 3.5: Statistics of the extended ELMD corpus. `Annotations` refers to all distinct mentions or apparitions of an entity of its corresponding type, whereas the `Entities` column refers to the number of distinct entities of each type.

	Annotations	Entities
DBpedia	58.6%	49.1%
MusicBrainz	93.6%	91.1%
Both	57.2%	47%
None	5%	9.2%

Table 3.6: Percentage of linked entities in the extended ELMD corpus.

distinct entities after the extension process are reported in Table 3.5. Note that MusicBrainz has a coverage of 93.6% over all the annotations, and 91.1% over all distinct entities (see Table 3.6).

3.7. Gold standard dataset

We envision a wide range of potential applications for ELMD, such as acting as a training set for a named entity recognition or entity linking system, or as an evaluation benchmark. However, it suffers from two major problems that differentiate it from a gold standard dataset which undergoes a full manual validation pass. First, although there are an important number of annotated entities, there are still many musical entities mentioned in ELMD texts that are not linked to any KB, nor even annotated. Second, as the dataset has been automatically generated, it is prone to errors, as we show in the evaluation in Section 3.5. To tackle these issues, a human expert manually annotated a randomly selected subset of 200 documents from ELMD. We asked the annotator to mark in each document all mentions of entities of the following types: *Artist*, *Album*, and *Song*. Record Label entities were discarded due to the low number of annotations present in the documents. In addition, the annotator manually searched for each entity in the MusicBrainz database. If it was present, the MusicBrainz URL was added to the annotation. The final number of annotations is shown in Table 3.7. This gold standard dataset has been used in the Task 3 of the third edition of the Open Knowledge Extraction Challenge, co-located with the Extended Semantic Web Conference (ESWC

	Annotations	**Entities**
All	5,184	2,803
Artist	3,828	1,926
Album	860	693
Song	496	184

Table 3.7: Statistics of the ELMD gold standard corpus. `Annotations` refers to all distinct mentions or apparitions of an entity of its corresponding type, whereas the `Entities` column refers to the number of distinct entities of each type.

2017) (Speck et al., 2017).

3.8. Conclusion

In this chapter we have described several contributions related to the problem of recognizing and linking musical entities in naturally occurring text. First, for the task of entity linking, we have presented an integration framework called ELVIS which, based on a voting procedure which leverages decisions made by an arbitrary number of off-the-shelf entity linking systems, provides high confident entity disambiguations. Currently, ELVIS incorporates three state-of-the-art systems, namely DBpedia Spotlight, Tagme and Babelfy, and can be easily extended with additional systems. Then, we have leveraged the potential of ELVIS for the creation of a novel benchmarking dataset for entity linking in the music domain, called ELMD. This corpus comes from a collection of `Last.fm` artist biographies, and contains 47,254 sentences with 92,930 annotated and classified entity mentions. From this set of annotations, 59,680 are linked to DBpedia (see Table 3.4), with a precision of at least 0.94 (see Figure 3.3). Moreover, we have extended the number of annotated entities in ELMD via several heuristics. Furthermore, in addition to the DBpedia linking, we successfully linked 93% of the annotations to MusicBrainz. Finally, we have manually annotated and linked to MusicBrainz a gold standard subset of 200 documents from ELMD, for its use within an entity linking challenge.

CHAPTER 4

Automated Construction of Music Knowledge Bases

4.1. Introduction

In this chapter, we present and evaluate an Information Extraction pipeline aimed at the construction of a Music Knowledge Base (MKB) entirely from scratch in an automated and unsupervised manner. We combine a state-of-the-art entity linking tool and a linguistically motivated rule-based algorithm to extract semantic relations between entity pairs. Our method is able to generate a fully disambiguated MKB with entity mappings against DBpedia and MusicBrainz. All relations have a relation pattern derived from a relation extraction procedure backed up by an algorithm that performs the following steps: (1) Morpho-syntactic rule-based *filtering*; (2) Syntactic dependency-based *clustering*; and (3) Relation *weighting* based on statistical evidence.

We validated our methodology on a large collection of documents in the music domain, obtained from *songfacts.com*, a website that collects "tidbits" (short stories) about songs. We carried out an intrinsic evaluation on each component of the algorithm, as well as an extrinsic evaluation which consists of a experiment on interpretation of music recommendations, where our automatically extracted MKB is used to provide explanations to song recommendations in *natural language*. Our experimental results indicate that our system is able to extract *high quality* relations (Precision >= 0.8) as well as *novel knowledge*. We unveil thousands of relations absent in both large-scale generic KBs, as well as in music specific resources. Moreover, the recommendation explanation experiment shows that explanations based on the newly extracted KB have a positive impact in user experience.

The rest of this chapter is organized as follows. In Section 4.2 we describe step by step the proposed methodology for relation extraction. Then, in Section 4.3 we illustrate the gathered dataset and the outcome of the relation extraction process. The results of our evaluation are reported in Section 4.4, and the chapter ends with a discussion about our findings.

4.2. Method

We propose a comprehensive pipeline that extracts a full-fledged MKB taking as input raw text collections. The experiments we report in this chapter are the result of applying our method to a dataset of plain text extracted from the Songfacts[9] website (see Section 4.3.1). This is a well suited resource both for KB construction and as a testebed for relation extraction due to its specificity. Essentially, Songfacts documents, while not being as rigid as encyclopedic text or newswire text, remain well-formed, sentences make sense, and there is no need for *ad-hoc* preprocessing (as it is required in social networks, e.g., Twitter). Our method, however, can be ported with little effort to music-related corpora of different registers.

4.2.1. Notation

Our method focuses on the extraction of semantic relations between pairs of linked entities (e.g., *Born in the USA*$_{dbr}$, *Bruce Springsteen*$_{dbr}$[10]), which are in turn associated to specific entity types (e.g., *Album*, *MusicalArtist*). In our KB, a relation r is defined by the tuple $\langle \boldsymbol{e_d}, \boldsymbol{e_r}, \boldsymbol{v_d}, \boldsymbol{v_r}, \boldsymbol{p}, \boldsymbol{c} \rangle$, where $\boldsymbol{d}$ and $\boldsymbol{r}$ refer to domain and range positions, $\boldsymbol{e_d}$ and $\boldsymbol{e_r}$ to the entities involved in the relation, $\boldsymbol{v_d}$ and $\boldsymbol{v_r}$ to their associated entity types, $\boldsymbol{p}$ to a relation pattern, and $\boldsymbol{c}$ to a cluster pattern. A relation pattern is a relation label that may be used in one or several relations (e.g., *was recorded by frontman*, *was recorded by singer/songwriter*). Relation patterns with similar semantic and syntactic characteristics may be grouped into cluster patterns (e.g., *was recorded by*). $\mathcal{R}$ denotes the set of all extracted relations included in the KB. For each $r \in \mathcal{R}$, triples of different nature can be constructed by arbitrarily combining elements in r (t for relations between entities and τ for relations between entity types).

- $t_{\boldsymbol{p}} : \langle \boldsymbol{e_d}, \boldsymbol{p}, \boldsymbol{e_r} \rangle$, e.g., {*Born in the USA*$_{dbr}$ - *was recorded by frontman* - *Bruce Springsteen*$_{dbr}$}.
- $t_{\boldsymbol{c}} : \langle \boldsymbol{e_d}, \boldsymbol{c}, \boldsymbol{e_r} \rangle$, e.g., {*Born in the USA*$_{dbr}$ - *was recorded by* - *Bruce Springsteen*$_{dbr}$}.
- $\tau_{\boldsymbol{p}} : \langle \boldsymbol{v_d}, \boldsymbol{p}, \boldsymbol{v_r} \rangle$, e.g., {*Album* - *was recorded by frontman* - *MusicalArtist*}.

[9] http://www.songfacts.com

[10] We use the *dbr* subscript to refer to disambiguated entities linked to DBpedia resources.

- $\tau_{\boldsymbol{c}} : \langle \boldsymbol{v_d}, \boldsymbol{c}, \boldsymbol{v_r} \rangle$, e.g., {*Album - was recorded by - MusicalArtist*}.

Finally, different subsets of $\mathcal{R}$ may be constructed by selectively filtering all $r \in \mathcal{R}$.

- $\mathcal{R}_p = \{r_1^p, ...r_n^p\}$ All relations with a specific relation pattern p.
- $\mathcal{R}_c = \{r_1^c, ...r_n^c\}$ All relations with a specific cluster pattern c.
- $\mathcal{R}_{\boldsymbol{\tau_p}} = \{r_1^{\tau_p}, ...r_n^{\tau_p}\}$ All relations with a specific relation pattern, and domain and range entity types.
- $\mathcal{R}_{\boldsymbol{\tau_c}} = \{r_1^{\tau_c}, ...r_n^{\tau_c}\}$ All relations with a specific cluster pattern, and domain and range entity types.

In what follows, we describe a method for acquiring new entities, types and relations, and combining them in a meaningful way for KB construction.

4.2.2. Morphosyntactic preprocessing

Our morphosyntactic preprocessing module takes as input a collection of text documents in the music domain. First, sentence splitting and tokenization is carried out thanks to the *Stanford NLP tokenizer*[11]. Next, a dependency parse tree is obtained via the MATE Parser, described in Bohnet (2010). We justify the use of the latter because of the richness of its tagset, as well as performance in terms of accuracy and speed, which were appropriate for the task at hand.

In a dependency tree, each node includes information, at least and depending of the model and the language, about surface and lemmatized forms, along with its part-of-speech. Each edge in the tree is labeled with a dependency relation such as *subject* or *noun modifier* (an example is shown in Figure 4.1).

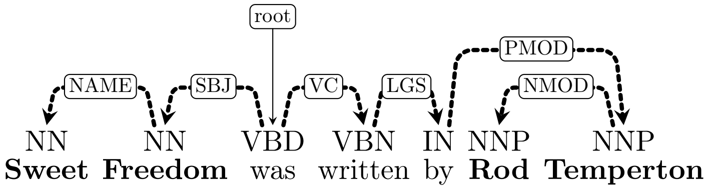


Figure 4.1: Example sentence with dependency parsing tree.

[11] http://nlp.stanford.edu/software/tokenizer.shtml

4.2.3. Semantic processing: entity linking

Entity linking acts as a semantic bridge between plain text and a reference knowledge inventory. As explained in Chapter 3, there is no benchmark of entity linking systems in the music domain, so we do not know *a priori* how well the different systems behave in music corpora. Musical entities may raise a plethora of challenges, derived mostly from ambiguity and polysemy. For example, an album may have the same name as the band who recorded it (e.g., *Weezer* the band and their first album). Moreover, an artist, a song or an album may have words or expressions much more common in another domain or area of knowledge (e.g., *Berlin, The Who*). Thus, the choice of the best entity linking algorithm or off-the-shelf tool(s) is crucial, as potential errors may propagate throughout the different modules and hinder considerably the quality of the resulting KB.

Among the available entity linking systems we considered, namely TagMe (Ferragina & Scaiella, 2010), Babelfy (Moro et al., 2014b), and DBpedia Spotlight (Mendes et al., 2011), we opted for the latter, as it has shown to be the least prone to errors in our corpus (further details are provided in Section 4.4.1).

Adding co-references

In the music domain, prototypical factoid documents such as artist biographies, album reviews, or song tidbits, normally refer to one specific entity. Based on this observation, we may exploit co-referential pronouns and *resource-specific co-references*, replacing them by the name of the reported entity. A similar approach is used in Voskarides & Meij (2015), where the frequency of pronouns "he" and "she" is computed in every document (Wikipedia articles in this specific case) to determine the entity's gender, and then, these pronouns are replaced by the entity title.

We have observed an exploitable *resource-specific co-reference* in music reviews, where terms like "this album" or "the song" can be replaced by the document's title. In the dataset used for the experiments (see Section 4.3.1), the expressions "this song" and "the song" are replaced with the name of the song as it appears in the document, and disambiguated with the URI of the entity they unequivocally refer to.

Co-reference resolution is a difficult and crucial task in NLP, affecting tasks such as Information Extraction (Soon et al., 2001) or document summarization (Saggion & Gaizauskas, 2004). It is also sensitive to the domain in which it appears (see, for instance, the case of the patents domain (Bouayad-Agha et al., 2014)). We acknowledge the difficulty of this task. However, while addressing this problem in its entirety is out of the scope of this chapter, the described strategy allows us to increase coverage of entity mentions while maintaining a high precision.

Our MKB	**DBpedia ontology**	**MusicBrainz**
MusicalArtist	Person/Artist/MusicalArtist Organization/Band Writer/MusicComposer Writer/SongWriter	Artist
OtherArtist	Person/Artist (¬ MusicalArtist) Person/Writer(¬ MusicComposer & ¬ SongWriter)	—
Album	Work/MusicalWork/Album	Release
Song	Work/MusicalWork/Song Work/MusicalWork/Single	Recording Work
Genre	TopicalConcept/Genre	—
Film	Work/Film	—
RecordLabel	Agent/Organization/Company/RecordLabel	Label

Table 4.1: Type mapping.

Type filtering

In DBpedia, most resources are associated with one or more types via the `rdf:type` property. In addition, among the different types present in DBpedia (coming from the DBpedia ontology, Yago types, or `schema.org`), the DBpedia ontology provides a relatively small and tidy taxonomy of 685 classes based on Wikipedia infoboxes. Other KBs such as Yago or Freebase have their own ontological structure, which is in general broader and noisier. MusicBrainz, in contrast, has a very narrow set of entity types.

This type information can be exploited in order to narrow down the set of allowed types for a given candidate and its potential annotations. In this way, we ensure that all entities will be, at least, related to the music domain. Restricting the search space to types such as Artist or Song reduces considerably the number of errors derived from cross-domain ambiguity. For instance, the entity linking system detects a substantial amount of entities whose DBpedia type is *FictionalCharacter*, which are in most of the cases misleading song titles or band names with fictional characters of the same name. This situation is observed also with other types of entities such as *Athlete*, *Species*, or *Disease*.

Depending on the envisioned application of the KB resulting from our pipeline, the predefined set of entity types may vary. In our case we restricted them to Musical Artists, Other Artists, Songs, Albums, Genres, Films and Record Labels. In Table 4.1 we present the mapping between the DBpedia ontology, MusicBrainz entity types, and our selected set of types.

4.2.4. Syntactic semantic integration

The information obtained from the syntactic and semantic processes is combined into a graph representation of the sentence. For each music entity identified during the semantic processing step (Section 4.2.3), all nodes in the dependency tree with a correspondence with an entity mention are collapsed into one single node: *Sweet* and *Freedom* into *Sweet Freedom (Album)*, and *Rod* and *Temperton* into *Rod Temperton (Artist)*. Figure 4.2 shows the resulting syntactic-semantic representation of a sentence.

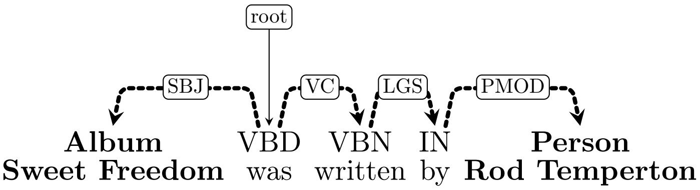


Figure 4.2: Semantic integration on syntactic dependencies.

4.2.5. Relation extraction and filtering

Our approach to relation extraction is lightweight, unsupervised, and rule-based. Having syntactic and semantic information available, potential relations between entities may be discovered by traversing the dependency tree. Two entities in such tree are considered to be related if there is a path between them that does not contain any other entity in between, and does not contain parentheses. If there is more than one path, we consider only the shortest path as the most representative path of the relation.

Our method encodes a relation pattern between two entities as all words in the shortest path between them. In the example provided in Figure 4.2, the shortest path between *Sweet Freedom* and *Rod Temperton* contains the words *was*, *written*, and *by*.

While relation extraction via shortest path in syntactic trees is common practice in the literature (Bovi et al., 2015b; Moro & Navigli, 2012; Nakashole et al., 2012), not all shortest paths are valid, and incorrect relations may be extracted from overly long and syntactically complex sentences. We aim at surmounting these problems by defining three filtering heuristics over surface forms (*lemma-paths*), part-of-speach patterns (*pos-paths*), and labels of syntactic dependencies (*dependency-paths*).

First, we filter out all relations with reporting verbs (e.g., “say”, “tell” or “express”) in the lemma-path (see the full list of banned lemmas in Table 4.2). The intuition being that sentences with these verbs are by definition syntactically complex, and semantic relations in them may not be encoded via shortest paths. We illustrate this with the following sample sentence, where the relation extracted with syntactic tree traversal by means of shortest path would be incorrect:

> **Sentence:** `Nile Rodgers` *told* NME that the first album he bought was Impressions by `John Coltrane`.
>
> **Relation**: `nile_rodgers` told that was impressions by `john_coltrane`

Filtering Heuristics	Description	Patterns
lemma-paths	banned lemmas	say, tell, speak, explain, express, mention, inform, thank, ask, admit
dependency-paths	allowed start patterns	PRD, VC, SBJ, NMOD SBJ, OBJ, APPO SBJ
pos-paths	special allowed patterns	NN, NN NN, NNS, NN CC NN, NN IN, IN NN

Table 4.2: Complete set of patterns used in the filtering heuristics

Second, we only selected relations where the syntactic function that connects in the dependency-path the fist entity with the first word of the relation pattern is a subject (which may be preceded by a nominal modifier or an apposition), a direct or indirect object, a predicative complement or a verb chain (see the full list of allowed patterns in Table 4.2). When this condition holds, the relation is considered *valid*. If the above condition does not hold, an extra validation step is applied over the pos-path in order to capture relations without verbs, which seem to be idiosyncratic of the music domain, e.g., $\langle e_d, \texttt{frontman of}, e_r\rangle$, $\langle e_d, \texttt{drummer}, e_r\rangle$, or $\langle e_d, \texttt{guitarist and singer}, e_r\rangle$ (see the full list of allowed pos-path patterns in Table 4.2).

4.2.6. Dependency-based loose clustering

In this section we describe a simple but powerful clustering algorithm aimed at reducing the number of relation patterns in the KB.

Let us consider the following three relation patterns: (1) *was written by blunt producer*, (2) *was written by singer/producer*, and (3) *was written by manager and guitarist*. Intuitively, these three relation patterns seem to be semantically similar, and if all of them were expressed as *was written by*, the original meaning would not be lost, and the set of relations would become more compact.

This observation, which we found to occur quite frequently, motivated the inclusion of a *dependency-based loose clustering* module. First, we perform a second run of dependency parsing over all relation patterns extracted by our system, aiming at discovering their root node. We apply this second run because the root of the original sentence does not need to correspond with the relation pattern's root. Then, our algorithm considers all possible paths from the root to every leaf node of the relation pattern dependency tree, and selects the path that complies with a predefined syntactic constraint (e.g., a sequence of verbs plus adverb or preposition, or adverb plus nominal and preposition modifiers) based on regular expressions of syntactic labels. The sequence of tokens that matches this regular expression constitutes the cluster pattern. The complete set of defined regular expressions is reported in Table 4.3.

As an illustrative case, consider the extracted relation pattern *is track was released on label* from the sentence ***Sing Out The Song*** *is the 7th track on*

<table>
<tr><th>Regular Expressions</th></tr>
<tr><td>^(VC)+\s+(DEP|LGS|LOC|TMP)</td></tr>
<tr><td>^(VC)+\s+(ADV)\s+(PMOD|NMOD|AMOD)*</td></tr>
<tr><td>^(DEP|LGS|LOC|TMP)</td></tr>
<tr><td>^(APPO)\s+(LGS)</td></tr>
<tr><td>^(SBJ)\s+(PMOD|NMOD|AMOD|ADV)</td></tr>
<tr><td>^(OBJ)\s+(PMOD|NMOD|DEP)</td></tr>
<tr><td>^(ADV)</td></tr>
<tr><td>^(NMOD|PRT|PMOD)$</td></tr>
</table>

Table 4.3: Regular expressions for dependency paths in cluster patterns.

Wishbone Four which was released in the UK May 1973 on the ***MCA*** *label.* After re-parsing the relation pattern, we obtain the parse tree shown in Figure 4.3 and a cluster pattern over those nodes in the dependency tree that satisfy one of the regular expressions crafted in the aforementioned syntactic constraint. Finally, the obtained relation is ***Sing_out_the_song*** *was released on label* ***MCA***.

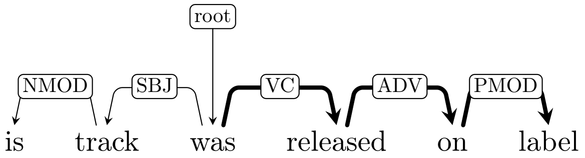


Figure 4.3: Example of a parsed relation pattern $p \in \mathcal{P}$ and a valid cluster pattern (bold).

Filtering out spurious information in OIE following similar approaches has proven effective while not being computationally expensive (Fader et al., 2011).

Ours is a *loose clustering* method because it does not enforce a pattern to fully match all rules, but rather allows partial matching. This module provides an enrichment of all $r \in \mathcal{R}$ such that $r = \langle e_d, e_r, v_d, v_r, p, c \rangle$, where c is the cluster pattern derived from the relation pattern p. A relation cluster is the set of all relations with the same cluster pattern, and is denoted as $\mathcal{R}_c$.

4.2.7. Scoring

So far, our approach has identified entity mentions in text and has linked them in meaningful relations, filtering out those that did not comply with predefined linguistic rules. We incorporate one additional factor $score(r)$ that takes into account statistical evidence computed over $\mathcal{R}$. It has three main components, which we flesh out as follows.

We hypothesize that the relevance of a cluster may be inferred by the number and proportion of triples it encodes, and whether these are evenly distributed. Our metric encompasses a combination of three different components. First, we focus on the *degree of specificity* of the relation cluster, as previous work has demonstrated that this can contribute to Information Extraction pipelines (Bovi et al., 2015a). Second, we analyze *intrinsic features* of the re-

Cluster pattern c	**Typed cluster pattern** τ_c	**Relation triples** t_p
was written by	S *was written by* MA	s1 *was writen by artist* ma1
		s2 *was written by composer* ma2
		s3 *was written by singer* ma2
		s4 *was writen by* ma1
		s5 *was written by frontman* ma3
	A *was written by* MA	a1 *was written by frontman* ma3
		a2 *was written by guitarist* ma1
		a3 *was written by artist* ma2
		a4 *was written by frontman* ma5

Table 4.4: Example of a relation cluster $\mathcal{R}_c$, where $c = was\ written\ by$. S refers to Song, MA to MusicalArtist and A to Album types, whilst sX refers to Song, maX to MusicalArtist and aX to Album entities.

lation pattern, such as frequency, length and fluency. Finally, we incorporate a *smoothing factor*, namely the proportion of the related typed cluster pattern in the cluster.

A cluster $\mathcal{R}_c$ may be decomposed into a set of typed cluster patterns τ_c (see Table 4.4). The intuition behind the specificity measure of a cluster is that clusters with one prominent τ_c are more specific, i.e., they are largely used for encoding one specific type of relations. One example of this would be *performed with*, which enforces a relation to include MusicalArtists on both the domain and range sides. Thus, we define $\mathcal{L}_c$ as the list of cardinalities (number of triples) of every typed cluster pattern $\tau_c \in \mathcal{R}_c$, being $\mathcal{L}_c = \{|\mathcal{R}_{\tau_c^1}|, ..., |\mathcal{R}_{\tau_c^n}|\}$. We define the specificity measure as the variance of $\mathcal{L}$, expressed as:

$$s(\mathcal{R}_c) = var(\mathcal{L}_c) \tag{4.1}$$

Furthermore, we consider a *relation's fluency* metric, which is aimed at capturing its comprehensibility. Simply put, the more the sentence's original word order is preserved in the relation pattern, the more understandable it should be. This metric is introduced due to the fact that word order is lost after modeling text under a dependency grammar framework, and so we design a *penalty measure* over the number of jumps needed to reconstruct the original ordered word sequence. Let k be the number of tokens in the relation pattern, w_i the ith word in the pattern, and $h(w_i)$ a function that returns the correspondent word index in the original sentence, we put forward a fluency measure f defined as:

$$f(p) = \frac{\sum_{i=1}^{k} \alpha_i |h(w_i) - h(w_{i-1})|}{k} \tag{4.2}$$

where $\alpha_i = 2$ if $h(w_{i-1}) > h(w_i)$ and $\alpha = 1$ otherwise. Note that higher values of f means low fluency. For instance, for the relation pattern *is hit for* the score would be much higher than a mixed-up order relation pattern such as *joined because added were and hit*, which would have a very high f.

Finally, the global confidence measure for each relation $r \in R$ is expressed as follows:

$$score(r) = \left(s(\mathcal{R}_c) + \frac{|\mathcal{R}_p|}{|p| + 2^{f(p)}} \right) \times \frac{|\mathcal{R}_{\mathcal{T}_c}|}{|\mathcal{R}_c|} \tag{4.3}$$

As an illustrative example of the measure, the score of a relation with the typed cluster pattern ⟨Song, *was released on*, RecordLabel⟩, will have a much higher score than a relation whose typed cluster pattern is ⟨Album, *was released on*, MusicalArtist⟩. This latter pattern is incorrect, probably due to a disambiguation error in the entity linking step. Relations like this show the type of errors which our proposed confidence score is expected to consider for pruning.

4.3. Experimental setup

In this section, we describe our experimental setting. We refer first to the source raw corpus, and second to the resulting KBs as output of different branches of our approach.

4.3.1. Source dataset

Songfacts is an online database that collects, stores, and provides facts, stories, and trivia about songs. These are collaboratively written by registered users, and reviewed by the website staff. It contains information about more than 30,000 songs from nearly 6,000 artists. This information may refer to what the song is about, who wrote it, who produced it, who collaborated with whom, or who directed the video. These texts are rich sources of information not only for well-known music facts, but also for music-specific trivia, as in the following sample sentence (about David Bowie's *Space Oddity*): "Bowie wrote this song after seeing the 1968 Stanley Kubrick movie 2001: A Space Odysse".

We crawled the Songfacts website in mid-January 2014. Then, for each song article, we performed a mapping between the song and its MusicBrainz recording ID, using the MusicBrainz Search API. We successfully mapped 27,655 songs.

The described methdology was run over the 27,655 documents in the Songfacts corpus, which amounts to 306,398 sentences. After the Semantic Processing step, we obtained 202,767 linked entities (8,880 for *Albums*, 3,136 *Record Labels*, 74,908 *Songs*, 107,253 *Musical Artists*, 1,760 *Genre* labels, 3,467 for *Other*

Artist, and 3,363 for *Film*). There were 48,122 sentences with at least two entities, and it is on this subset where we apply our relation extraction pipeline.

4.3.2. Extracted knowledge bases

Our aim is to assess to what extent each of the modules integrating our approach contributes to the quality of the resulting KB. After executing the whole pipeline, we generate two *extracted* KBs (KBSF-ft and KBSF-th), two *baseline* KBs (KBSF-co and KBSF-raw), and a *competitor* KB (KBSF-rv).

The *extracted* KBs are the result of applying the relation extraction method to the Songfacts dataset under different conditions. KBSF-ft is derived from applying the relation extraction pipeline entirely, and KBSF-th comes from a selection of all triples in KBSF-ft with a confidence score above a certain threshold. To determine the best threshold to prune KBSF-ft, we aimed at maximizing the number of triples and at the same time minimizing the number of relation patterns. Our intuition is that fewer patterns means a tidier KB. Therefore, we computed the percentage of triples and relation patterns from KBSF-ft that remain in a pruned KB, whose triples have a score greater than a certain threshold θ. We computed these percentages for every θ value ranging from 0 to 1 in bins of 0.01 (see Figure 4.4). Our goal was to discover the θ value which maximizes the distance between the amount of triples and the amount of relation patterns in a pruned KB. After confirming a maximized difference with $\theta = 0.05$, we created KBSF-th, whose triples have a score greater than or equal to 0.05. In this pruned KB, we have 36.56% of KBSF-ft triples, with only 12.52% of its relation patterns.

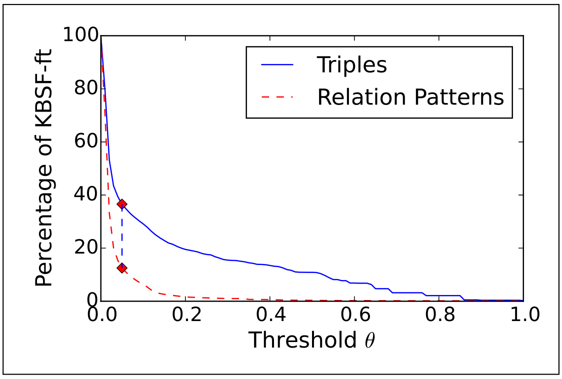


Figure 4.4: Percentage of triples and relation patterns from KBSF-ft that remain after pruning at different values of θ. Maximum distance at $\theta = 0.05$.

In addition, we created two baseline KBs for evaluation purposes. KBSF-co is a baseline which consists of simple entity co-occurrence. More specifically, if two entities are mentioned in the same sentence, an unlabeled triple that anchors them is added to the KB. In addition, KBSF-raw was created following the relation extraction pipeline, but without applying the filtering process described in Section 4.2.5. Finally, KBSF-rv constitutes the competitor KB, and is built as follows: After running ReVerb (Fader et al., 2011), a state-of-the-art relation extraction system, over the Songfacts dataset, we search coinciding relations, at both domain and range positions, that include entity mentions identified in our disambiguation step. These relations are included in KBSF-rv. Statistics about the five KBs are reported in Table 4.5.

KB	Entities	Triples	Relation Patterns	Cluster Patterns
KBSF-ft	20,744	32,055	20,438	14,481
KBSF-th	10,977	11,720	2,484	828
KBSF-co	30,671	113,561	—	—
KBSF-raw	29,280	71,517	47,089	32,712
KBSF-rv	9,255	7,532	2,830	—

Table 4.5: Statistics of all the extracted KBs

4.4. Experiments

4.4.1. Quality of entity linking

In this section, we performed a set of experiments to select the best-suited entity linking tool for our task, among some of the best known and reputed. Specifically, we performed evaluation experiments on DBpedia Spotlight, TagMe, and Babelfy.

As stated in Chapter 3, most entity linking systems *speak their own language*. Since their output is heterogeneous in format, performing a comparison between them is not straightforward. In order to evaluate the aforementioned entity linking systems, we used ELVIS (see Section 3.3), an entity linking integration tool, which provides a common output for different entity linking systems.

In addition, we created a dataset of annotated musical entities based on our corpus of documents, and applied both quantitative and qualitative evaluations in order to verify which system performs better with musical entities, and is more suitable for our task.

Evaluation data

We created an *ad-hoc* ground truth dataset to evaluate the different entity linking systems in an excerpt of the corpus where they will be later applied, the Songfacts dataset (Section 4.3.1). In this corpus, each document univocously refers to one single song. In addition, we have information about artist and song names at our disposal. We used this information to obtain the Mu-

	Album		Artist		Song		Macro Average		
	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec	F-measure
Babelfy	0.93	0.28	0.98	0.55	0.96	0.31	0.96	0.38	0.54
Tagme	0.75	0.69	0.97	0.77	0.65	0.71	0.79	0.72	**0.76**
Spotlight	0.80	0.52	0.94	0.83	0.59	0.42	0.78	0.59	0.67

Table 4.6: Precision and recall of the entity linking systems considered.

sicBrainz ID for songs and artists. In MusicBrainz, artist and song items sometimes have information about their equivalent Wikipedia page. We leveraged this information, when available, to obtain their corresponding DBpedia URIs. Finally, we obtained a mapping with DBpedia of 7,691 songs and 3,670 artists. From the DBpedia data about each song, we gathered their corresponding album name and URI, if available, obtaining information of about 2,092 albums. Then, for every document, we looked for exact string matches of the reported song, and its related album and artist names. Every detected entity is thus annotated with its DBpedia URI. At the end of this process, the newly created evaluation dataset contains 6,052 documents where 17,583 sentences are annotated with the following entities: 5,981 Song, 12,137 Artist and 1,722 Album entities. As mentioned in Section 4.2.3, there are typical cases of ambiguity in musical entities where songs, artists, and albums can potentially share the same name. Therefore, we manually corrected the entities detected in 212 documents where this kind of ambiguity was present.

Entity linking evaluation

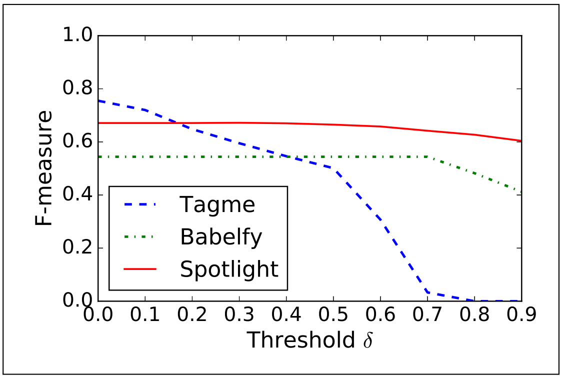


Figure 4.5: F-measure of the entity linking systems at different confidence thresholds.

The three entity linking systems under review provide their own confidence measure. Hence, we evaluated their output filtering out the entities with a confidence measure below a certain threshold δ. We ran the evaluation for different values of δ, ranging from 0 to 0.9 in bins of 0.1. After evaluating on the ground truth dataset, the best results in terms of F-measure were obtained by all the systems at $\delta = 0$ (see Figure 4.5), which means that there is no

	Album		Artist		Song		Macro Average		
	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec	F-measure
Babelfy	0.93	0.28	0.98	0.55	0.96	0.31	0.96	0.38	0.54
Tagme	0.75	0.69	0.97	0.77	0.65	0.71	0.79	0.72	**0.76**
Spotlight	0.80	0.52	0.94	0.83	0.59	0.42	0.78	0.59	0.67

Table 4.7: Precision and recall of the entity linking systems considered.

need to apply any filtering process based on the entity linking system own confidence score. Detailed results on the run of every system at $\delta = 0$ are shown in Table 4.7. We used macro-average Precision and Recall measures, i.e., we averaged their values from the three sets of entities.

We may conclude from these results that Babelfy is the system with highest Precision on musical entities. However, its recall is lower than the other systems under consideration, and specifically with respect to Tagme, which in turn, shows much lower precision. DBpedia Spotlight, on the other hand, achieves a similar precision score as Tagme, but with a slightly lower recall.

This evaluation experiment was only focused on measuring the precision in the annotation of entities present in the ground truth dataset. However, since all possible entities in a document may be not annotated, we also report on specific types of false positives which emerged during a qualitative inspection of classification results. For example, a frequent error that was not being evaluated concerns cases in which a text span not annotated in the ground truth was identified incorrectly as an entity by any system. Therefore, to complement the evaluation, we listed the most frequently identified entities by each system (see Table 4.8). As we can see, Babelfy and Tagme are misidentifying common words as entities very frequently, whereas DBpedia Spotlight is not doing so. These errors may propagate to the rest of the relation extraction pipeline, penalizing the accuracy of the final KB. Although a filtering process could be applied to filter out misidentified entities by computing their tf-idf score in each document, we opted for using DBpedia Spotlight, as it has shown pretty good performance, its output does not require any further processing, and it is released as open source, which means that there are no limitations on the number of queries.

4.4.2. Quality of relations

Relation extraction evaluation is not trivial, as semantic relations between entities may vary in terms of correctness over time. Also, correct relations may be linguistically flawed, i.e., not fluent. Previous approaches assessed automatically extracted relations in terms of correctness according to human judgment (Fader et al., 2011; Mausam et al., 2012). Additionally, a finer-grained analysis is carried out in Banko et al. (2007), adding a prior step in which relations are judged as being *concrete* or *abstract*.

System	Song	Album	Artist
Babelfy	**Carey**	**Debut**	John_Lennon
	Stephen	**Song_For**	Eminem
	Rap_Song	**Sort_Of**	Paul_McCartney
	Singing_This_Song	**First_Song**	Bob_Dylan
	A_Day_in_the_Life	**Debut_Album**	Drake
Tagme	**The_Word**	**Up!**	John_Lennon
	The_End	**When_We_On**	**The_Notorious_B.I.G.**
	If	Up	**Do**
	Once	**Together**	Paul_McCartney
	For_You	**By_the_Way**	**Neil_Young**
Spotlight	Sexy_Sadie	The_Wall	Madonna
	Helter_Skelter	Let_It_Be	Eminem
	Cleveland_Rocks	Born_This_Way	Rihanna
	Stairway_to_Heaven	Thriller	John_Lennon
	Minnie_the_Moocher	Robyn	Britney_Spears

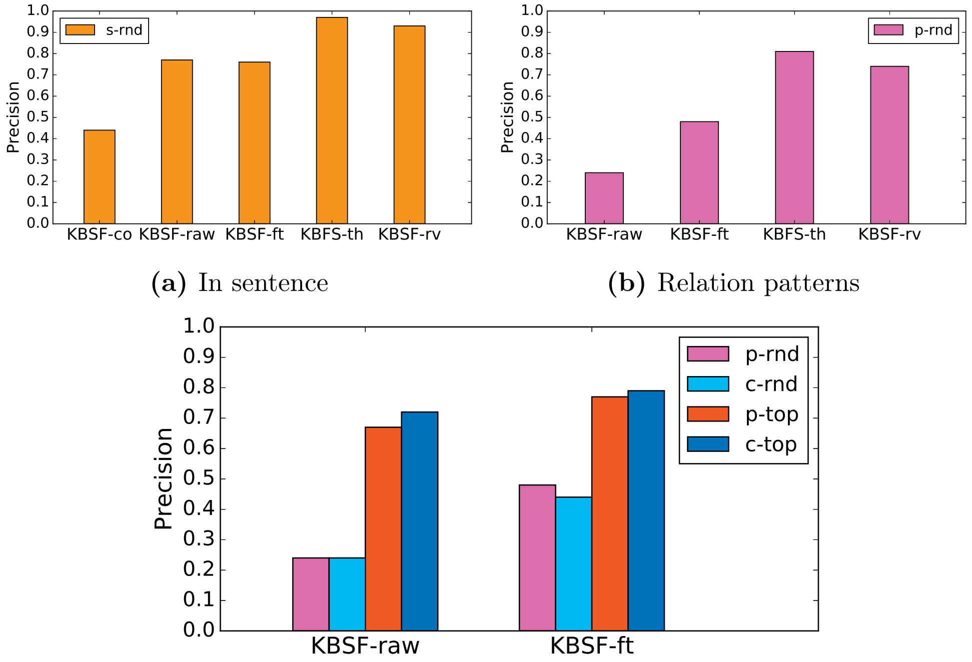
Table 4.8: Top-5 most frequent entities by type and tool. Disambiguation errors appear in bold.

In this chapter, we made use of extensive human input and asked two experts in Computational Linguistics to evaluate the *top 100* scoring relations as yielded by our weighting policy (Section 4.2.7), as well as a random sample of 100 relations. This was done for all the KBs produced by our pipeline and for KBSF-rv. Cohen's kappa coefficient among annotators ranged from 0.60 to 0.81, which is generally considered as *substantial* agreement.

In Figures 4.6a and 4.6b, where we compare random samples from each KB, we observe a gradual improvement of the quality of relations as the different modules of our implementation are incorporated. The difference between these figures is that in the former, a relation is deemed correct if it has extracted a relation *expressed in the original sentence*, whereas the latter figure reports numbers on whether the extracted relation pattern was correct, i.e., if it *meant* the same as it was intended in the source sentence. We may infer from the difference of precision between KBSF-co and KBSF-raw in Figure 4.6a that co-occurrence between entities does not guarantee an explicit relation, whereas the presence of a path between two entities over a sentence dependency tree, without any other entity mention in between, generally suggests a monosemous and unambiguous relation.

It is remarkable how well ReVerb performs (Figure 4.6b), only being surpassed by the KB resulting from the complete implementation described in this chapter. We note that the good results of the ReVerb extractor are also due to the semantic processing of our system, which is forcing ReVerb to select good candidates as relation arguments. Recall that the difference between KBSF-ft and KBSC-th is the inclusion of the *scoring* module, and the increase in Precision confirms that incorporating *statistical evidence contributes to better relations.*

This is further confirmed in the results showcased in Figure 4.6c, where we



(a) In sentence

(b) Relation patterns

(c) Patterns and clusters

Figure 4.6: Precision of relations at sentence (s), relation pattern (p) and cluster pattern (c) levels in top (top) and random (rnd) samples of relations

provide a comparison between top 100 relations according to our ranking policy against a random sample. Note that *in all KBs, highly scoring relations are more often marked as correct*, which constitutes additional support for the contribution of the scoring module. Together with the quality of the relation pattern, this figure shows the quality of the cluster pattern associated with the evaluated relations. We observe that cluster patterns inferred in our clustering module have similar quality than relation patterns in the random sample, and slightly better in the top 100 sample. This result implies that the scoring module is rewarding good clusters.

4.4.3. Coverage of the extracted knowledge base

With this experiment, we aimed to compare the coverage of music relations in our final KB with respect to other resources with human intervention, such as DBpedia, MusicBrainz, and with automatically created resources. For the latter, we considered DefIE (Bovi et al., 2015b) as our closest competitor due to several methodological similarities (dependency parsing, entity linking, and relation extraction over shortest paths).

We selected all triples in KBSF-th whose domain and range entities could be

	KBSF-th	MusicBrainz	DBpedia	DefIE
Relation instances	3,633	1,535	1,240	456

Table 4.9: Number of triples with labeled relations in the different KBs for the same set of domain-range entity pairs.

mapped to both DBpedia and MusicBrainz. In addition, since entities in DefIE are disambiguated against BabelNet ids, we mapped all DBpedia URIs to their corresponding BabelNet id. After mapping the entities, we obtained a subset of 3,633 triples. From here, we selected all possible pairs of domain-range entities present in these triples, and retrieved from the other KBs all triples involving the same pairs, and counted them. The procedure to do so on DBpedia was via SPARQL queries. From the retrieved triples after querying, we discarded those with predicate *wikiPageWikiLink*, as this predicate means an unlabeled relation. By contrast, the mapping with MusicBrainz was not trivial. MusicBrainz is not a KB of triples, but a relational database. Entities are stored in tables, and relations between entities are represented in a set of tables of relations, having one table for each possible relation. In addition, an entity of type Song in KBSF-th may refer to either a Recording or a Work entity in MusicBrainz (see Section 4.2.3). Therefore, for the analysis of relations involving a Song entity, we obtained the equivalent Recording and Work MusicBrainz entities, and looked up relations where any of them where involved.

Mapping results are shown in Table 4.9. Let us highlight the fact that most semantic relations encoded in KBSF-th are novel, as they were not found in any of the other resources we compared against. In the overlapping cases, most of the times the relation labels were semantically equivalent, and often the relation label of KBSF-th triples was more specific than the ones retrieved from other KBs (e.g., *frontman* vs. *member of*)

4.4.4. Interpretation of music recommendations

The main aim of this experiment was to evaluate the suitability of KBSF-th to explain relations between songs, and study their impact on user's experience in music recommendation. Since our aim was not to measure the performance of a recommender system, we implemented a baseline recommender approach. Recommendations were based on the concept of song similarity, which exploits the graph-based structure of our KB. Maximal common subgraph score is computed between the item neighborhood graphs of every song (this methodology for entity similarity is fully described in Section 6.2.3). Once the similarity scores are computed, similar songs are ranked.

We designed the experiment as an online survey, where the participant is first asked to select 5 songs from different artists of his/her choice. From each selected song, the system randomly selects 3 recommendations among the list of its top-10 most similar songs. One of them is shown together with an explanation in natural language (the source text), another with an explanation based on relation patterns, and finally the third one appears without explanation. Participants can listen to all songs with an embedded player. After listening to the recommendation and reading the explanation attached to it, participants were asked to rate each recommendation from 1 to 5 (1 being worst), and to mention whether they were familiar or not with the recommended songs (see Figure 4.7).

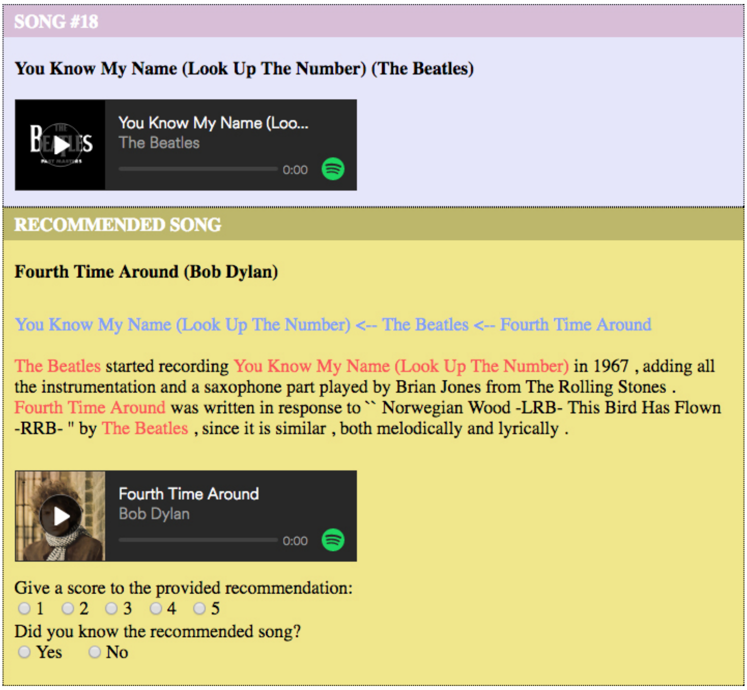
The experiment involved 35 participants, 28 males and 7 females, ranging from 26 to 38 years old and with different musical background and listening habits. Most of the participants said that they had previous experience with recommendation systems. A total of 525 answers (corresponding to individual song recommendations) were collected. In 38% of the cases, the user was familiar with the recommended songs.

The average rating of recommendations with natural language explanations is slightly higher (3.20±1.29) than recommendations without explanations (3.08±1.35), or with explanations based on relation labels (3.04±1.34). In addition, for musically educated individuals, recommendations of unfamiliar songs, whether accompanied with or without explanations, have similar average rating (2.87 and 2.95 respectively). However, for untrained users, recommendations with explanations have a remarkable higher average rating (2.93) than without them (2.36). Thus, we can infer that the introduction of explanations in recommender systems improves the user experience of musically untrained subjects when discovering songs.

We also asked the subjects to select among a set of adjectives (*enjoyable, useful, enriching, complicated, confusing,* and *too geeky*) those that better described the recommendation experience. The general trend was to rate positively the experiment. Most users rated the experience as *enjoyable* (40%), followed by *useful* (31%) and enriching (29%). Negativity was much lower in general, with *confusing* being the most voted (17%), followed by *complicated* and *too geeky* (8% in both cases). This suggests that the introduction of explanations generated from our MKB in the recommendations was in general a satisfactory experience to users.

4.5. Conclusion

We have presented an NLP pipeline that extracts a KB in the music domain taking raw text collections as input. It combines methods easily applicable to a general-purpose application with domain-specific heuristics which are designed to exploit particularities of the music domain.



SONG #18

You Know My Name (Look Up The Number) (The Beatles)

RECOMMENDED SONG

Fourth Time Around (Bob Dylan)

You Know My Name (Look Up The Number) <-- The Beatles <-- Fourth Time Around

The Beatles started recording You Know My Name (Look Up The Number) in 1967 , adding all the instrumentation and a saxophone part played by Brian Jones from The Rolling Stones . Fourth Time Around was written in response to `` Norwegian Wood -LRB- This Bird Has Flown -RRB- '' by The Beatles , since it is similar , both melodically and lyrically .

Give a score to the provided recommendation:
○ 1 ○ 2 ○ 3 ○ 4 ○ 5
Did you know the recommended song?
○ Yes ○ No

Figure 4.7: User interface for the music recommendation experiment.

The result of applying our approach over a dataset of stories about songs is a new MKB, which encodes semantic relations among musical entities. Our method relies on the syntactic structure (defined via dependency parsing) of sentences and the use and adaptation of music-specific heuristics for both entity linking and relation extraction. In addition, we include modules for semantic clustering and pattern scoring, aimed at the efficient removal of noisy relations. Our modular evaluation shows that our relation extraction module is able to capture a highly precise and compact set of weighted triples, and demonstrates the positive impact of the novel scoring metric we introduced. Moreover, we have shown that a high percentage of the knowledge encoded in our MKB is not present in other KBs, both general and domain-specific. Finally, regarding extrinsic evaluation, the experiment on recommendation interpretation confirms that explanations based on the extracted KB are positively regarded by the users.

CHAPTER 5

Applications in Musicology

5.1. Introduction

A vast amount of musical knowledge has been gathered for centuries by musicologists and music enthusiasts. Most of this knowledge is implicitly expressed in artist biographies, reviews, facsimile editions, etc. This context results in the existence of large repositories of unstructured knowledge, which have great potential for musicological studies. For instance, aggregating musical and musicological information after processing large collections of naturally occurring text can provide search engines with much richer and fine-grained information about musicians, their life and work, and even their relation with other musical entities.

In this chapter we propose to explore two use cases where we reconcile, on one hand, intelligent text processing techniques, and on the other, musical knowledge acquired both from structured and unstructured resources. In the first use case, we create a culture-specific knowledge base, in particular, a knowledge base of flamenco music. The methodology applied to its creation combines content aggregation from different data sources and knowledge extraction. Then, a methodology for the creation of a knowledge graph from a set of unstructured text documents using entity linking is proposed and tested for computing artist relevance ranking. Evaluation shows a high level of agreement between a flamenco expert and our system. In the second use case, we provide a diachronic study of music criticism via a quantitative analysis of the polarity associated to music album reviews gathered from `Amazon`[12]. Our analysis hints at a potential correlation between key cultural and geopolitical events and the language and evolving sentiments found in music reviews. In addition, trends observed in the data reveals to be useful to study the evolution of music genres.

[12] `http://www.amazon.com`

The rest of the chapter is organized as follows. First, in Section 5.2, we describe the process of creation of a culture-specific knowledge base. We begin introducing the problem and the context of application (Section 5.2.1). Then, we describe the obtained knowledge base (Section 5.2.2) and the processes of knowledge curation and extraction applied (Sections 5.2.3 and 5.2.4). Finally, we employ the knowledge base to compute artist relevance ranking and present some insights that can be drawn from computing statistics over the dataset (Section 5.2.5). In Section 5.3, we describe how sentiment associated with music reviews changes over time. We start by describing the dataset of music reviews used (Section 5.3.1) and the process of aspect-based sentiment analysis applied (Section 5.3.2). Then, two experiments are performed, one aggregating sentiment scores by review publication year (Section 5.3.3), and other by album publication year (Section 5.3.3). Finally, we conclude the chapter with a discussion about our findings (Section 5.4).

5.2. Building culture-specific knowledge bases: the flamenco case

Although some existing repositories of music information are quite complete and accurate, there is still a vast amount of music information out there, which is generally scattered across different sources on the Web. Hence, harvesting and combining that information is a crucial step in the creation of practical and meaningful music knowledge bases. In addition, the creation of culture-specific knowledge bases may be highly valuable for research and dissemination purposes, and can be particularly impactful in non-western traditions (Serra, 2014).

In this section, we propose a methodology for the creation of a culture-specific knowledge base; in particular, a knowledge base of flamenco music. The proposed methodology combines content curation and knowledge extraction processes. First, a large amount of information is gathered from different data sources, which are subsequently combined by applying pair-wise entity resolution. Next, new knowledge is extracted from unstructured texts and employed to populate the knowledge base. To this end, an *ad hoc* entity linking system has been developed. Finally, the content of the knowledge base is used to compute artist relevance and results are evaluated according to flamenco experts' criteria.

5.2.1. Flamenco music

Several musical traditions contributed to the genesis of flamenco music as we know it today. Among them, the influences of the Jews, Arabs, and Spanish folk music are recognizable, but indubitably the imprint of Andalusian Gypsies' culture is deeply ingrained in flamenco music. Flamenco occurs in a wide range of settings, including festive *juergas* (private parties), *tablaos* (flamenco venues), concerts, and big productions in theaters. In all these settings we find the main components of flamenco music: *cante* or singing, *toque* or guitar playing, and *baile* or dance. According to Gamboa (2005), flamenco music grew out of the singing tradition, as a melting process of all the traditions mentioned above, and therefore the role of the singer soon became dominant and fundamental. *Toque* is subordinated to *cante*, especially in more traditional settings, whereas *baile* enjoys more independence from voice.

In the flamenco jargon styles are called *palos*. Criteria adopted to define flamenco *palos* are rhythmic patterns, chord progressions, lyrics and its poetic structure, and geographical origin. In flamenco geographical variation is important to classify *cantes* as often they are associated to a particular region where they were originated or where they are performed with gusto. Rhythm or *compás* is a unique feature of flamenco. Rhythmic patterns based on 12-beat cycles are mainly used. Those patterns can be classed as follows: binary patterns, such as *tangos* or *tientos*; ternary patterns, which are the most common ones, such as *fandangos* or *bulerías*; mixed patterns, where ternary and binary patterns alternate, such as *guajira*; free-form, where there is no a clear underlying rhythm, such as *tonás*. For further information on fundamental aspects of flamenco music, see the book by Fernández (2004). For a comprehensive study of styles, musical forms and history of flamenco the reader is referred to the books of Blas Vega & Ríos Ruiz (1988), Navarro & Ropero (1995), and Gamboa (2005), and the references therein.

5.2.2. FlaBase

FlaBase (Flamenco Knowledge Base) is the acronym of a new knowledge base of flamenco music. Its ultimate aim is to gather all available online editorial, biographical and musicological information related to flamenco music. Its content is the result of the curation and extraction processes explained in Sections 5.2.3 and 5.2.4. FlaBase contains information about 1,174 artists, 76 *palos* (flamenco genres), 2,913 albums, 14,078 tracks, and 771 Andalusian locations.

Ontology definition

The FlaBase data structure is defined following an ontology schema. One of the advantages of using an ontology is that it can be easily modified. Thus, our design is a first building block that can be enhanced and redefined in the future. The initial ontology is structured around five main classes: MusicArtist, Album, Track, Palo, and Place, and three domain-specific artist subclasses: *Cantaor* (flamenco singer), *Guitarist* (flamenco guitar player), and *Bailaor* (flamenco dancer). These three classes were defined because they are the most

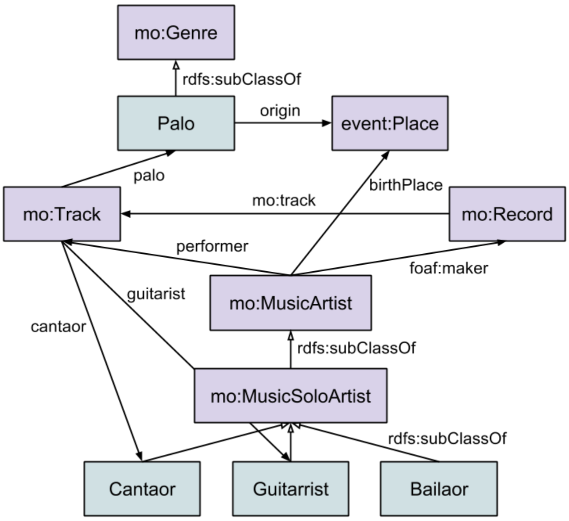


Figure 5.1: Ontology schema

frequent types of artists in the data. Other instrument players may be instantiated directly from the MusicArtist class. A diagram with the main classes and some properties of the ontology is shown in Figure 5.1. We have tried to reuse as much vocabulary as we could. We re-utilized most of the classes and some properties from the Music Ontology[13], a standard model for publishing music-related data. We selected the classes according to the ones used by the LinkedBrainz project[14], which maps concepts from MusicBrainz to the Music Ontology.

5.2.3. Content curation

The first step towards building a domain-specific knowledge base is to gather all possible content from available data sources. This implies at least two problems: data gathering, and matching between entities from different sources (entity resolution). In what follows we enumerate the involved data sources and describe the methodology applied for entity resolution.

Data acquisition

Our aim is to gather a significant amount of information about musical entities, including textual descriptions and available metadata. A schema of the selected data sources is shown in Figure 5.2. We started by looking at Wikipedia. Each Wikipedia article may have a set of associated categories. Categories

[13] `http://musicontology.com`

[14] `https://wiki.musicbrainz.org/LinkedBrainz`

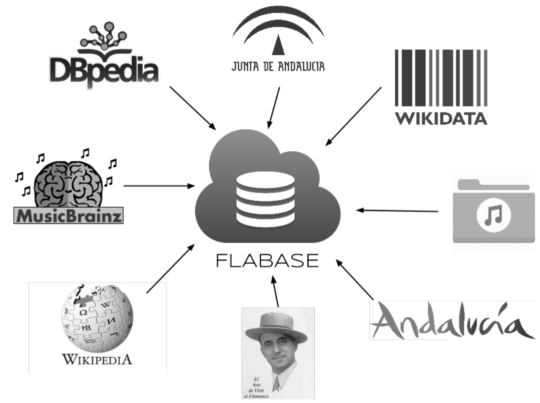


Figure 5.2: Selected data sources

are intended to group together pages on similar subjects and are structured in a taxonomical way. To find Wikipedia articles related to flamenco music, we first looked for flamenco categories. We queried the Spanish version of DBpedia[15] for categories related to the flamenco category. We obtained 17 different categories (e.g., *cantaores de flamenco, guitarristas de flamenco*).

We gathered all DBpedia resources related to at least one of these categories. We obtained a total number of 438 resources in Spanish, of which 281 were also in English. Each DBpedia resource is associated with a Wikipedia article. Text and HTML code were then extracted from Wikipedia articles in English and Spanish. Next, we classified the extracted articles according to our ontology (Section 5.2.2). For this purpose, we exploited classification information provided by DBpedia (DBpedia types and Wikipedia categories). At the end, from all gathered resources, we only kept those related to artists and *palos*, totaling 291 artists and 56 *palos*.

As the amount of information present in Wikipedia related to flamenco music is somewhat scarce, we decided to expand our knowledge base with information from two different websites. First, *Andalucia.org*, the touristic web from the Andalusia Government[16]. It contains 422 artist biographies in English and Spanish, and the description of 76 *palos* also in both languages. Second, a website called *El arte de vivir el flamenco*[17], which includes 749 artist biographies among *cantaores*, *bailaores* and guitarists.

We used MusicBrainz to fill our knowledge base with information about flamenco album releases and recordings. For every FlaBase artist mapped to MusicBrainz, all content related to releases and recordings was gathered. Thus,

[15] `http://es.dbpedia.org`

[16] `http://andalucia.org`

[17] `http://www.elartedevivirelflamenco.com/`

814 releases and 9,942 recordings were collected.

The information gathered from MusicBrainz is a little part of the actual flamenco discography. Therefore, to complement it we used a flamenco recordings database gathered by Rafael Infante and available at CICA website[18] (Computing and Scientific Center of Andalusia). This database has information about releases from the early time of recordings until present time, counting 2,099 releases and 4,136 songs. For every song entry, a *cantaor* name is provided, and most of the times also guitarist and *palo*, which is an important piece of information to define flamenco recordings.

Finally, we supplied our knowledge base with information related to Andalusian towns and provinces. We gathered this information from the official database SIMA[19] (Multi-territorial System of Information of Andalusia).

Entity resolution

Entity resolution is the problem of extracting, matching, and resolving entity mentions in structured and unstructured data (Getoor, 2012). There are several approaches to tackle the entity resolution problem. For the scope of this research, we selected a pair-wise classification approach based on string similarity between entity labels.

The first issue after gathering the data is to decide whether two entities from different sources are referring to the same one. Therefore, given two sets of entities A and B, the objective is to define an injective and non-surjective mapping function f between A and B that decides whether an entity $a \in A$ is the same as an entity $b \in B$. To do that, a string similarity metric $sim(a, b)$ based on the Ratcliff-Obershelp algorithm (Ratcliff & Metzener, 1988) has been applied. It measures the similarity between two entity labels and outputs a value between 0 and 1. We consider that a and b are the same entity if their similarity is bigger than a parameter θ. If there are two entities $b, c \in B$ that satisfy that $sim(a, b) \geq \theta$ and $sim(a, c) \geq \theta$, we consider only the mapping with the highest score. To determine the value of θ, we tested the method with several θ values over an annotated dataset of entity pairs. To create this dataset, the 291 artists gathered from Wikipedia were manually mapped to the 422 artists gathered from Andalucia.org, obtaining a total amount of 120 pair matches. As it is shown in Figure 5.3 the best F-measure (0,97) was obtained with $\theta = 0.9$. Finally, we applied the described method with $\theta = 0.9$ to all gathered entities from the three data sources. Thanks to the entity resolution process, we reduced the initial set of 1,462 artists and 132 *palos* to a set of 1,174 artists and 76 *palos*.

[18] http://flun.cica.es/index.php/grabaciones

[19] http://www.juntadeandalucia.es/institutodeestadisticaycartografia/sima

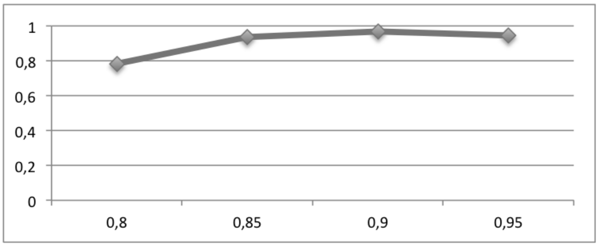


Figure 5.3: F-measure for different values of θ

Once we had our artist entities resolved, we began to gather their related discography. First, we tried to find out the MusicBrainz ID of the gathered artists. Depending on the information about the entity, two different processes were applied. First, we leveraged mapping information between Wikipedia and MusicBrainz present in Wikidata[20]. Wikidata is a free linked database, which acts as a structured data storage of Wikipedia. For those artists without this mapping information, we queried the MusicBrainz API, and then applied our entity resolution method to the obtained results.

Finally, to integrate the discography database of CICA into our knowledge base, we applied the entity resolution method to the fields *cantaor*, guitarist and *palo* of each recording entry in the database. From the set of 202 *cantaores* and 157 guitarists names present in the recording entries of the database, a total number of 78 *cantaores* and 44 guitarists were mapped to our knowledge base. The number of mapped artists was low due to differences between the way of labeling an artist. An artist name may be written using one or two surnames, or using a nickname. In the case of *palos*, there were 162 different *palos* in the database, 54 of which were mapped with the 76 of our knowledge base. These 54 *palos* correspond to an 80% of *palo* assignments present in the recording entries.

5.2.4. Knowledge extraction

While the resulting knowlege base does already encode relevant culture and music-specific information, a notable portion of the data collected during the knowledge base creation process currently remains unexploited due to its unstructured nature. Consequently, to enhance the amount of structured data in FlaBase, a process of knowledge extraction has been carried out. This implicit knowledge may vary from biographical data, such as place and date of birth, to more complex semantic relations involving different entities. In this section, we focus on named entity recognition (NER) and entity linking (EL) tasks. In what follows, our *ad hoc* system for entity linking is described and evaluated.

[20] https://www.wikidata.org

Lastly, an information extraction process is applied to populate the knowledge base.

Entity linking

In order to extract knowledge from text, the first step is to semantically annotate it identifying all entity mentions. In entity linking, disambiguation can be applied to n-grams extracted from text, or to the output of a NER system. We propose a method that employs a combination of both approaches, depending on the category of the entity. For NER, we used the Stanford NER system (Finkel et al., 2005), implemented in the library Stanford Core NLP[21] and trained on Spanish texts.

For the scope of this research, we focused on Spanish texts, as flamenco texts are mostly written in Spanish. Although there are many entity linking tools available, state-of-the-art systems are well-tuned for English texts, but may not perform as well in languages other than English, and even less with music-related texts (see Section 3.2). In addition, we wanted to have a system that uses our own knowledge base for disambiguation. Therefore, we developed our own system, which is able to detect and disambiguate three categories of entities: Person, *Palo*, and Location. Three different approaches were defined by combining NER and n-grams in the selection of annotation candidates: only using n-grams; disambiguating Location and Person entities from the NER output, and *Palo* from text n-grams; and only disambiguating Location entities from the NER output, and Location and *Palo* directly from text n-grams.

To determine which approach performs better, three artist biographies coming from three different data sources were manually annotated, having a total number of 49 annotated entities. Results on the different approaches are shown in Table 5.1. We observe that applying NER to entities of the Person category worsens performance significantly, as recall suddenly decreases by half. After manually analyzing false negatives, we observed that this is caused because many artist names have definite articles between name and surname (e.g., *de, del*), and this is not recognized by the NER system. In addition, many artists have a nickname that is not interpreted as a Person entity by the NER system. The best approach is the third (NER to LOC), which is slightly better than the first (no NER) in terms of precision. This is due to the fact that many artists have a town name as a surname or as part of his or her nickname. Therefore, applying entity linking directly to text n-grams is misclassifying Person entities as Location entities. Thus, by adding a previous step of NER to Location entities we have increased overall performance, as it can be seen on the F-measure values.

[21] `http://nlp.stanford.edu/software/corenlp.shtml`

Approach	Precision	Recall	F-measure
1) no NER	0.829	**0.694**	0.756
2) NER to PERS & LOC	0.739	0.347	0.472
3) NER to LOC	**0.892**	0.674	**0.767**

Table 5.1: Precision, Recall and F-measure of entity linking approaches.

Knowledge base population

A process of information extraction is necessary to transform the unstructured information present in FlaBase into structured knowledge. For the scope of this research, we focused on extracting two specific pieces of information from the gathered artist biographies: birth year and birth place, as they can be relevant for anthropological studies. We observed that this information is often in the first sentences of the biographies, and always near the word *nació* (Spanish translation of “was born"). Therefore, to extract this information, we looked for this word in the first 250 characters of every biographical text. If it is found, we apply our entity linking method to this piece of text. If a Location entity is found near the word "nació", we assume that this entity is the place of birth of the biography subject. In addition, by using regular expressions, we look for the presence of a year expression in the context of the Location entity. If it is found, we assume it as the year of birth. If more than one year is found, we select the one with the smaller value.

To evaluate our approach, we tested the extraction of birth places in all texts coming from the web Andalucia.org (442 artists). We manually annotated the province of provenance of these 442 artists for building ground truth data. After the application of the extraction process on the annotated test set, we obtained a precision value of 0,922 and a recall of 0,648. Therefore, we may argue that our method is extracting biographic information with high precision and quite reasonable recall. We finally applied the extraction process to all artist entities with biographical texts. Thus, 743 birth places and 879 birth years were extracted.

5.2.5. Looking at the data

Artist relevance

We assume that an entity mention inside an artist biography signals a semantic relation between the entity that constitutes the main theme of the biography (subject entity) and the mentioned entity. Based on this assumption, we built a semantic graph by applying the following steps. First, each artist of the knowledge base is added to the graph as a node. Second, entity linking is applied to artist’s biographical texts. For every linked entity identified in the biography, a new node is created in the graph (only if it was not previously

Cantaor	Guitarist	*Bailaor*
Antonio Mairena	Paco de Lucía	Antonio Ruiz Soler
Manolo Caracol	Ramón Montoya	Rosario
La Niña de los Peines	Niño Ricardo	Antonio Gades
Antonio Chacón	Manolo Sanlúcar	Mario Maya
Camarón de la Isla	Sabicas	Carmen Amaya

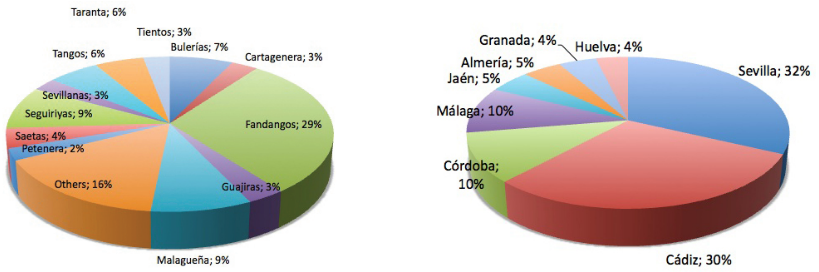
Table 5.2: PageRank Top-5 artists by category.

	Top-5	Top-10
PageRank	0.933	0.633
HITS Authority	0.6	0.4

Table 5.3: Precision values of artist relevance ranking.

created). Next, an edge is added, connecting the subject entity with the linked entity found in its biography. This way, a directed graph connecting the entities of FlaBase is finally obtained. Entities identified in a text can be seen as hyperlinks. Thus, algorithms to measure the relevance of nodes in a network of hyperlinks can be applied to our semantic graph (Bellomi & Bonato, 2005). In order to measure artist relevance, we applied PageRank (Brin & Page, 1998) and HITS (Kleinberg, 1999) algorithms to the obtained graph.

Using this approach, we built an ordered list with the top-10 entities of the different artist categories (*cantaor*, guitarist and *bailaor*) for the two algorithms. For evaluation purposes, we asked a reputed flamenco expert to build a list of top-10 artists for each category according to his knowledge and the available bibliography. The concept of artist relevance is somehow subjective and there is no unified or consensual criterion for flamenco experts about who the most relevant artists are. Despite that, there is a high level of agreement among them on certain artists that should be on such a hypothetical list. Thus, the expert provided us with this list of hypothetical top-10 artists by category and we considered it as ground truth. We define precision as the number of identified artists in the resulting list that are also present in the ground truth list divided by the length of the list. We evaluated the output of the two algorithms by calculating precision over the entire list (top-10), and over the first five elements (top-5) (see Table 5.3). We can observe that PageRank results (see Table 5.2) show the greatest agreement with the flamenco expert. High values of precision, especially for the top-5 list, indicates that the content gathered in FlaBase is highly complete and accurate (see Table 5.3), and the proposed methodology adequate to compute relevance of artists.



(a) Songs by *palo* **(b)** Artists by province of birth

Figure 5.4: FlaBase distributions.

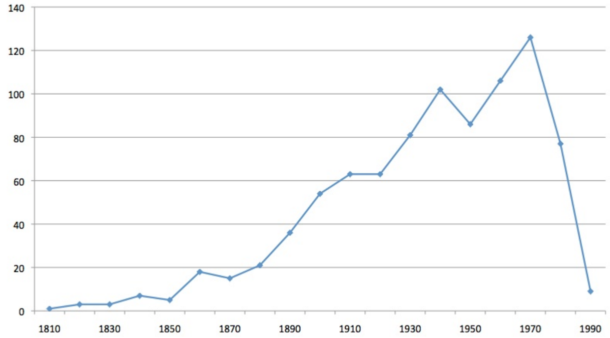


Figure 5.5: Artists by decade of birth

Statistics

For the sake of completeness, we computed the distribution of different items present in FlaBase. Data shown in Figure 5.4a was produced after the knowledge acquisition process, while data shown in Figures 5.4b and 5.5 was obtained thanks to the knowledge extraction process. In Figure 5.4a it is shown that the most representative *palos* in flamenco music are represented in our knowledge base, with a higher predominance of fandangos. We can observe in Figure 5.4b that most flamenco artists are from the Andalusian provinces of Seville and Cadiz. Finally, in Figure 5.5 we observe a higher number of artists in the data were born from the 30's to the 80's of the 20th century.

5.3. Diachronic study of music criticism

In this Section, we put forward an integration procedure for enriching a large corpus of Amazon customer reviews (McAuley et al., 2015a,b), with metadata obtained from MusicBrainz. In addition, we further extend the *semantics* of the textual content with the application of an aspect-based sentiment analysis framework (Dong et al., 2013), which provides specific sentiment scores for different aspects present in the text, e.g., album cover, guitar, voice, or lyrics.

This enriched dataset, henceforth referred to as Multimodal Album Reviews Dataset (MARD), includes affective features and music metadata. We benefit from this substantial amount of information at our disposal for performing a diachronic analysis of music criticism. Specifically, we combine the metadata retrieved for each review with their associated sentiment information, and generate visualizations to help us investigate any potential trends in diachronic music appreciation and criticism. Based on this evidence, and since music evokes emotions through mechanisms that are not unique to music (Juslin & Västfjäll, 2008), we may go as far as using musical information as means for a better understanding of global affairs. Previous studies argue that national confidence may be expressed in any form of art, including music (Moïsi, 2010), and in fact, there is strong evidence suggesting that our emotional reactions to music have important and far-reaching implications for our beliefs, goals, and actions, as members of social and cultural groups (Alcorta et al., 2008). Our analysis hints at a potential correlation between the language used in music reviews and major geopolitical events or economic fluctuations. Finally, we argue that applying sentiment analysis to music-related text corpora may be useful for diachronic musicological studies.

5.3.1. Dataset

The collected dataset contains texts and accompanying metadata originally obtained from a much larger dataset of Amazon customer reviews (McAuley et al., 2015a,b). The original dataset provides millions of review texts together with additional information such as overall rating (between 0 to 5), date of publication, or creator id. Each review is associated to a product and, for each product, additional metadata is also provided, namely Amazon product id, list of similar products, price, sell rank, and genre categories. From this initial dataset, we selected the subset of products categorized as *CDs & Vinyls*, which also fulfill the following criteria. First, considering that the Amazon taxonomy of music genres contains 27 labels in the first hierarchy level, and about 500 in total, we obtain a music-relevant subset and select 16 of the 27 which really define a music style and discard for instance region categories (e.g., World Music) and other categories not specifically related to a music style (e.g., Soundtrack, Miscellaneous, Special Interest), function-oriented categories (Karaoke, Holiday & Wedding) or categories whose albums might also be found under other categories (e.g., Opera & Classical Vocal, Broadway & Vocalists). We compiled albums belonging only to one of the 16 selected categories, i.e., no multi-label. Note that the original dataset contains not only reviews about CDs and Vinyls, but also about music DVDs and VHSs. Since these are not strictly speaking music audio products, we filter out those products also classified as "Movies & TV". Finally, since products classified as Classical and

Pop are substantially more frequent in the original dataset, we compensate this unbalance by limiting the number of albums of any genre to 10,000. After this preprocessing, MARD amounts to a total of 65,566 albums and 263,525 customer reviews. A breakdown of the number of albums per genre is provided in Table 5.4.

Genre	**Amazon**	**MusicBrainz**
Alternative Rock	2,674	1,696
Reggae	509	260
Classical	10,000	2,197
R&B	2,114	2,950
Country	2,771	1,032
Jazz	6,890	2,990
Metal	1,785	1,294
Pop	10,000	4,422
New Age	2,656	638
Dance & Electronic	5,106	899
Rap & Hip-Hop	1,679	768
Latin Music	7,924	3,237
Rock	7,315	4,100
Gospel	900	274
Blues	1,158	448
Folk	2,085	848
Total	66,566	28,053

Table 5.4: Number of albums by genre with information from the different sources in MARD.

Having performed genre filtering, we enrich MARD by extracting artist names and record labels from the Amazon product page. We pivot over this information to query the MusicBrainz search API to gather additional metadata such as release id, first release date, song titles and song ids. Mapping with MusicBrainz is performed using the same methodology described in Section 5.2.3, following a pair-wise entity resolution approach based on string similarity with a threshold value of $\theta = 0.85$. We successfully mapped 28,053 albums to MusicBrainz.

5.3.2. Sentiment analysis

Following the work of Dong et al. (2013, 2014) we use a combination of shallow NLP, opinion mining, and sentiment analysis to extract opinionated features from reviews. For reviews R_i of each album, we mine bi-grams and single-noun aspects (or review features), see Hu & Liu (2004); e.g., bi-grams which conform to a noun followed by a noun (e.g., *chorus arrangement*) or an adjective followed by a noun (e.g., *original sound*) are considered, excluding bi-grams whose adjective is a sentiment word (e.g., *excellent*, *terrible*). Separately, single-noun aspects are validated by eliminating nouns that are rarely associated with sentiment words in reviews, since such nouns are unlikely to refer to item aspects. We refer to each of these extracted aspects A_j as review aspects.

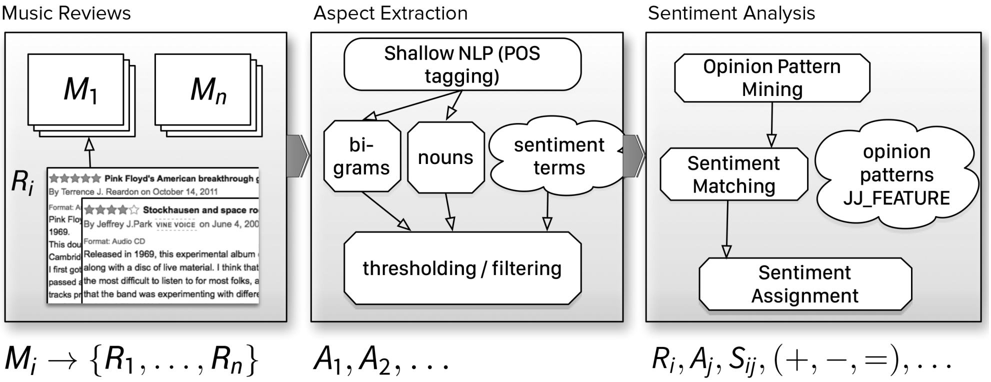

Music Reviews
Aspect Extraction
Sentiment Analysis
M_1
M_n
R_i
Pink Floyd's American breakthrough
By Terrence J. Reardon on October 14, 2011
Stockhausen and space ro
By Jeffrey J.Park VINE VOICE on June 4, 200
Format: Audio CD
Released in 1969, this experimental album
along with a disc of live material. I think that
the most difficult to listen to for most folks, a
that the band was experimenting with differe
Shallow NLP (POS tagging)
bi-grams
nouns
sentiment terms
thresholding / filtering
Opinion Pattern Mining
Sentiment Matching
opinion patterns JJ_FEATURE
Sentiment Assignment
$M_i \to \{R_1, \ldots, R_n\}$
$A_1, A_2, \ldots$
$R_i, A_j, S_{ij}, (+, -, =), \ldots$

Figure 5.6: Overview of the opinion mining and sentiment analysis framework.

For a review aspect A_j we determine if there are any sentiment words in the sentence containing A_j. If not, A_j is marked neutral, otherwise we identify the sentiment word w_{min} with the minimum word-distance to A_j. Next we determine the part-of-speach tags for w_{min}, A_i and any words that occur between w_{min} and A_i. We assign a sentiment score between -1 and 1 to A_j based on the sentiment of w_{min}, subject to whether the corresponding sentence contains any negation terms within 4 words of w_{min}. If there are no negation terms, then the sentiment assigned to A_j is that of the sentiment word in the sentiment lexicon; otherwise this sentiment is reversed. Our sentiment lexicon is derived from SentiWordNet (Esuli & Sebastiani, 2006) and is not specifically tuned for music reviews. An overview of the process is shown in Figure 5.6. The end result of sentiment analysis is that we determine a sentiment score S_{ij} for each aspect A_j in review R_i. A sample annotated review is shown in Figure 5.7. Finally, the sentiment score of a review R_i is calculated as the average of the sentiment score S_{ij} of every aspect A_j in R_i.

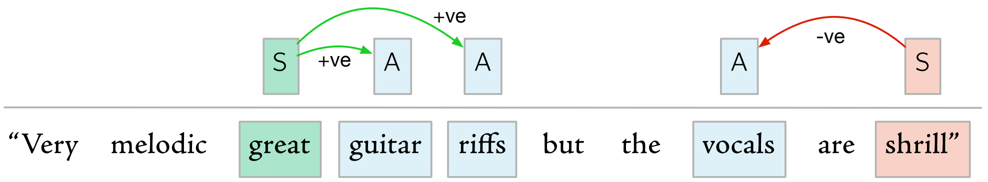


Figure 5.7: A sentence from a sample review annotated with opinion and aspect pairs.

5.3.3. Experiments

We carried out a study of the evolution of music criticism from two different temporal standpoints. Specifically, we consider when the review was written and, in addition, when the album was first published. We define the sentiment score of a review as the average score of all aspects in the review. Since we have sentiment information available for each review, we first computed an average sentiment score for each year of review publication (between 2000 and 2014). In this way, we may detect any significant fluctuation in the evolution of affective language during the 21st century. Then, we also calculated an average sentiment score by year of album publication. This information is complemented with the averages of the Amazon rating scores.

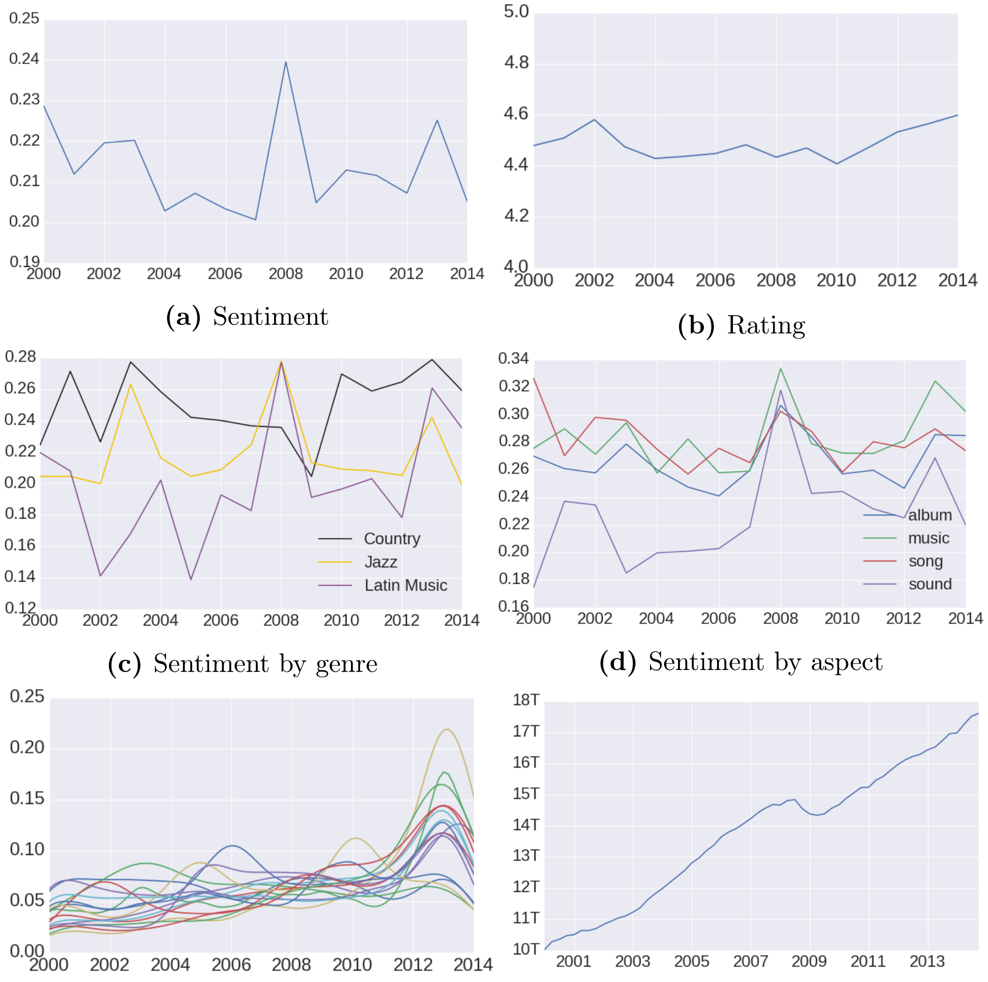
In what follows, we show visualizations for sentiment scores and correlation with ratings given by Amazon users, according to these two different temporal dimensions. Although arriving to musicological conclusions is out of the scope of this chapter, we provide *food for thought* and present the readers with hypotheses that may explain some of the facts revealed by these data-driven trends.

Evolution by review publication year

We applied sentiment and rating average calculations to the whole MARD dataset, grouping album reviews by year of publication of the review. Figure 5.8a shows the average of the sentiment scores of all the reviews published in a specific year, whilst Figure 5.8b shows average review ratings per year. At first sight, we do not observe any correlation between the trends illustrated in the figures. However, the sentiment curve (Figure 5.8a) shows a remarkable peak in 2008, a slightly lower one in 2013, and a low between 2003 and 2007, and also between 2009 and 2012. Figure 5.8e shows the kernel density estimation of the distribution of reviews by year of the 16 genres. The shapes of these curves suggest that the 2008 peak in the sentiment score is not related to the number of reviews published that year. The peak persists if we construct the graphs with the average sentiment associated with the most repeated aspects in text (Figure 5.8d). It is not trivial to give a proper explanation of this variations on the average sentiment. We speculate that these curve fluctuations may suggest some influence of economical or geopolitical circumstances in the language used in the reviews, such as the 2008 election of Barack Obama as president of the US. As stated by the political scientist Dominique Moïsi in Moïsi (2010):

> In November 2008, at least for a time, hope prevailed over fear. The wall of racial prejudice fell as surely as the wall of oppression had fallen in Berlin twenty years earlier [...] Yet the emotional dimension of this election and the sense of pride it created in many Americans must not be underestimated.

If we calculate the sentiment evolution curve for the different genres (see Figure 5.8c), we observe that 2008 constitutes an all-time-high for almost all genres. It is remarkable that genres traditionally related to more diverse com-



(a) Sentiment

(b) Rating

(c) Sentiment by genre

(d) Sentiment by aspect

(e) Kernel density est.

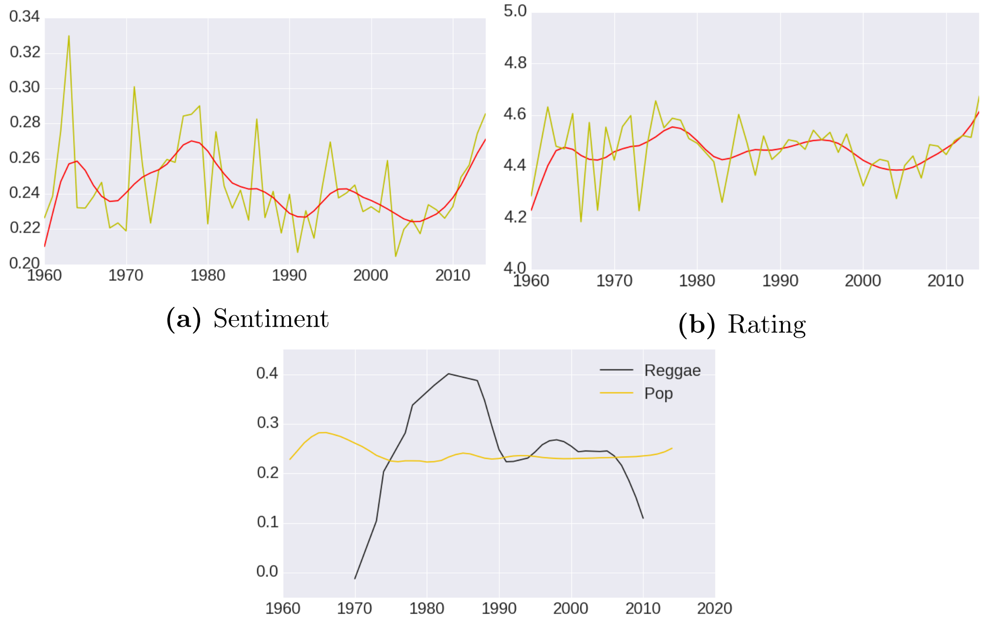
(f) USA GDP trend

Figure 5.8: Sentiment (a, c, and d) and rating (b) averages by review publication year; Kernel density estimation of the distribution of reviews by year (e); GDP trend in USA from 2000 to 2014 (f)

munities such as Jazz and Latin Music experience such an increase, whilst other genres such as Country do not.

Another factor that might be related to the positiveness in use of language is the economical situation. After several years of continuous economic growth, in 2007 a global economic crisis started[22], whose consequences were visible in the society after 2008 (see Figure 5.8f). In any case, further study of the different implied variables is necessary to reinforce any of these hypotheses.

[22] https://research.stlouisfed.org



(a) Sentiment

(b) Rating

(c) Sentiment by genre

Figure 5.9: Sentiment (a), rating (b), and sentiment by genres (c) averages by album publication year.

Evolution by album publication year

In this case, we study the evolution of the polarity of language by grouping reviews according to the album publication date. This date was gathered from MusicBrainz, meaning that this study is conducted on the 42.1% of the MARD that was successfully mapped. We compared again the evolution of the average sentiment polarity (Figure 5.9a) with the evolution of the average rating (Figure 5.9b). Contrary to the results observed by review publication year, here we observe a strong correlation between ratings and sentiment polarity. To corroborate that, we computed first a smoothed version of the average graphs, by applying 1-D convolution (see line in red in Figures 5.9a and 5.9b). Then we computed Pearson's correlation between smoothed curves, obtaining a correlation $r = 0.75$, and a p-value $p \ll 0.001$. This means that in fact there is a strong correlation between the polarity identified by the sentiment analysis framework in the review texts, and the rating scores provided by the users. This correlation reinforces the conclusions that may be drawn from the sentiment analysis data.

To further dig into the utility of this polarity measure for studying genre evolution, we also computed the smoothed curve of the average sentiment by genre, and illustrate it with two idiosyncratic genres, namely *Pop* and *Reggae* (see Figure 5.9c). We observe in the case of *Reggae* that there is a time period where reviews have a substantial use of a more positive language between the second half of the 70s and the first half of the 80s, an epoch which is often called the golden age of *Reggae* (Alleyne & Dunbar, 2012). This might be related to the publication of Bob Marley albums, one of the most influential artists in this genre, and the worldwide spread popularity of reggae music. In the case of *Pop*, we observe a more constant sentiment average. However, in the 60s and the beginning of 70s there are higher values, probably consequence by the release of albums by The Beatles. These results show that the use of sentiment analysis on music reviews over certain timelines may be useful to study genre evolution and identify influential events.

5.4. Conclusions

In this Chapter we have shown two different use cases in the context of making sense of large amounts of music related documents from a musicological perspective. (1) A culture-specific music knowledge base has been created, applying a process of knowledge curation, which combines information coming from different data sources. In addition, the knowledge base has been enriched with content extracted directly from unstructured texts by using a custom entity linking system. A methodology to build knowledge graphs is described and tested for computing artist relevance ranking. Evaluation shows high correlation between the obtained ranking of artists and the opinion of a flamenco expert. (2) A diachronic study of the sentiment polarity expressed in customer reviews from two different standpoints has been presented. First, an analysis by year of review publication suggests that geopolitical events or macro-economical circumstances may influence the way people speak about music. Second, an analysis by year of album publication shows how sentiment analysis can be useful to study the evolution of music genres. Moreover, according to the observed trend curves, we can state that we are now in one of the happiest periods of the recent history of music.

In conclusion, the main contribution of the work presented in this chapter is a demonstration of the utility of applying systematic linguistic processing on texts about music. Although further work is necessary to elaborate on the hypotheses or claims that may be derived from purely data-driven analyses, the proposed methodologies have shown their suitability in the quest of knowledge discovery from large amounts of documents, which may be highly useful for musicologists and humanities researchers in general.

CHAPTER 6

Semantic Enrichment for Similarity and Classification

6.1. Introduction

This chapter describes several methods for the semantic enrichment of music documents using entity linking and their application in the context of two widely studied MIR tasks, artist similarity and music genre classification. First, a method for computing semantic similarity at document-level is presented. The cornerstone of this approach is the intuition that semantifying and formalizing relations between entities in documents (both at in-document and cross-document levels) can represent the relatedness of two documents. Specifically, in the task of artist similarity, this derives in a measure to quantify the degree of relatedness between two artists by looking at their biographies. The evaluation results indicate that semantic-based approaches clearly outperform a baseline based on shallow word co-occurrence metrics. Second, we perform experiments on music genre classification from album customer reviews, exploring a variety of feature types, including semantic features obtained through entity linking, sentimental features and acoustic features. These experiments show that modeling semantic information contributes to outperforming strong bag-of-words baselines.

The remainder of this chapter is structured as follows: Section 6.2 describes a methodology for computing artist similarity from artist biographies using semantic information. Within this section, different types of knowledge representations and similarity measures are described (Sections 6.2.2 and 6.2.3). Then, the settings in which experiments were carried out together with the evaluation metrics used are presented (Section 6.2.4). Finally, evaluation results are presented and the performance of our method discussed (Section 6.2.5). Section 6.3 describes a methodology for computing music genre classification using album customer reviews. Within this section, a dataset of music reviews is first described (Section 6.3.1). Then, the linguistic processes applied and the different types of employed features are outlined (Sections 6.3.2 and 6.3.3). An experiment on genre classification is performed and results are discussed (Sections 6.3.5 and 6.3.6). Finally Section 6.4 summarizes the main topics covered in this chapter.

6.2. Artist similarity

We propose a method for leveraging semantic information extracted from music-related documents and knowledge repositories, for the computation of a similarity measure between musical entities. In this case we focus on computing similarity between artists based on their biographies.

The proposed method can be divided in three main steps, as depicted in Fig 6.1. The first step consists on the application of an entity linking process to every document. The second step derives a semantically motivated knowledge representation from the identified entities. This can be achieved by exploiting natural language text as anchor between entities, or by incorporating semantic information from an external Knowledge Base. In the latter case, a document is represented either as a semantic graph or as a set of semantic vectors projected on a vector space. Finally, the third step computes a similarity measure between documents (artist biographies in our case) based on the obtained knowledge representations.

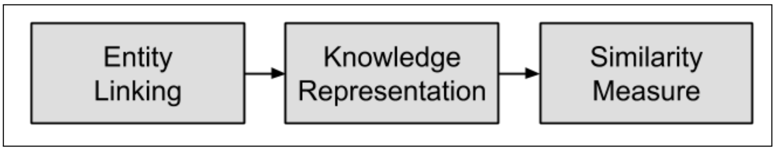


Figure 6.1: Workflow of the proposed method.

6.2.1. Entity linking

To obtain the entity mentions present in the documents and link them to a Knowledge Base we used Babelfy (Moro et al., 2014a) through ELVIS (see Section 3.3). Babelfy provides BabelNet URIs, and ELVIS enriches the information of every identified entity with DBpedia URIs, DBpedia Ontology types, and Wikipedia categories. We opted to use Babelfy for consistency purposes, as in a later step we exploit *SensEmbed* (Iacobacci et al., 2015), a vector space representation of concepts based on BabelNet (Navigli & Ponzetto, 2010). Moreover, the use of a single tool across approaches guarantees that the evaluation will only reflect the appropriateness of each one of them, and in case of error propagation all the approaches will be affected the same.

6.2.2. Knowledge representation

Relations graph

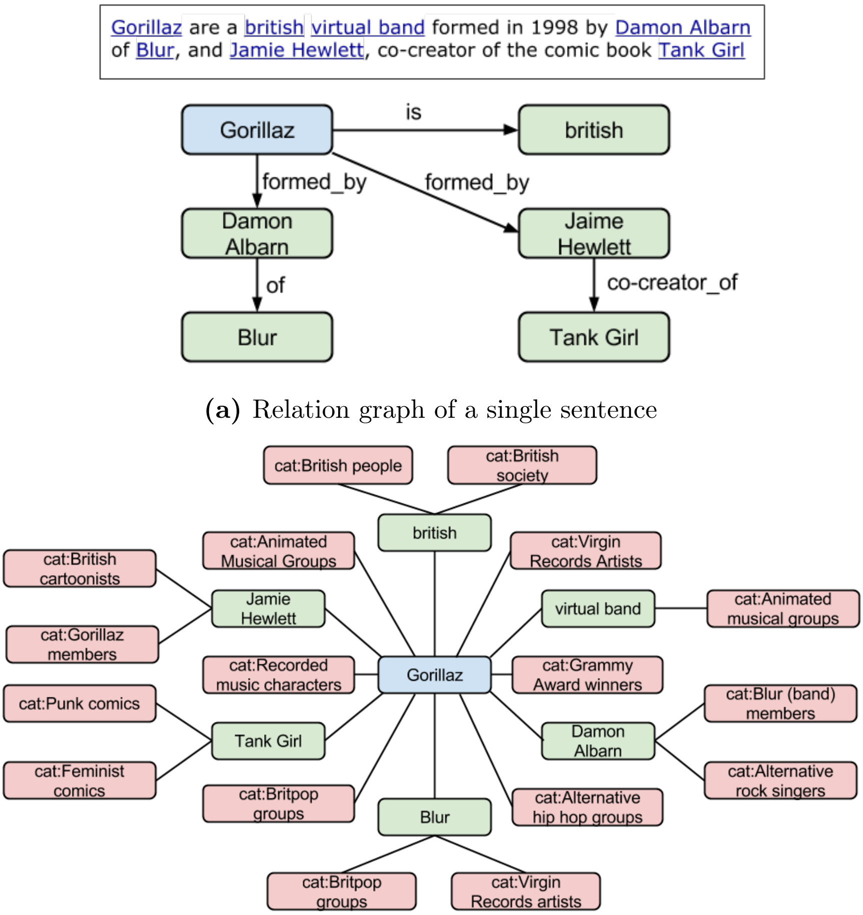
In order to exploit the semantic relations between entities present in artist biographies, we applied the method defined in Chapter 4 for relation extraction in the music domain. The method basically consists of three steps. First, entities are identified in the text by applying entity linking. Second, relations between pairs of entities occurring in the same sentence are identified and filtered by analyzing the structure of the sentence, which is obtained by running a syntactic parser based on the formalism of dependency grammar (Bohnet, 2010). Finally, the identified entities and relations are modeled as a knowledge graph. We apply this methodology to the problem of artist similarity, by creating a graph that connects the entities detected in every artist biography. We call this approach RG (relations graph). Figure 6.2a shows the expected output of this process for a single sentence.

Semantically enriched graph

A second approach is proposed using the same set of linked entities previously identified in the biographies. However, instead of exploiting natural language text, we use semantic information from an external Knowledge Base to enrich the semantics of the linked entities. We use semantic information coming from DBpedia. DBpedia resources are categorized using the DBpedia Ontology among others (e.g., Yago, schema.org) through the `rdfs:type` property. In addition, DBpedia resources are related to Wikipedia categories through the property `dcterms:subject`.

We take advantage of these two properties to build our semantically enriched graph. We consider three types of nodes for this graph: 1) artist entities, obtained by matching the artist name of the biography main theme to DBpedia; 2) named entities detected by the entity linking step; and 3) Wikipedia categories associated to all the previous entities. Edges are then added between artist entities and the named entities detected in their biographies, and between entities and their corresponding Wikipedia categories. For the construction of the graph, we can select all the detected named entities, or we can filter them out according to the information related to their `rdfs:type` property. A set of six types was selected, including *Artist*, *Band*, *Work*, *Album*, *MusicGenre*, and *Person*, which we consider more appropriate to semantically define a musical artist.

From the previous description, we define five variants of this approach. The first variant, which we call AEC (Artists-Entities-Categories), considers all 3 types of nodes along with their relations (as depicted in Figure 6.2b). The second variant, named AE (Artists-Entities) ignores the categories of the en-



(a) Relation graph of a single sentence

(b) Semantically enriched subgraph, variant AEC with h=2 (see MCS in Section 6.2.3)

Figure 6.2: Knowledge graphs.

tities. The third and fourth variant, named AEC-FT and AE-FT, are similar to the first and second variant, respectively, except that the named entities are filtered using the above mentioned list of 6 entity types. Finally, the fifth variant, EC, ignores the artist entities of node type 1.

Sense embeddings

The semantic representation used in this approach is based on SensEmbed (Iacobacci et al., 2015). SensEmbed is a vector space semantic representation of words similar to word2vec (Mikolov et al., 2013b), where each vector represents a BabelNet synset and its lexicalization. Let A be the set of artist biographies in our dataset. Each artist biography $a \in A$ is converted to a set of disambiguated concepts Bfy_a after running Babelfy over it, where each concept has a corresponding SensEmbed vector.

6.2.3. Similarity approaches

SimRank

SimRank is a similarity measure based on an simple graph-theoretic model (Jeh & Widom, 2002). The intuition is that two nodes are similar if similar nodes reference them. In particular we use the definition of bipartite SimRank (Jeh & Widom, 2002). We build a bipartite graph with named entities and their corresponding Wikipedia categories (the EC variant from Section 6.2.2). The similarity between two named entities (say p and q) is computed with the following recursive equation:

$$s(p,q) = \frac{C}{|O(p)||O(q)|} \sum_{i=1}^{|O(p)|} \sum_{j=1}^{|O(q)|} s(O_i(p), O_j(q)) \tag{6.1}$$

where O denotes the out-neighboring nodes of a given node and C is a constant between 0 and 1. For $p = q$, $s(p,q)$ is automatically set up to 1. Once the similarity between all pairs of entities is obtained, we proceed to calculate the similarity between pairs of artists (say a and b) by aggregating the similarities between the named entities identified in their biographies, as shown in the following formula:

$$sim(a,b) = Q(a,b)\frac{1}{N} \sum_{e_a \in a} \sum_{e_b \in b} s(e_a, e_b) \quad \text{if } s(e_a, e_b) \geq 0.1 \tag{6.2}$$

where s denotes the SimRank of entities e_a and e_b and N is the number of (e_a, e_b) pairs with $s(e_a, e_b) \geq 0.1$. This is done to filter out less similar pairs. Finally, $Q(a,b)$ is a normalizing factor that accounts for the pairs of artists with more similar entity pairs than others.

Maximal common subgraph

Maximal common subgraph (MCS) is a common distance measure on graphs. It is based on the maximal common subgraph of two graphs. MCS is a symmetric distance metric, thus $d(A,B) = d(B,A)$. It takes structure as well as content into account. According to Bunke & Shearer (1998), the distance between two non empty graphs G_1 and G_2 is defined as

$$d(G_1, G_2) = 1 - \frac{|mcs(G_1, G_2)|}{max(|G_1|, |G_2|)} \tag{6.3}$$

It can also be seen as a similarity measure s, assuming that $s = 1 - d$, as applied in Lux & Granitzer (2005). To compute this similarity measure we need a graph for each artist, which may be obtained following the approaches defined in Section 6.2.2. An artist graph will include an artist entity node and its neighboring nodes.

Let us formally define the knowledge graph as a multi-relational graph $G = \{t \mid t \in E \times R \times E\}$, where E denotes the set of entities and R indicates the set of properties or relations, namely the edge labels. With E_i^h we denote the set of entities reachable in *at most* h hops from i according to the shortest path in G. For a generic item i we then define its h-hop neighborhood graph $G_i^h = \{t = (e_i, r_j, e_k) \mid t \in E_i^h \times R \times E_i^h\}$ that is the subgraph of G induced by the set of triples involving entities in E_i^h.

Following this approach, we obtain an h-hop item neighborhood graph for each artist. Then, maximal common subgraph is computed between the h-hop item neighborhood graphs of each pair of artists.

Cumulative cosine similarity

For each pair of concepts $c \in \mathrm{Bfy}_a$ and $c' \in \mathrm{Bfy}'_a$ (as defined in Section 6.2.2), we are interested in obtaining the similarity of their closest senses. This is achieved by first deriving the set of associated SensEmbed vectors V_c and $V'_{c'}$ for each pair of concepts c, c', and then optimizing

$$\max_{v_c \in V_c, v'_{c'} \in V'_{c'}} \left(\frac{v_c \times v'_{c'}}{||v_c||\,||v'_{c'}||} \right) \tag{6.4}$$

i.e., computing cosine similarity between all possible senses (each sense represented as a vector) in an all-against-all fashion and keeping the highest scoring similarity score for each pair. Finally, the semantic similarity between two artist biographies is simply the average among all the cosine similarities between each concept pair.

6.2.4. Experiments

To evaluate the accuracy of the proposed approaches we designed an experimental evaluation over two datasets. The first dataset contains 2,336 artists and it is evaluated using the list of similar artists provided by the Last.fm API as a ground truth. The second dataset contains 188 artists, and it is evaluated against user similarity judgments from the MIREX Audio Music Similarity and Retrieval task. Apart from the defined approaches, a pure text-based approach for document similarity is added to act as a baseline for the obtained results.

Last.fm dataset

A dataset of 2,336 artist biographies was gathered from Last.fm. The artists in this dataset share a set of restrictions. Their biography has at least 500 characters and is written in English. All of the artists have a correspondent Wikipedia page, and we have been able to map it automatically, obtaining the DBpedia URI of every artist. For every artist, we queried the getSimilar method of the Last.fm API and obtained an ordered list of similar artists. Every artist in the dataset fulfills the requirement of having at least 10 similar artists within the dataset. We used these lists of similar artists as the ground truth for our evaluation.

MIREX dataset

To build this dataset, the gathered artists from Last.fm were mapped to the MIREX Audio Music Similarity task dataset. The AMS dataset (7,000 songs from 602 unique artists) contains human judgments of song similarity. According to Schedl et al. (2013), the similarity between two artists can be roughly estimated as the average similarity between their songs. We used the same approach in Schedl et al. (2013), that is, two artists were considered similar if the average similarity score between their songs was at least 25 (on a fine scale between 0 and 100).

After the mapping, we obtained an overlap of 268 artists. As we want to evaluate Top-10 similarity, every artist in the ground truth dataset should have information of at least 10 similar artists. However, not every artist in the MIREX evaluation dataset fulfills this requirement. Therefore, after removing the artists with less than 10 similars, we obtained a final dataset of 188 artists, and used it for the evaluation.

Baseline

In order to assess the goodness of our approaches, we need a baseline approach. The baseline used in this section is a classic Vector Space Model (VSM) approach used in many Information Retrieval systems. A text document is represented as a vector of word frequencies (after removing English stopwords and words with less than 2 characters), and a matrix is formed by aggregating all the vectors. The word frequencies in the matrix are then re-weighted using *tf-idf*, and finally Latent Semantic Analysis (LSA) (Deerwester et al., 1990) is used to produce a vector of concepts for each document. The similarity between two documents can be obtained by using a cosine similarity over their corresponding vectors.

Evaluated approaches

From all possible combinations of knowledge representations, similarity measures and parameters, we selected a set of 10 different approach variants. The prefixes AEC, RG, EC, and AE refer to the graph representations (see Section 6.2.2). SE refers to the sense embeddings approach, and LSA to the latent semantic analysis baseline approach. When these prefixes are followed by FT, it means that the entities in the graph have been filtered by type. The second term in the name refers to the similarity measure. MCS refers to maximal common subgraph, and SimRank and Cosine to SimRank and cumulative cosine similarity measures. A number indicating the number of h-hops of the neighborhood subgraph follows MCS approaches.

Evaluation measures

To measure the accuracy of the artist similarity we adopt two standard performance metrics such as Precision@N, and nDCG@N (normalized discounted cumulative gain). Precision@N is computed as the number of relevant items (i.e., true positives) among the top-N items divided by N, when compared to a ground truth. Precision considers only the relevance of items, whilst nDCG takes into account both relevance and rank position. Denoting with s_{ak} the relevance of the item in position k in the Top-N list for the artist a, then nDCG@N for a can be defined as:

$$\text{nDCG@N} = \frac{1}{\text{IDCG@N}} \sum_{k=1}^{N} \frac{2^{s_{ak}} - 1}{\log_2(1+k)} \tag{6.5}$$

where IDCG@N indicates the score obtained by an ideal or perfect Top-N ranking and acts as a normalization factor. We run our experiments for $N = 5$ and $N = 10$.

6.2.5. Results and discussion

We evaluated all the approach variants described in Section 6.2.4 on the MIREX dataset, but only a subset of them on the Last.fm dataset, due to the high computational cost of some of the approaches.

Table 6.1 shows the Precision@N and nDCG@N results of the evaluated approaches using the MIREX dataset, while Table 6.2 shows the same results for the Last.fm dataset. We obtained very similar results in both datasets. The approach that gets best performance for every metric, dataset and value of N is the combination of the Artists-Entities-Categories graph filtered by types, with the maximal common subgraph similarity measure using a value of $h = 1$ for obtaining the h-hop item neighborhood graphs.

	Precision@N		nDCG@N	
Approach variants	N=5	N=10	N=5	N=10
LSA	0.100	0.169	0.496	0.526
RG MCS 1-hop	0.059	0.087	0.465	0.476
RG MCS 2-hop	0.056	0.101	0.433	0.468
AE MCS	0.106	0.178	0.503	0.517
AE-FT MCS	0.123	0.183	0.552	0.562
AEC MCS 1-hop	0.120	0.209	0.573	0.562
AEC MCS 2-hop	0.086	0.160	0.550	0.539
AEC-FT MCS 1-hop	**0.140**	**0.218**	**0.588**	**0.578**
AEC-FT MCS 2-hop	0.100	0.160	0.527	0.534
EC SimRank	0.097	0.171	0.509	0.534
SE Cosine	0.095	0.163	0.454	0.484

Table 6.1: Precision and normalized discounted cumulative gain for Top-N artist similarity in the MIREX dataset (N={5, 10}). LSA stands for Latent Semantic Analysis, RG for Relation Graph, SE for Sense Embeddings, and AE, AEC and EC represent the semantically enriched graphs with Artists-Entities, Artist-Entities-Categories, and Entities-Categories nodes, respectively. As for the similarity approaches, MCS stands for Maximum Common Subgraph.

	Precision@N		nDCG@N	
Approach variants	N=5	N=10	N=5	N=10
LSA	0.090	0.088	0.233	0.269
RG MCS 1-hop	0.055	0.083	0.126	0.149
AE MCS	0.124	0.200	0.184	0.216
AE-FT MCS	0.136	0.201	0.224	0.260
AEC MCS 1-hop	0.152	0.224	0.277	0.297
AEC-FT MCS 1-hop	**0.160**	**0.242**	**0.288**	**0.317**

Table 6.2: Precision and normalized discounted cumulative gain for Top-N artist similarity in the Last.fm dataset (N={5, 10})

	Genres							
Approach variants	Blues	Country	Edance	Jazz	Metal	Rap	Rock	Overall
Ground Truth	5.78	5.46	6.88	7.04	7.10	8.68	5.17	6.53
LSA	4.43	4.12	3.80	4.64	5.79	5.08	4.74	4.69
RG MCS 1-hop	2.63	3.50	1.50	2.95	4.00	2.54	1.70	2.68
RG MCS 2-hop	4.14	4.92	1.69	2.80	3.78	3.06	2.77	3.27
AE MCS	5.52	5.15	4.36	7.00	4.34	5.36	4.46	5.11
AE-FT MCS	5.43	6.12	4.16	6.20	6.32	5.36	3.77	5.26
AEC MCS 1-hop	**7.22**	5.92	5.24	7.12	5.48	6.92	4.86	6.02
AEC MCS 2-hop	4.22	3.69	4.56	6.20	4.55	4.64	4.09	4.54
AEC-FT MCS 1-hop	6.91	**6.80**	**6.04**	**7.60**	**6.79**	**7.12**	**5.37**	**6.59**
AEC-FT MCS 2-hop	4.09	4.36	5.56	6.72	4.39	4.16	3.77	4.67
EC SimRank	6.74	5.38	3.16	6.40	4.59	4.44	3.80	4.85
SE Cosine	3.39	5.50	5.32	5.16	4.31	5.36	4.31	4.75

Table 6.3: Average genre distribution of the top-10 similar artists using the MIREX dataset. In other words, on average, how many of the top-10 similar artists are from the same genre as the query artist.

Furthermore, given that the MIREX AMS dataset also provides genre data, we analyzed the distribution of genres in the top-10 similar artists for each artist, and averaged them by genres. The idea is that an artist's most similar artists should be from the same genre as the seed artist. Table 6.3 presents the results. Again, the best results are obtained with the approach that combines the Artists-Entities-Categories graph filtered by types, with the maximal common subgraph similarity measure using a value of $h = 1$ for the h-hop item neighborhood graphs.

We extract some insights from these results. First, semantic approaches are able to improve pure text-based approaches. Second, using knowledge from an external Knowledge Base provides better results than exploiting the relations inside the text. Third, using a similarity measure that exploits the structure and content of a graph, such as maximal common subgraph, overcomes other similarity measures based on semantic similarity among entity mentions in document pairs.

6.3. Music genre classification

In this section we describe a method to enrich music related documents with semantic and affective information, which are in turn exploited in a classification problem. To measure the impact of the proposed approach, we perform an experiment on music genre classification. The goal of the experiment, given an album review, is to predict the music genre it belongs to. Different combinations of features are explored, including semantic, sentimental, and acoustic. Experiments are performed on a subset of the Multimodal Album Reviews Dataset (MARD) described in Section 5.3.1.

6.3.1. Dataset description

We first enriched the MARD dataset (see Section 5.3.1) with acoustic information. To this end, we retrieved songs' audio descriptors from AcousticBrainz[23], a database of music and audio descriptors, computed from audio recordings via state-of-the-art Music Information Retrieval algorithms (Porter et al., 2015). From the 28,053 albums initially mapped to MusicBrainz, a total of 8,683 albums were further linked to AcousticBrainz, which encompasses 65,786 songs. Starting from this enriched version of the MARD dataset, our purpose is to create a subset suitable for music genre classification, including 100 albums per genre class. We enforced these albums to be authored by different artists, and that review texts and audio descriptors of their songs are available in MARD. Then, for every album, we selected audio descriptors of the first song of each album as a representative sample of the album. From the original 16 genres, 3 of them did not have enough instances complying with these prerequisites (Reggae, Blues and Gospel). This results in a classification dataset composed of 1,300 albums, divided in 13 different genres. The review texts of each album were aggregated and then truncated around 1,000 characters (this length slightly varies from one album to another, as we wanted to keep the complete text of every individual review). We limited text length to avoid any bias towards popular albums.

6.3.2. Linguistic processing

Given the set of documents, two linguistic processes are applied. First, following the work of Dong et al. (2013, 2014), we apply an aspect-based sentiment analysis technique to extract opinionated features from each document (see Section 5.3.2). Second, we apply an entity linking process to each document. In this case, entity linking was performed taking advantage of TagMe (Ferragina & Scaiella, 2012) and ELVIS (see Section 3.3). TagMe provides for each detected entity its Wikipedia page ID, whereas ELVIS enriches the obtained entities with DBpedia URIs, DBpedia Ontology types, and Wikipedia categories.

6.3.3. Features

Textual Surface Features

We use a standard Vector Space Model (VSM) representation, where documents are represented as bag-of-words (BoW) after tokenizing and stopword removal. All words and bigrams (sequences of two words) are weighted using *tf-idf*.

[23] `https://acousticbrainz.org/`

Semantic features

We enrich the documents with semantic information thanks to the application of entity linking. Specifically, for each named entity disambiguated with TagMe, its Wikipedia ID and its associated Wikipedia categories are added at the end of the document as extra words. Wikipedia categories are hierarchically organized, so we enrich the documents by adding one level more of broader categories by querying DBpedia, using the *skos:broader* property. Then, a feature vector is obtained after applying a VSM approach with *tf-idf* weighting to the semantically enriched texts.

Sentiment features

Based on the aspects and associated polarity extracted with the applied aspect-based sentiment analysis technique (see Section 6.3.2), we implement a set of sentiment features following Montero et al. (2014):

- Positive to All Emotion Ratio: fraction of all sentimental features that are identified as positive (sentiment score greater than 0).

- Document Emotion Ratio: fraction of total words with sentiments attached. This feature captures the degree of affectivity of a document regardless of its polarity.

- Emotion Strength: This document-level feature is computed by averaging sentiment scores over all aspects in the document.

- F-Score[24]: This feature has proven to be useful for describing the contextuality/formality of language. It takes into consideration the presence of *a priori* “descriptive” POS tags (nouns and adjectives), as opposed to “action” ones such as verbs or adverbs.

Acoustic features

Acoustic features are obtained from AcousticBrainz (Porter et al., 2015). They are computed using Essentia (Bogdanov et al., 2013b). These encompass loudness, dynamics, spectral shape of the signal, as well as additional descriptors such as time-domain, rhythm, and tone.

6.3.4. Baseline approaches

Two baseline systems are implemented. First, we implement the text-based approach described in Hu et al. (2005) for music review genre classification.

[24] Not to be confused with the evaluation metric.

	BoW	BoW+SEM	BoW+SENT
Linear SVM	**0.629**	**0.691**	**0.634**
Ridge Classifier	0.627	0.689	0.61
Random Forest	0.537	0.6	0.521

Table 6.4: Accuracy of the different classifiers.

In this work, a Naïve Bayes classifier is trained on a collection of 1,000 review texts, and after preprocessing (tokenization and stemming), BoW features based on document frequencies are generated. The second baseline is computed using the AcousticBrainz framework for song classification (Porter et al., 2015). Here, genre classification is computed using multi-class support vector machines (SVMs) with a one-vs.-one voting strategy. The classifier is trained with the set of low-level features present in AcousticBrainz.

6.3.5. Experiments

We tested several classifiers typically used for text classification, namely Linear SVM, Ridge Classifier, and Nearest Centroid, using the implementations provided by the scikit-learn library[25]. Among them, Linear SVM has shown better performance when combining different feature sets (see Table 6.4). Therefore, we trained a Linear SVM classifier with L2 penalty over different subsets of the features described in Section 6.3.3, which are combined via linear aggregation. Specifically, we combine the different feature sets into five systems, namely BoW (text features only), BoW+SEM (text and semantic features without broader categories), BoW+SEMb (text and semantic features with broader categories), BoW+SENT (text and sentiment features), and BoW+SEM+SENT (text, semantic, and sentiment features). In this way, we aim at understanding the extent to which sentiment and semantic features (and their interaction) may contribute to the review genre classification task. Note that this section is focused on the influence of textual features in genre classification, and classification based on acoustic features is simply used as a baseline for comparison. A proper combination of acoustic and textual features in text classification is a challenging problem and would require a deeper study, as the one provided in Chapter 9. The dataset is split 80-20% for training and testing, and accuracy values are obtained after 5-fold cross validation.

6.3.6. Results and discussion

Accuracy results of the two baseline approaches introduced in Section 6.3.4 along with our approach variants are shown in Figure 6.3. At first sight, we may

[25] `http://scikit-learn.org/`

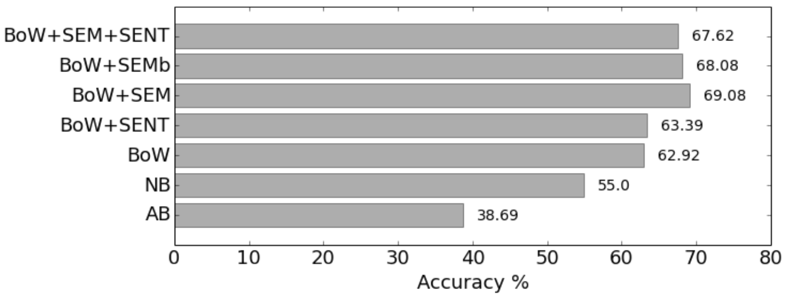


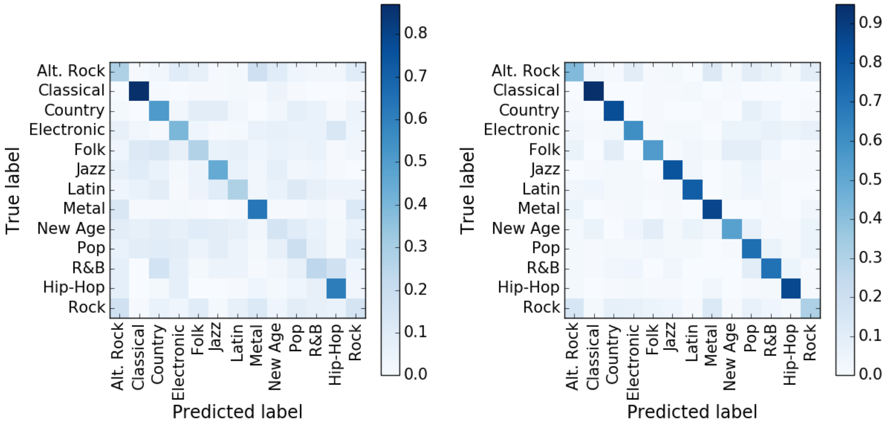
Figure 6.3: Percentage of accuracy of the different approaches. AB refers to the AcousticBrainz framework. NB refers to the method based on Naïve Bayes from Hu et al. (2005).

conclude that sentiment features contribute to slightly outperforming purely text-based approaches. This result implies that affective language present in a music review is not a salient feature for genre classification (at least with the technology we applied), although it certainly helps. On the contrary, semantic features clearly boost pure text-based features, achieving 69.08% of accuracy. The inclusion of broader categories does not improve the results in the semantic approach. The combination of semantic and sentiment features improves the BoW approach, but the achieved accuracy is slightly lower than using semantic features only.

Let us review the results obtained with baseline systems. The Naïve Bayes approach in Hu et al. (2005) is reported to achieve an accuracy of 78%, while in our results it is below 55%. The difference in accuracy may be due to the substantial difference in length of the review texts. In Hu et al. (2005), review texts were at least 3,000 characters long, much larger that ours. Moreover, the addition of a distinction between Classic Rock and Alternative Rock is penalizing our results. As for the acoustic-based approach, although the obtained accuracy may seem low, it is in fact a good result for purely audio-based genre classification, given the high number of classes and the absence of artist bias in the dataset (Bogdanov et al., 2016). Finally, we refer to Figure 6.4 to highlight the fact that the text-based approach clearly outperforms the acoustic-based classifier, although in general both show a similar behavior across genres. Also, note the low accuracy for both Classic Rock and Alternative Rock, which suggests that their difference is subtle enough for making it a hard problem for automatic classification.

6.4. Conclusion

In this chapter we presented several methodologies that exploit semantic technologies for computing artist similarity and music genre classification. Par-



(a) AcousticBrainz acoustic-based.

(b) BoW+SEM text-based.

Figure 6.4: Confusion matrices.

ticularly, we focused on the use of entity linking as a medium to enrich the information present in musical documents. Results in both tasks show that the addition of semantic information via entity linking clearly yields performance improvements.

Different methods to embed this semantic information have been proposed, from knowledge graphs to vector space models. In the case of artist similarity, the proposed methodology is divided in three main steps: First, named entity mentions are identified in the text and linked to a Knowledge Base. Then, these entity mentions are used to construct a semantically motivated knowledge representation. Finally a similarity function is defined on top of the knowledge representation to compute the similarity between artists. For each one of these steps we explored several approaches, and evaluated them against a small dataset of 188 artist biographies, and a larger one of 2,336, both obtained from Last.fm. Results showed that the combination of semantically enriched graphs via entity linking, and a maximal common subgraph similarity measure clearly outperforms a baseline approach that exploits word co–occurrences and latent factors.

In the case of music genre classification, a multimodal dataset of album customer reviews combining text, metadata, and acoustic features gathered from Amazon, MusicBrainz, and AcousticBrainz respectively was used. Customer review texts were further enriched with data from Wikipedia along with polarity information derived from aspect-based sentiment analysis. Based on this information, a classifier is trained using different combinations of features. A comparative evaluation of features suggests that a combination of text and semantic information has higher discriminative power, outperforming competing systems in terms of accuracy.

In the light of these results on both tasks, the following conclusions can be drawn: The described semantic enrichment approaches outperform pure text-based approaches thanks to the enrichment of texts with external knowledge, boosting the performance on both tasks. In addition, reducing noise by filtering linked entities by type is a rewarding step that contributes to an improved performance.

CHAPTER 7

Sound and Music Recommendation with Knowledge Graphs

7.1. Introduction

In this chapter we tackle the problem of computing sound and music recommendations following a hybrid approach that leverages semantic content features extracted from textual descriptions and collaborative features from implicit user feedback. The approach we propose to recommend musical items consists mainly of two parts: (i) the enrichment of original data attached to items by linkage to knowledge repositories, (ii) the effective exploitation of the graph-based nature of such data for computing the recommendations.

The enrichment of data consists in using entity linking techniques for extracting semantic entities from item textual descriptions and linking them to external knowledge bases such as WordNet (Miller, 1995) and DBpedia (Bizer et al., 2009) for gathering additional knowledge. All those different information are eventually merged together and represented by means of a new knowledge graph (KG), following a similar approach to the one described in Section 6.2.2. This latter graph is thus exploited together with collaborative information from implicit feedback for computing the recommendations. Two graph feature mappings are defined to leverage the new knowledge graph and obtain expressive feature representations. All different features are combined together in a feature combination hybrid schema (Burke, 2002) and used to feed a content-based recommender. An extensive experimental evaluation was carried out on two different datasets—one related to sounds and other to songs—to evaluate the recommendation quality in terms of accuracy, novelty, and aggregated diversity.

In this chapter, we deal with two slightly different problems in the music ecosystem. We address the songs recommendation problem and that of recommending sounds to users in online sound sharing platforms. The two tasks addresses two separate categories of users in the music domain: on the one hand, we have music consumers (songs and artists recommendation); on the other hand, we have music producers (sounds recommendation).

Music recommendation has received a lot of attention in the last decade (Celma, 2010; Knees & Schedl, 2013). As a matter of fact, the discovery of new songs and artists is a task that the music consumers of a Web radio or of a music store are naturally led to perform daily. Hence, helping them by recommending the best choices results in immediate impact also in industrial and commercial scenarios.

Differently from the previous case, recommendation of sounds has received scant attention even though it may be of interest in many scenarios of music creation. As an example, we may consider producers of electronic music that typically downloads and use sound samples. They might be interested in the recommendation of relevant sounds downloaded by users with similar tastes or similar (not equal) to those they previously used in their musical compositions. Likely, they are also looking not just for popular sounds, as they want their production to be unique. To this end, we first centered our study in Freesound[26], one of the most popular sites on the Web for sharing audio clips, accounting more than 6 million registered users and about 350k uploaded sounds, which are described in terms of textual descriptions and tags. In Freesound, different kind of users may be observed (Font et al., 2012) (e.g., music producers, composers, sound designers, soundscape enthusiasts), and also different types of sounds (e.g., sound samples, field recordings, soundscapes, loops). We have the intuition that collaborative features may help in the personalization of the recommendations, whilst the introduction of semantic features may lead to a better exploitation of less popular items. To evaluate this hypothesis, a dataset composed of sound descriptions and historical data about user's download behavior was collected.

To demonstrate the suitability of the proposed methodology for both types of musical users (producers and consumers), a music recommendation experiment was also performed. To this end, a dataset of songs, which combines tags and textual descriptions with users' implicit feedback was created. This dataset aggregates information gathered from Songfacts[27] and Last.fm[28]. Songfacts is an online database that collects, stores and provides facts, stories and trivia about songs, whilst in Last.fm a detailed profile of each user's musical taste is built by recording details of the tracks the user listens to.

[26] `http://freesound.org`

[27] `http://songfacts.com`

[28] `http://last.fm`

The evaluation performed on both datasets showed that the semantic expansion of the original data combined with user collaborative features allows the system to enhance recommendation quality especially in terms of aggregated diversity and novelty while keeping high performance in terms of accuracy.

The reminder of the chapter is structured as follows. The next section introduces the basic technologies used to build the knowledge graph at the basis of our recommendation system. Section 7.2 describes the problem and the semantic expansion applied to the initial data. Then, Section 7.3 defines the adopted recommendation approach while in Section 7.4 we explain the experimental evaluation and discusses the obtained results. Finally, Section 7.5 concludes the chapter.

7.2. Knowledge enrichment via entity linking

In order to add more semantics to the description of musical items, we exploit contextual information, i.e., tags and text descriptions, and then use this information to create a knowledge graph. Several approaches have been developed to enrich tags with semantics (Garcia-Silva et al., 2012). We follow an ontology-based approach, enriching both tags and keywords extracted from textual descriptions by associating them with relevant entities defined in online knowledge repositories. The first step in this direction is to link and disambiguate tags and keywords to Linked Data resources. For this purpose we adopted Babelfy (Moro et al., 2014b). We selected this tool as it is able not only to disambiguate named entities, but also concepts. Our intuition here is that the disambiguation of concepts used to describe sounds may be useful to enrich their descriptions. Babelfy output is further enriched with ELVIS (see Section 3.3). Thus, for every mapped and disambiguated text fragment, we obtain the related DBpedia and/or WordNet *synset*. For DBpedia entities we also obtain the associated Wikipedia categories.

To build our semantically enriched graph, the entity linking tool is firstly run on both tags and keywords of every item. Identified named entities are linked to DBpedia resources, whilst disambiguated words are linked to WordNet *synsets*. Every musical item is added to the graph. Then, for each item, text spans from its description identified as entities are added to the graph, and connected to the item. We refer to these text spans as *keywords*. Keywords are in turn connected with their corresponding URIs, whether they are a DBpedia resource or a WordNet *synset*. Subsequently, we use both WordNet and DBpedia to semantically expand the entities added to the graph after the entity linking phase. Each synset obained from the linking is further expanded considering other concepts in the WordNet hierarchy of sysnsets by following the hypernymy[29] relations. From the WordNet hierarchy we extract up to 2-hop hypernyms starting from the mapped synset. We empirically selected the maximum distance of two hops because we wanted to avoid too broad generalization of the original concept. For the same reason we discard those hypernyms less than six hops away from the root of the WordNet hierarchy. Regarding DBpedia, our entity linking pipeline returns the URI of the linked entity and a set of related Wikipedia categories. In DBpedia, resources are related to categories through the property `dcterms:subject`. Those categories are in turn organized in a taxonomy. In particular, more specific categories are related to more generic ones by means of the `skos:broader` property. Thus, for each category retrieved, all the direct broader categories were gathered and added to our knowledge graph. To avoid too broad or unrelated categories, only one level of broader categories was considered.

To show an example of entity linking performed by Babelfy we use the sound `prac-snare2.wav`[30] from Freesound. The description associated to this sound is *"standard snare sample. lower/mid tuning on the head"* and tags are *drums*, *percussion*, *snare*. Babelfy was able to detect and link most of the entities. Just to describe a few of them, the word *sample* from the description was linked to the DBpedia entity `Sampling_(music)`, the tag *percussion* was mapped to the DBpedia entity `Rythm_section`, the tag *snare* was linked to the WordNet concept `snare_drum.n.01` and DBpedia entity `Snare_drum`. As shown in Figure 7.1, DBpedia entities and WordNet synsets are then further enriched with their related categories and hypernyms. Following the Linked Data principles[31], we reused classes and properties from external vocabularies. The final knowledge graph after the entity linking and expansion process contains four main classes: `wordnet:Synset`, `Entity`, `Tag`, and `skos:Concept`, and seven relations: `hasTag`, `hasKeyword`, `wordnet:synset_member`, `dcterms:relation`, `dcterms:subject`, `skos:broader`, and `wordnet:hypernym`[32].

7.3. Recommendation approach

As aforementioned, we adopted a hybrid recommendation approach to leverage both collaborative information coming from the user's community and content information coming from the knowledge graph. According to the taxonomy of hybrid recommender systems presented in Burke (2002), we developed a hybrid feature combination recommender system. In our case, hybridization is not based on the combination of different recommendation components but instead on the combination of different data sources. Specifically, collaborative

[29] Hypernymy models generalization relations between synsets.

[30] `http://www.freesound.org/people/TicTacShutUp/sounds/439/`

[31] `http://www.w3.org/DesignIssues/LinkedData.html`

[32] All the prefix we use here are the ones available via the `http://prefix.cc` service

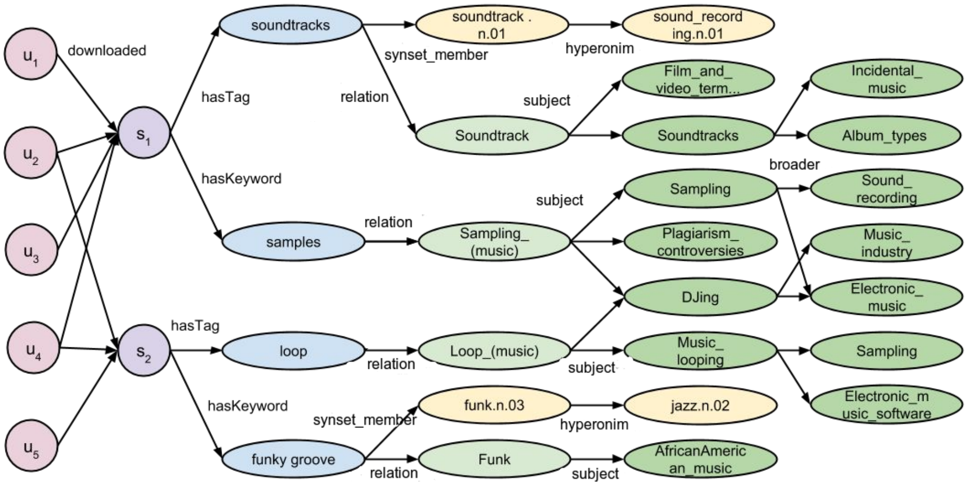


Figure 7.1: Portion of the final knowledge graph enriched with WordNet and DBpedia

information is treated as additional features of the content feature space and a content-based technique is used over this augmented space. Therefore, we build feature item representations by considering the item graph-based descriptions represented in the knowledge graph and enrich such feature vectors with collaborative features. Subsequently, we use such data to feed a content-based recommendation engine.

Content-based recommendations are typically computed by learning a function that, for each item in the system, predicts the relevance of such item for the user. A *top-N* item recommendation problem in a standard content-based setting is mainly split into two different tasks: (i) given a collection of items for which past user's preferences are available, learn a regression or classification model to predict the relevance associated to items unknown to the user; (ii) according to the obtained scores, recommend the most relevant items to the user. Past user's preferences can be obtained from either explicit or implicit feedback. As for Freesound, we considered as an implicit positive feedback the "download data". The rationale behind our choice is that if a user downloads a sound it is reasonable to assume that she likes it even without an explicit rating, as the system lets users listen to sounds before downloading. Also the Last.fm dataset used in the experimental evaluation contains user song listening actions, which is another form of implicit feedback. Thus, in the following we will refer to the problem of computing recommendations from implicit feedback data. Following the notation introduced by Rendle et al. (2009) for implicit feedback scenarios, let M be the matrix of implicit feedback, where $m_{ui} = 1$ if item i was downloaded from user u, 0 otherwise. Starting from M we define

$I_u^+ = \{i \in I | m_{ui} = 1\}$ as the set of relevant items for u. The main problem with implicit feedback is that they reflect only positive user preferences. On the contrary, the system cannot infer anything about what the user dislikes. The unobserved data are a mixture of actually negative and missing values (Rendle et al., 2009), but the system does not have any information for discriminating between them. Then, learning a predictive model from such unary data becomes infeasible because there are no negative examples. To overcome this issue for each user we select a portion of unobserved items $I_u^- \subset (I \setminus I_u^+)$ to be used as negative data points in the training of the model. In Ostuni et al. (2013), the authors show that choosing $|I_u^-| = 2 \cdot |I_u^+|$ does not affect accuracy results. The unobserved items are exactly the items that have to be ranked. The ultimate goal of the system is to rank in the *top-N* positions items likely to be relevant for the user.

Given the generic user u, let T_u be the training set for u defined as:

$$T_u = \{\langle x_i, m_{ui}\rangle | i \in (I_u^+ \cup I_u^-)\}$$

where $x_i \in \mathbb{R}^D$ is the feature vector associated to the item i and let TS_u be the test set defined as:

$$TS_u = \{\langle x_i, s_{ui}^*\rangle | i \in (I \setminus I_u^+)\}$$

The two tasks for the *top-N* recommendation problem, in our setting, consist then of: (i) learning a function $f_u : \mathbb{R}^D \to \mathbb{R}$ from the training data T_u which assigns a relevance score to the items in I; (ii) using such function to predict the unknown score m_{ui}^* in the test set TS_u, to rank them and recommend the *top-N*.

Given that items are represented as entities in a knowledge graph we are particularly interested in those machine learning methods that are appropriate for dealing with objects structured as graphs. There are two main ways of learning with structured objects. The first is to use *Kernel Methods* (Shawe-Taylor & Cristianini, 2004). Given two input objects i and j, defined in an input domain space D, the basic idea behind Kernel Methods is to construct a kernel function $k : D \times D \to \mathbb{R}$, that can be informally seen as a similarity measure between i and j. This function must satisfy $k(i, j) = \langle \phi(i), \phi(j)\rangle$ for all $i, j \in D$, where $\phi : D \to F$ is a mapping function to a inner product feature space F. Then, the classification or regression task involves linear convex methods based exclusively on inner products computed using the kernel in the embedding feature space. The alternative way is to explicitly compute the *explicit feature mapping* $\phi(i)$ and to directly use linear methods in the related space. By transforming the graph domain into a vector domain any traditional learning algorithm working on feature vectors can be applied.

While kernel methods have been widely applied to solve different tasks, their usage becomes prohibitive when dealing with large datasets. In addition, when

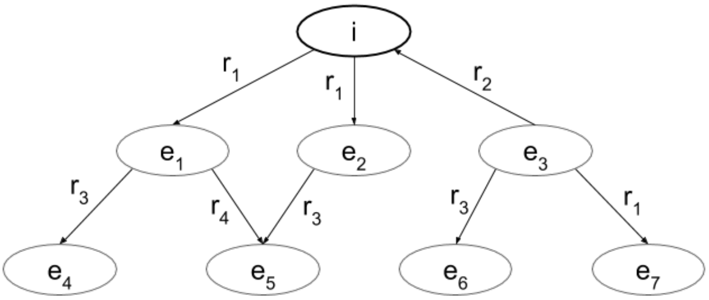


Figure 7.2: An example of 3-hop item neighborhood graph for the item i.

the input data lie in a high-dimensional space, linear kernels have performances comparable to more complex non-linear ones. Due to the high volume of users we deal with in our Freesound dataset (see Section 7.4), we focused on learning methods that are computationally efficient. For this reason we adopted the approach of computing the explicit feature mapping of the item graphs and use linear methods to learn the user model. Specifically, we use the Linear Support Vector Regression (Ho & Lin, 2012) algorithm. Regarding the explicit feature mapping computation we define two sparse high-dimensional feature maps: the one based on entities, the other on paths that we call *entity-based item neighborhood mapping* and *path-based item neighborhood mapping*, respectively. In the following we formalize the computation of such graph embeddings.

7.3.1. Explicit feature mappings for graph-based item representations

We follow the formal definition of a multi-relational knowledge graph $G = \{t \mid t \in E \times R \times E\}$ stated in Section 6.2.3, where E denotes the set of entities and R indicates the set of properties or relations. Moreover, we have $I \subseteq E$ since we consider artist items as a particular type of entities. For a generic item i, its h-hop item neighborhood graph is defined as $G_i^h = \{t = (e_i, r_j, e_k) \mid t \in E_i^h \times R \times E_i^h\}$. In Figure 7.2, an example of a 2-hop item neighborhood graph for item i, namely G_i^2, is shown. We see that, if we consider the shortest path, all the entities are no more than 2 hops distant from i.

To clarify the definition and computation of G_i^h and E_i^h for item i, we show their computation with reference to the example shown in Figure 7.2:
$G_i^1 = \{(i, r_1, e_1), (i, r_1, e_2), (e_3, r_2, i)\}$
$G_i^2 = G_i^1 \bigcup \{(e_1, r_3, e_4), (e_1, r_3, e_5), (e_2, r_4, e_5), (e_3, r_6, e_6), (e_3, r_1, e_7)\}$
$E_i^1 = \{e_1, e_2, e_3\}$

$E_i^2 = E_i^1 \bigcup \{e_4, e_5, e_6, e_7\}$

Starting from those item graph-based representations we define the two different feature mappings which are described in what follows.

Entity-based item neighborhood mapping

In this mapping each feature refers to an entity in E and the corresponding score represents the weight associated to that entity in G_i^h. The resulting feature vector $\phi_E(G_i^h)$ is:

$$\phi_E(G_i^h) = (w_{i,e_1}, w_{i,e_2}, ...w_{i,e_m}, ..., w_{i,e_t})$$

where the weight associated to the generic entity e_m is computed as follows:

$$w_{i,e_m} = \sum_{l=1}^{h} \alpha_l \cdot c_{l,e_m}$$

with

$$\alpha_l = \frac{1}{1 + \log(l)}$$

and

$$c_{l,e_m} = |\{(e_n, r, e_m) \mid e_n \in \widehat{E}_i^{l-1} \ \wedge \ e_m \in \widehat{E}_i^l\} \bigcup \\ \{(e_m, r, e_n) \mid e_m \in \widehat{E}_i^l \ \wedge \ e_n \in \widehat{E}_i^{l-1}\}|$$

where $\widehat{E}_i^l = E_i^l \setminus E_i^{l-1}$ is the set of entities *exactly* l hops far from i.
In particular, c_{l,e_m} corresponds to the number of triples connecting e_m to entities in the previous hop $(l-1)$, whether e_m appears either as subject or object of the triple. In other words, c_{l,e_m} can be seen as the *occurrence* of the entity e_m in the item neighborhood at distance l. The more the entity e_m is connected to neighboring entities of i, the more it is descriptive of i. α_l can be seen as a decay factor depending on the distance l from the item i, whose aim is to incrementally penalize farther entities from the item. It allows us to take into account the *locality* of those entities in the graph neighborhood. The closer an entity e_m to the item i, the stronger its relatedness to it. We use a logarithmic decay.

With reference to example showed in Figure 7.2, the c_{l,e_m} values are computed as follows: $c_{1,e_1} = 1$, $c_{1,e_2} = 1$, $c_{1,e_3} = 1$, $c_{2,e_4} = 1$, $c_{2,e_5} = 2$, $c_{2,e_6} = 1$, $c_{2,e_7} = 1$.

Path-based item neighborhood mapping

Differently from the previous case, in this mapping we represent a feature as a sequence of nodes in G. Given two entities e_1 and e_n, we consider the sequence of nodes $e_1 \cdot e_2 \cdot \ldots \cdot e_{n-1} \cdot e_n$ met while traversing the graph to go from e_1 to e_n and we refer to such sequence as a *path*. In this mapping, a feature is then represented by a path. In particular, in this mapping each feature refers to several variants of paths rooted in the item node. We first collect all the paths rooted in i which can be indicated as sequence of entities $i \cdot e_1 \cdot e_2 \cdot \ldots \cdot e_{n-1} \cdot e_n$. Then, starting from those paths we define various features considering sub-paths of the original paths. Specifically we form sub-paths composed by only those entities progressively farther from the item. Considering the path given above we build the following features: $e_1 \cdot e_2 \cdot \ldots \cdot e_{n-1} \cdot e_n$, $e_2 \cdot \ldots \cdot e_{n-1} \cdot e_n$, ..., $e_{n-1} \cdot e_n$, e_n. The rationale behind this choice is that it allows to explicitly represent substructures shared between items with no overlapping in their immediate neighborhoods but somehow connected at further distance. Items connected to the same entities have same common structures because both closer and further entities are shared. Items connected to different entities which are however linked directly or at a farther distance to same entities share less or none sub-paths depending on how much far the common entities are, if any.

More formally, let P_i be the set of paths rooted in i and P_i^* be the list of all possible sub-paths extracted from them. We use $p_m(i)$ and $p_m^*(i)$ to refer to the mth elements in P_i and P_i^*, respectively. Then, the feature mapping for item i is:

$$\phi_P(G_i^h) = (w_{i,p_1^*}, w_{i,p_2^*}, ...w_{i,p_m^*}, ..., w_{i,p_t^*})$$

where each $w_{i,p_m^*}$is computed as:

$$w_{i,p_m^*} = \frac{\#p_m^*(i)}{|p_m| - |p_m^*|}$$

where $|p_m|$ indicates the length of path p_m and $\#p_m^*(i)$ the occurrence of $p_m^*(i)$ in P_i^*. The denominator is a discounting factor, which takes into account the difference between the original path p_m and its sub-path p_m^*. The shorter the sub-path the bigger the discount, because it contains entities farther from the item.

For example, with respect to item i in Figure 7.2, we have:
$P_i = \{i \cdot e_1 \cdot e_4,\ i \cdot e_1 \cdot e_5,\ i \cdot e_2 \cdot e_5,\ i \cdot e_3 \cdot e_6,\ i \cdot e_3 \cdot e_7\}$
$P_i^* = [e_1 \cdot e_4,\ e_4,\ e_1 \cdot e_5,\ e_5,\ e_3 \cdot e_6,\ e_6,\ e_3 \cdot e_7,\ e_7]$

7.3.2. Feature combination

Each final feature vector x_i is obtained by concatenating a vector of collaborative features $\phi_{col}(i)$ to the item neighborhood mapping vector $\phi(G_i^h)$. Collaborative features are simply added by encoding in the feature vector those users who downloaded that item. The collaborative feature vector regarding the generic item is then:

$$\phi_{col}(i) = (w_{i,u_1}, w_{i,u_2}, ..., w_{i,u_1})$$

where $w_{i,u_1} = 1$ if user u_1 downloaded item i.

Although more sophisticated and advanced methods can be used for feature combination (Beliakov et al., 2015), our experimental evaluation (see Section 7.4) shows the effectiveness of our choice.

7.4. Experimental evaluation

For the evaluation of our approach we adopted the `All Unrated Items` methodology presented in Steck (2013). It consists in creating a *top-N* recommendation list for each user by predicting a score for every item not rated by that particular user, whether the item appears in the user test set or not. Then, performance metrics are computed comparing recommendation lists with test data. The evaluation has been carried out using the holdout method consisting in splitting the data in two disjoint sets: the one for training and the other for testing. We used 80% of user downloads for building the training set T and remaining 20% as test data for measuring recommendation accuracy. We repeated the procedure three times by randomly drawing new training/test sets in each round and averaged the results.

For measuring recommendation accuracy we adopted the following standard performance metrics: Precision and Recall. Precision@N (P@N) is computed as the fraction of *top-N* recommended items appearing in the test set, while Recall@N (R@N) is computed as the ratio of *top-N* recommended items appearing in the test set to the number of items in the test set. Note that in such implicit feedback setting all items in the test set are relevant. In addition to the standard precision and recall metrics we also measure the Mean Reciprocal Rank (MRR) which measure the quality of the highest ranked recommendations. For each user recommendation list the Reciprocal Rank (RR) measures how early in the list is positioned the first relevant recommendation.

As pointed out by McNee et al. (2006), the most accurate recommendations according to the standard metrics are sometimes not the recommendations that are most useful to users. In order to assess the utility of a recommender system, it is extremely important to evaluate also its capacity to suggest items that users would not readily discover for themselves, i.e., its ability to generate novel and unexpected results. The *Entropy-Based Novelty* (*EBN*) (Bellogín et al., 2010) expresses the ability of a recommender system to suggest fewer popular items, i.e., items not known by a wide number of users. In particular,

dataset	items	avg. tags	avg. keywords	resources	synsets	categories
Freesound	21,552	6.44	11.36	16,407	20,034	54,419
Last.fm	8,640	42.09	77.33	46,109	27,708	96,942

Table 7.1: Number of tags and keywords identified by Babelfy averaged by item, plus total number of distinct DBpedia resources, WordNet synsets and Wikipedia categories.

for each user's recommendation list L_u, the novelty is computed as:

$$EBN_u@N = -\sum_{i \in L_u} p_i \cdot \log_2 p_i$$

where:

$$p_i = \frac{|\{s_{ui} = 1 | u \in U\}|}{|U|}$$

Particularly, p_i is the ratio of users who downloaded item i. The lower $EBN_u@N$, the better the novelty.

Another important quality of the system is aggregate diversity. In our chapter we adopt the *diversity-in-top-N* metric presented in Adomavicius & Kwon (2012) that measures the distinct items recommended across all users. In particular we compute its normalized version with respect to the size of the item catalog. For brevity we refer to it as $ADiv@N$ and we compute it as follows:

$$ADiv@N = \frac{|\bigcup_u L_u|}{|I|}$$

This metric is an indicator of the level of personalization provided by a recommender system. Low values of aggregated diversity indicate that all users are being recommended almost the same few items. This corresponds to a low level of personalization of the system. In contrast, high values mean that users receive very different recommendations, which can be indirectly seen as a high level of personalization of the system.

All the reported metrics, besides aggregated diversity, are computed for each single user and eventually averaged.

7.4.1. Datasets description

Freesound dataset

We evaluated our approach on historical data about sound downloads collected from February 2005 to October 2013. In addition, we further enriched our knowledge graphs (see Section 7.2) with information coming from the Freesound Ontology (Font, 2015, chapter 6), a lightweight ontology where the 500 most popular Freesound tags are classified into 23 tag categories. From the original data dump, we selected a subset of sounds that fulfilled some criteria. We selected those sound with at least two tags classified in the Freesound Ontology. After that we filtered out all sounds with less than 10 downloads to reduce the sparsity of the implicit feedback matrix and have a fairer comparison with pure collaborative filtering methods. After some further data cleansing, the final dataset consisted in 20,000 users, 21,552 items, and 2,117,698 downloads. The sparsity of the implicit feedback matrix was 99.51%. Statistics on the enriched knowledge graph of the final dataset are shown in Table 7.1.

Last.fm dataset

To recreate most of the conditions of the Freesound dataset in a music recommendation scenario, a new dataset is created combining user's implicit feedback, tags, and textual descriptions of songs. This dataset combines a corpus of user's listening habits (Vigliensoni & Fujinaga, 2014), with tags and textual descriptions about songs. For every user in the corpus we chose the users' average listening count as a threshold to identify the relevant songs for each user. We only selected for our implicit feedback dataset user-song relations with a number of listens above each user's threshold. Moreover, only those songs that were relevant to at least 10 users, and users with at least 50 relevant songs were added to the dataset. The final dataset consisted in 5,199 users, 8,640 songs and 751,531 relations between users and songs. The sparsity of the implicit feedback matrix was 98.33%. This collaborative information was complemented with the list of top tags of every song provided by the Last.fm API, and a textual description of each song coming from Songfacts.com (cf. Section 4.3.1). Information about the enriched knowledge graph is shown in Table 7.1.

7.4.2. Experiment settings

As mentioned in Section 7.3, each user model is learnt using the Linear Support Vector Regression method. In particular we adopted the efficient *LIBLINEAR*[33] library and chose the *L2-regularized Support Vector Regression* (Ho & Lin, 2012). The tuning of the model hyper-parameters of the learning algorithm was performed through cross-validation on validation data obtained by selecting the 15% of feedback for each user from the training data. We set the parameters C and e by using a grid-search varying C from 0.1 to 1000 with step 10 and $e = \{0.1, 0.01\}$ (tolerance of termination criterion). Before the training we performed some pre-processing on the feature vectors. We removed those features appearing in fewer than 5 items and scaled all features to the range $[0, \ldots, 1]$ using min-max normalization. Finally each feature vector was normalized to unit length using the L2 norm.

[33] `http://www.csie.ntu.edu.tw/~cjlin/liblinear/`

Approach	Enrichment	h-hops	MRR	P@10	R@10	EBN@10	ADiv@10
Ent	fso	h=3	0.303	0.113	0.065	2.791	0.257
Ent	fso+KB/tag	h=3	0.303	0.115	0.066	2.617	0.332
Ent	fso+KB/tag	h=4	0.302	0.114	0.065	2.507	0.368
Ent	fso+KB/kw+tag	h=3	**0.306**	**0.118**	**0.067**	2.426	0.361
Ent	fso+KB/kw+tag	h=4	**0.306**	0.117	0.066	2.303	0.391
Path	fso	h=3	0.301	0.113	0.065	2.750	0.287
Path	fso+KB/tag	h=3	0.301	0.114	0.064	2.279	0.461
Path	fso+KB/tag	h=4	0.292	0.106	0.059	1.863	**0.556***
Path	fso+KB/kw+tag	h=3	0.304	0.116	0.065	2.019	0.461
Path	fso+KB/kw+tag	h=4	0.296	0.111	0.061	**1.618***	0.532
Col			0.293	0.110	0.062	2.890	0.181
Ent-noCol	fso+KB/kw+tag	h=3	0.154	0.058	0.034	0.384	0.591
Path-noCol	fso+KB/kw+tag	h=3	0.151	0.049	0.028	**0.369**	**0.670**
VSM	kw+tag	h=1	0.301	0.116	0.066	2.621	0.305
VSM-noCol	kw+tag	h=1	0.151	0.055	0.032	0.389	**0.670**
Audio Sim			0.022	0.004	0.002	0.382	0.044

Table 7.2: Accuracy, Novelty, and Aggregate Diversity results for different versions of the Freesound dataset. Best values in each column are in bold. The * symbol indicates best values for hybrid and collaborative configurations. `Ent` and `Path` refers to graph embedding options; `fso` to the initial Freesound Ontology, `KB` to WordNet and DBpedia enrichment; `tag` to item tags, and `kw` to text description keywords; `h` indicates the length of the h-hop neighborhood graph; `Col` means that only collaborative features are considered; `noCol` that no collaborative features are considered; `VSM` refers to Vector Space Model embedding; `Audio Sim` to the audio-based approach.

In the following we describe the experiments we carried out to evaluate our approach. In particular we are interested in evaluating the impact of semantic enrichment of the original data on the recommendation quality and the differences among the two feature mapping methods we implemented. Furthermore, we compare our approach with state-of-the-art algorithms for implicit feedback scenarios. All the differences between approaches and with respect to other baselines are statistically significant ($p < 0.01$) according to the paired t-test.

7.4.3. Sound recommendation experiment

Evaluation of the semantic item description enhancement

To evaluate the impact of the various features and information sources we built several variants of item feature vectors by varying: the information sources considered, the size of the item neighborhood graphs (number of hops) and the feature mapping method. In addition, we built a content-based approach purely based on 352 low-level audio features[34] extracted from the sound signal by using Essentia (Bogdanov et al., 2013b). In this approach, predictions are computed by aggregating the Euclidean distances between the feature vectors

[34] `https://www.freesound.org/docs/api/analysis_example.html#all-descriptors`

of each sound downloaded by the user and the target sound to recommend. All the results are reported in Table 7.2.

Looking at the accuracy results we see that there are no marked differences among all the feature vector variants. Noteworthy is that without considering the collaborative information (`noCol`) the accuracy drops significantly. In addition, when considering only collaborative features accuracy performances are comparable with respect to hybrid feature combination variants. The best hybrid semantic version `Ent(fso+KB/kw+tag/h=3)` is slightly better than pure collaborative. Regarding the comparison of the two mapping methods, the Entity-based item neighborhood mapping has generally slightly higher accuracy than the Path-based one. We can also note that considering too far entities in the graph does not improve accuracy. In fact, in both the two feature mapping when four hops are considered the results drop slightly with respect to three hops. Finally, we see that the semantic expansion of tags and terms do not improve consistently accuracy with respect to the usage of pure keywords and tags combined with collaborative information. The semantic configuration with highest accuracy (`Ent(fso+KB/kw+tag/h=3)`) is slightly better in terms of P@10 with respect to `VSM kw+tag`. We can also observe that the pure audio-based approach (`Audio Sim`) has by far lower performances than all the others.

Novelty and aggregate diversity results instead show more interesting insights. We observe that the semantic expansion, with both feature mappings, results in an improving of both novelty and aggregated diversity. In fact, the semantic enriched variant `(fso+KB+kw+tag/h=4)` has much better novelty and diversity than considering only the original Freesound Ontology `(fso)`. Furthermore, with respect to the variants without semantic expansion, that is the variants based only on keywords and tags, the usage of semantic expansion improves considerably novelty and diversity. Hence, thanks to this exploitation of the knowledge graph we are able to recommend good items that are also not so popular. We also see that the Path-based embedding has better performances than the Entity-based one. Such approaches allow to explore better the long-tail distribution of items and to increase the personalization of the system.
The variants without collaborative information are the ones with better novelty and diversity. The reason behind this behavior is that pure content-based approaches are not influenced by popularity biases. However, when using only content data the system recommends unpopular but very inaccurate items. Good novelty without accuracy does not imply good recommendation quality. Finally, the usage of collaborative information alone has much lower catalog coverage (aggregate diversity) than feature vectors containing also semantic features. For example `Path(fso+KB+kw+tag/h=4)` has comparable performances in terms of accuracy with respect to `Collab` but considerably better catalog coverage and novelty (lower EBN).

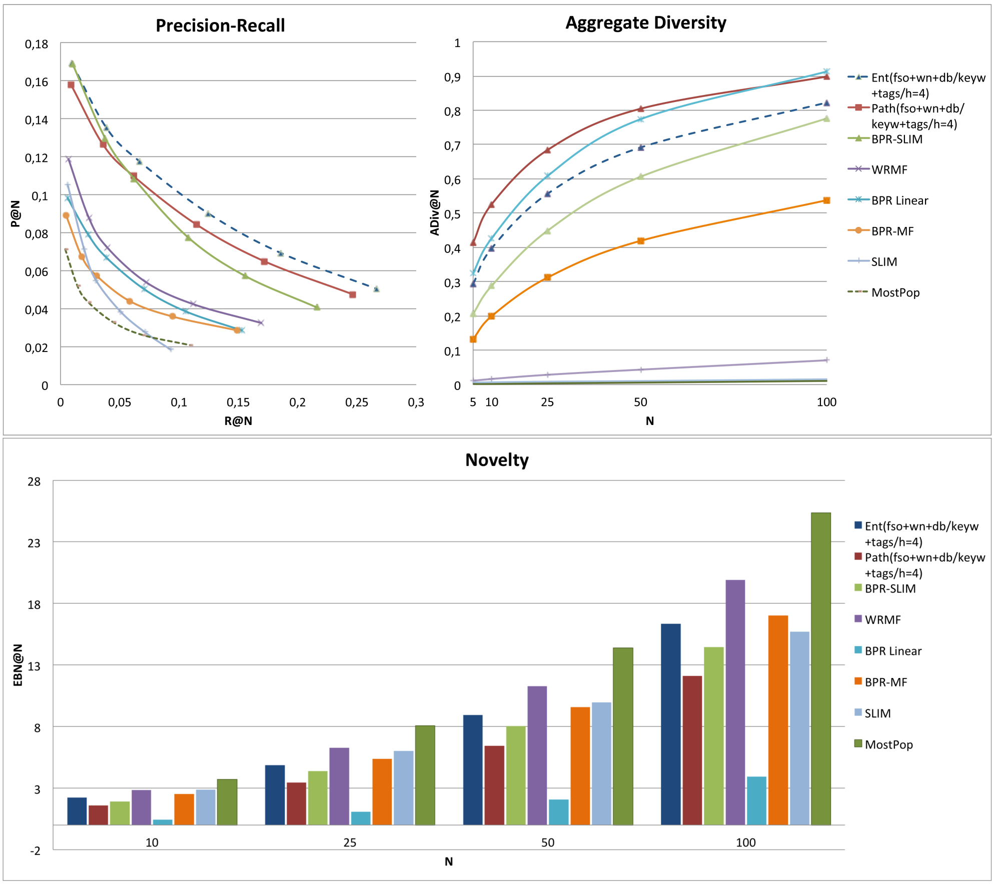


Figure 7.3: Precision-Recall, Novelty, and Aggregate Diversity plots in Freesound dataset

To conclude, we can state that the semantic expansion, especially when combined with the Path-based mapping, improves recommendation quality in terms of novelty and aggregated diversity. The intuition behind these results is that the semantic expansion allows the system to find items semantically related to the ones in the user profile. Conversely, when using only keyword or tag-based representations the system is able to retrieve only those few items with an exact keyword/tag match with those liked by the user. Thus, the system is unable to widely explore the item space to find those items that are semantically related to the ones liked by the user.

Comparison with other methods

We compared our approach with several state-of-the-art recommendation algorithms. `MostPop` is a popularity-based baseline that provides the same recommendation to all users based on the global popularity of items. `BPR-MF`

(Rendle et al., 2009) is a matrix factorization-based method optimized with Bayesian Personalized Ranking optimization criterion. `WRMF` is a weighted matrix factorization method (Hu et al., 2008). `SLIM` (Ning & Karypis, 2012) uses a Sparse Linear method for learning a sparse aggregation coefficient matrix. `BPR-SLIM` is similar to `SLIM` but it uses the BPR optimization criterion. `BPR Linear` is a hybrid matrix factorization method able to chapter with sparse datasets (Gantner et al., 2010). We used keywords and tags as item attribute data. The computation of the recommendations for all these comparative algorithms has been done with the publicly available software library *MyMediaLite*[35].

Figure 7.3 shows precision-recall, novelty and aggregated diversity plots. In those plots we report the competitive algorithms used for comparison and the `Ent(fso+KB/kw+tag/h=4)` and `Path(fso+KB+kw+tag/h=4)` configurations which we chose as representative for our approach due to its performances in terms of novelty and aggregate diversity.

With reference to the accuracy results we notice that our two approaches largely outperforms the others. The only method which is close to the approaches we propose is `BPR-SLIM`, which slightly outperforms `Path(fso+KB+kw+tag/h=4)` for low values of recommendation list length ($N = 5, 10$). With respect to the Novelty plot, our approach has much better novelty than all the other collaborative filtering algorithms but `BPR Linear`, which however have much lower accuracy. Our approach outperforms most of the collaborative filtering algorithms in terms of aggregated diversity. It is able to achieve a coverage of almost 80% and 90% for $N = 50$ and $N = 100$, respectively. The approach closer to ours is `BPR Linear` that for $N = 100$ reaches same performances. Also, `BPR-SLIM` and `BPR-MF` have acceptable diversity results. Instead, all the others have very low diversity results meaning that they focus mostly on a few specific items and recommend them to all users indiscriminately.

Summing up, the experimental results show that our approach is able to give more accurate and at the same time less popular recommendations, than collaborative filtering methods. It is able to better find good recommendations in the long tail. Effective recommendation systems should promote novel and relevant items taken primarily from the tail of the distribution. In addition, our approach shows much higher aggregated diversity which can be seen as a higher personalization of the system.

7.4.4. Music recommendation experiment

The recommendation algorithms we propose have been further validated on the Last.fm dataset. We performed the same experiments on this dataset to

[35] `http://www.mymedialite.net/`.

Approach	**Enrichment**	**h-hops**	**MRR**	**P@10**	**R@10**	**EBN@10**	**ADiv@10**
Ent	KB/tag	h=2	**0.612**	0.321	**0.122**	2.414	0.357
Ent	KB/tag	h=3	**0.612**	0.319	0.121	2.383	0.374
Ent	KB/tag	h=4	0.599	0.314	0.119	2.356	0.389
Ent	KB/kw+tag	h=3	0.604	0.315	0.114	2.448	0.316
Ent	KB/kw+tag	h=4	0.601	0.312	0.113	2.424	0.331
Path	KB/tag	h=3	0.570	0.287	0.108	2.112	0.479
Path	KB/tag	h=4	0.537	0.260	0.097	**1.911***	**0.544***
Path	KB/kw+tag	h=3	0.570	0.289	0.104	2.173	0.411
Path	KB/kw+tag	h=4	0.537	0.259	0.093	1.942	0.484
Collab			0.597	0.313	0.113	2.664	0.240
Ent-noCol	KB/tag	h=3	0.292	0.114	0.043	0.983	0.703
Path-noCol	KB/tag	h=3	0.285	0.113	0.043	**0.981**	**0.736**
VSM	tags	h=1	0.610	**0.322**	**0.122**	2.454	0.346
VSM	keyw	h=1	0.599	0.309	0.112	2.642	0.249

Table 7.3: Accuracy, Novelty, and Aggregate Diversity results for different versions of the Last.fm dataset. Best values in each column are in bold. The * symbol indicates best values for hybrid and collaborative configurations.

assess the applicability of the approach to other musical contexts.

Evaluation of the semantic item description enhancement

As we may notice from the results shown in Table 7.3, Entity-based embedding, `Collab`, and `VSM tags` approaches have very similar performance in terms of precision and recall. The first two Entity-based embedding variants have slightly higher MRR than `VSM tags`, meaning that they better locate relevant items in the top positions. Analogously to the previous sounds recommendation task, the approaches exploiting semantic expansion outperform the others in terms of novelty and aggregated diversity. The same tendency of the previous experiment is observed with the Entity-based and Path-based item neighborhood mappings. The Path-based approaches have lower precision, but much better novelty and aggregated diversity. Moreover, it is very interesting to observe that for both embedding options if we expand the graph by means of farther entities (h=4) precision decreases whilst novelty and diversity improve. It is noteworthy that differently from the results of the Freesound experiment, here we obtain higher accuracy with the approach that uses only tags and not keywords. Our interpretation of this trend is that, as shown in Table 7.1, the number of tags in the Freesound dataset is somehow scarce, and the addition of keywords taken from the textual descriptions improves the annotation of the items. On the other side, in the Last.fm dataset, the set of tags is already very rich, then the addition of keywords introduces noise within the items description thus deteriorating the accuracy of recommendations. Also in this experiment we can observe that when no collaborative features are used, accuracy is significantly worse even if novelty and diversity seem to be better. We may confirm from results in both experiments that collaborative features are a

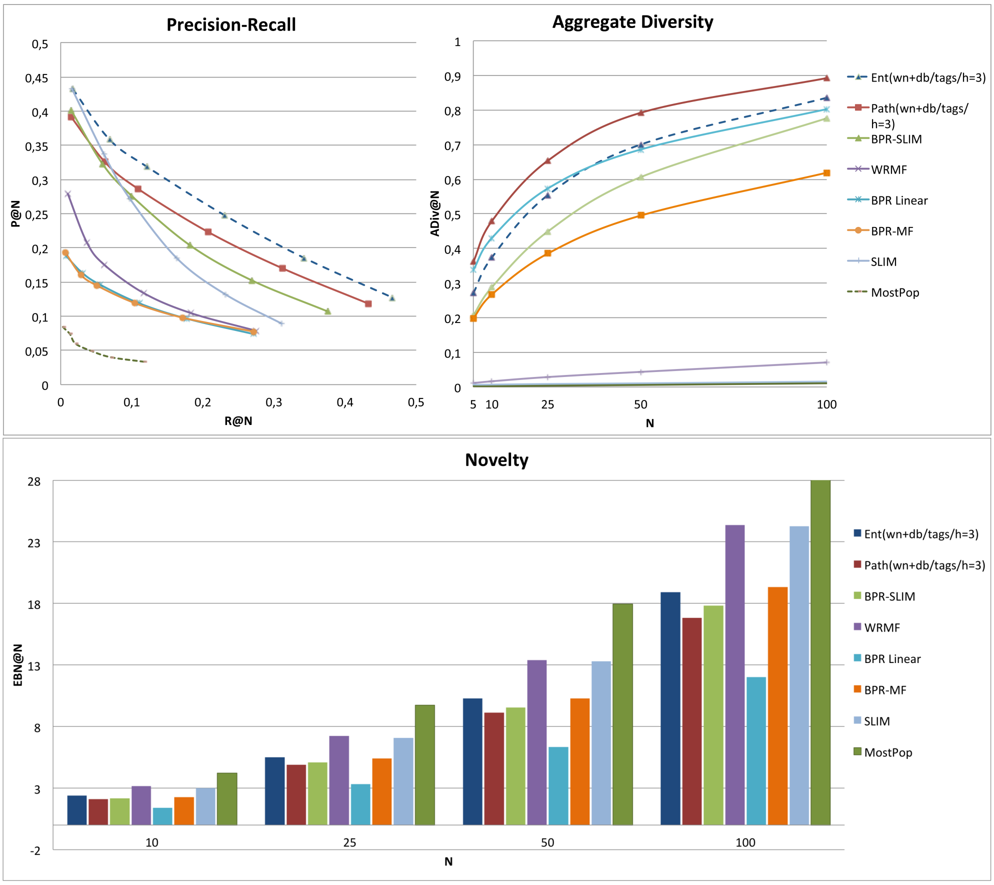


Figure 7.4: Precision-Recall, Novelty, and Aggregate Diversity plots in Last.fm dataset

very strong signal for the accuracy of the recommendations. Nonetheless, the inclusion of semantic features allows the system to further improve accuracy and provide novel and diverse recommendations, thus better leveraging the long tail.

Comparison with other methods

We compared our approach with the same set of state-of-the-art algorithms presented in the sound recommendation experiment. Based on the observations made in the previous paragraph, for this experiment we used only tags as item attribute data for `BPR Linear`. Figure 7.4 shows precision-recall, novelty, and aggregated diversity plots of the comparison with the other methods. We compare the competitive algorithms with the `Ent(KB/tag/h=3)` and `Path(KB/tag/h=3)` configurations which in this scenario results to be the most representative for our approach. Results are pretty similar to the ones observed in the sound recommendation experiment. Our two approaches largely outperform the others in terms of accuracy. `BPR-SLIM` and `SLIM` have performance similar to our Entity-based mapping approach for low values of recommendation list length (N = 5, 10), and slightly higher that the Path-based one. Our approaches have much better novelty results than all other collaborative filtering algorithms but `BPR Linear`, which again has much lower accuracy. In terms of aggregated diversity, our approach outperforms most of the collaborative filtering algorithms. `BPR Linear` achieves similar diversity, but much lower accuracy. Summing up, our approach is able to recommend less popular items with higher accuracy than other collaborative filtering algorithms also in this recommendation scenario. Therefore, our approach is able to improve the level of personalization of the recommended items, and better explore the long tail also for songs recommendation.

7.5. Conclusion

We have presented a hybrid approach to recommend musical items, i.e., sounds and songs, by exploiting the information encoded within a knowledge graph. We conducted various experiments on two different datasets, the one of sounds coming from Freesound.org, the other one of songs gathered from Last.fm and Songfacts.com. They may be considered as representative of the two classes of users we find in the music domain: producers looking for sounds to create new music and consumers looking for new songs to listen to.

Information coming from item descriptions and tags has been enriched with data coming from two external knowledge repositories: DBpedia and WordNet. Entity linking tools have been adopted to extract relevant entities from textual sources associated to musical items, namely tags and text descriptions, thus creating a new graph encoding the knowledge associated to users, items, and their mutual interactions. We then developed a recommendation engine that combines different features, that is semantic content-based ones extracted from the resulting knowledge graph and collaborative information from implicit user feedback. An evaluation with two explicit feature mappings, *entity-based item neighborhood* and *path-based item neighborhood*, has been conducted on both datasets in order to assess the performance of the system in terms of accuracy, diversity and novelty.

Experimental results in sounds and songs recommendation show that the proposed approach is able to improve the quality of the recommended list with respect to state of the art collaborative filtering algorithms and with respect to other content-based baselines. Our results also show that the data related to the music knowledge domain encoded in freely available datasets such as DBpedia or WordNet have reached a quality level that makes possible its usage in the creation of recommendation engines whose target are either music producers or music consumers. The semantic enrichment of the initial knowledge graph performed by means of entity linking techniques is a good choice to boost the performances of the system in terms of novelty and aggregate diversity. A knowledge-based approach can improve the degree of personalization in the recommendations of musical items from various points of view such as prediction accuracy, catalog coverage, and promote long-tail recommendations. We have presented a methodology that achieves these objectives by combining semantic knowledge with collaborative information.

Part II

Representation Learning from Multimodal Data

CHAPTER 8

Cold-start Music Recommendation

8.1. Introduction

An increasing amount of digital music is being published daily. Music streaming services often ingest all available music, but this poses a challenge: how to recommend new artists for which prior knowledge is scarce? In this chapter we aim to address this so-called cold-start problem by learning and combining multimodal data representations and user feedback data using deep learning architectures.

Social tags have been extensively used as a source of artist content features to recommend music (Knees & Schedl, 2013), however, these tags are usually collectively annotated, which often introduce an artist popularity bias (Turnbull et al., 2008a). Artist biographies and press releases, on the other hand, do not necessarily require a collaborative effort, as artists themselves may produce them. However, they have seldom been exploited for music recommendation. Part of this chapter focuses on learning data representations from these biographies. Furthermore, we also make use of audio signals, since these are generally always available and have shown to be helpful when recommending music in the long tail (Van den Oord et al., 2013).

According to Gülçehre & Bengio (2016), composing simpler tasks is more likely to yield effective local minima for neural networks. In addition, as stated in Larochelle et al. (2009), directly training all the layers of a deep network together make it difficult to exploit all the extra modeling power of a deeper architecture. Therefore, we decided to separate the problem of music recommendation into artist and song levels. Artist feature embeddings are learned from artist metadata in an artist recommendation scenario. Track feature embeddings are learned from audio signals in a song recommendation scenario. In both cases, a hybrid recommendation approach is used based on learning attribute-to-feature mappings (Gantner et al., 2010). This method addresses the lack of feedback for uncommon items in two steps: (1) factorizing the collaborative matrix, and (2) learning a mapping between item content features and item latent factors (Van den Oord et al., 2013; Bansal et al., 2016). Lastly, both feature embeddings are combined in a multimodal network to predict song recommendations of cold-start artists. We show how dividing the problem into artists and songs, and combining text and audio in a multimodal approach yields improved recommendations.

The rest of the chapter is organized as follows. First, we describe in detail the recommendation approach (Section 8.2). Then, we describe the architectures used to obtain artist text embeddings (Section 8.3), track audio embeddings (Section 8.4), and their combination (Section 8.5). Experiments and evaluation results are reported in Section 8.6, and the chapter ends with a discussion about our findings (Section 8.7).

8.2. Recommendation approach

To produce cold-start music recommendations, we propose the following framework. Given the set of artist features A_s of a song s, and the set of track features T_s of s, the complete feature set of s is defined as the aggregation of its artist and track features $F_s = A_s \cup T_s$.

Given the heterogeneity of these two feature sets (audio and text), a learning process involving them may under-explore one of the modalities, as the stronger modality may dominate quickly. To ensure that the variability of the input data is fully represented, we divide the problem into three phases (see Figure 8.1). First, we aggregate the collaborative information of all songs of the same artist, and learn an artist feature embedding A'_s from A_s in an artist recommendation scenario. Second, we learn a track feature embedding T'_s from T_s in a pure audio-based recommendation scenario. Third, we combine both feature embeddings A'_s and T'_s in a multimodal network and compute song recommendations.

Since songs from the same artist share the same set A_s, if different songs from the same artist appear in multiple sets (e.g., train and test), a problem of overfitting may arise (Flexer, 2007). To approach this issue, we use non-overlapping artists across the train, validation, and test sets.

Let M be the matrix of implicit feedback, where m_{us} is the number of play counts for user u on song s. M is split into M_{train}, M_{val} and M_{test}, for train, validation and test, respectively, where no artist is shared across sets. Factorizing M_{train} using weighted matrix factorization (WMF) (Hu et al., 2008) yields I_k and U_k, the k dimensional sets of song and user latent factors, respectively. We set $k = 200$, and apply the alternating least squares (ALS) optimization

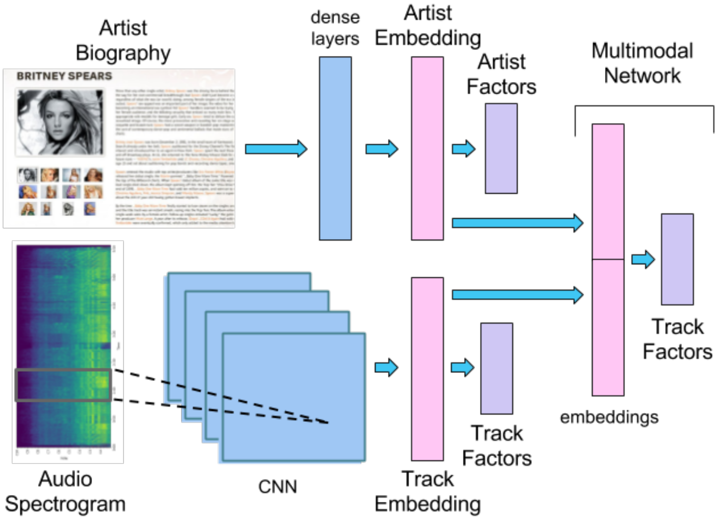


Figure 8.1: Model architecture.

method.

To learn the artist embeddings, we obtain the matrix of artist implicit feedback R from M, being $R_{ua} = \sum_s m_{us}$ for all songs s from the same artist a. This matrix is split into train, validation, and test sets following the same partition of artists made for M, and thus keeping the mutual exclusion restriction. Latent factors of artists and users are later obtained via WMF. Lastly, a deep neural network is trained on the prediction of artist latent factors from artist content features A. On the other hand, the song latent factors are predicted with a deep convolutional network, using I_k as training data and the track features T as input (similar to Van den Oord et al. (2013)).

Once the artist and track models are trained and optimized, we gather the activations from the penultimate layer of each network for all the sets. These activations constitute what we call the artist and track feature embeddings A'_s and T'_s, which are in turn used as input to a third network. This final multimodal network is trained on the prediction of song latent factors I_k from $S'_s = A'_s \cup T'_s$. Finally, the list of item recommendations for user u is obtained by ranking the results of computing the dot product between the user latent factor $f_u \in U_k$ and the set of item factors.

The different architectures used in each one of the three neural networks involved in the approach are described in Sections 8.3, 8.4, and 8.5, respectively. Nevertheless, all networks have a final fully connected layer of 200 units[36] with linear activation and L2-normalization. In addition, mini batches of 32 items are randomly sampled from the training data to compute the gradient in all

[36] to match the dimensions of the factors to be predicted.

Entity class	**Properties**
MusicalArtist | Band	activeYearsStartYear, homeTown, birthPlace, gerne, instrument, recordLabel, associatedBand, associatedMusicalArtist, bandMember, formerBandMember, mentor
MusicalWork	writer, artist, genre, recordLabel, album, musicalArtist, musicalBand, releaseDate, producer, recordedIn
RecordLabel	location, parentCompany, genre, foundedBy
MusicGenre	stylisticOrigin, instrument, subject

Table 8.1: DBpedia properties selected for each entity class.

networks, and Adam (Kingma & Ba, 2014) is the optimizer used to train the models, with the default suggested learning parameters. Given that the outputs of the architectures are L2-normalized, we use cosine proximity as the loss function, as in Chollet (2016).

8.3. Learning artist representations from text

In this section we describe two different, competing approaches to exploit artist texts in a deep learning process.

8.3.1. Semantic enrichment

We propose a method for enriching artist biographies by associating text fragments with relevant entities defined in online knowledge repositories, and then gathering relevant semantic information about them. For this purpose, we adopted Babelfy (Moro et al., 2014b) and ELVIS (see Section 3.3). We use semantic information about the identified entities coming from DBpedia to enrich the biographies.

As shown in Chapter 7, entity linking systems may be useful for music recommendation. However, as illustrated in Chapters 3 and 4, they are not optimized for the music domain, and are prone to errors. The application of a filtering process over the set of identified entities based on their classification within the DBpedia Ontology, has demonstrated its utility to improve music retrieval tasks, such as artist similarity (see Section 6.2). Therefore, we only keep entities of classes related to the music domain such as *MusicalArtist*, *Band*, *MusicGenre*, *MusicalWork*, *RecordLabel*, *Instrument*, *Engineer*, *Language*, *EthnicGroup*, and *Place*. Then, we query DBpedia to get all the available information about the filtered entities. From the information gathered, we keep some specific properties for every entity, depending on the entity class (see Table 8.1). In addition, we also kept all the Wikipedia categories associated to each entity.

To build the enriched biographies we proceed as follows: First, Babelfy is applied over the biography texts. Second, information is gathered from DBpedia for the entities of the selected classes. Finally, the collected data are added at the end of the biography text separated by spaces. A Vector Space Model (VSM) is then applied to the set of enriched biographies, and tf-idf weighting (Zobel & Moffat, 1998) is used, similarly to the enrichment process applied in Section 6.3. We limited the vocabulary size to 10,000 terms for the VSM, as this number provides a good trade-off between performance and number of parameters required for training. Note that either words, entities, dates, or categories may be part of this vocabulary. From this data representation, a feedforward network with two dense layers of 2048 neurons each is trained to predict the artist latent factors. The latter of these hidden layers becomes the vector embedding with the learned data representations to be used in the multimodal approach described in Section 8.5.

8.3.2. Word embeddings

Much of the work with deep learning in Natural Language Processing has involved the learning of word vector representations (Bengio et al., 2003; Mikolov et al., 2013a), and their further composition (Collobert et al., 2011). Word embeddings aim to represent words as low-dimensional dense vectors. They have demonstrated to greatly benefit NLP tasks, such as word similarity, sentiment analysis, or parsing (Nguyen et al., 2016).

The use of convolutional neural networks (CNN) over pre-trained word vectors has become state of the art in sentence classification (Kim, 2014). We re-adapt the architecture proposed in Kim (2014) for sentence classification to learn artist latent factors from artist biographies. This consists in an embedding layer, followed by a one-dimensional convolutional layer with multiple filter widths, a max-over-time pooling layer, a dense hidden layer and the output layer. We employ the same architecture and parameters, changing only the output layer and the loss function. We initialize the input embedding layer of the network with word2vec word embeddings pre-trained on the Google News dataset, and also with word embeddings trained in our own corpus of biographies. The dense hidden layer right before the output layer constitutes the vector embedding to later use in the multimodal approach (see Section 8.5).

8.4. Learning track representations from audio

It is common in the field of music informatics to make use of CNNs to learn higher-level features from spectrograms. These data representations are typically contained in $\mathbb{R}^{\mathcal{F}\times N}$ matrices with $\mathcal{F}$ frequency bins and N time frames. In our approach, we compute 96 frequency bin, log-compressed constant-Q transforms (CQT) (Schörkhuber & Klapuri, 2010) for all the tracks in our dataset using `librosa` (Mcfee et al., 2015) with the following parameters: audio sampling rate at 22050 Hz, hop length of 1024 samples, Hann analysis window, and 12 bins per octave. Following a similar approach to Van den Oord et al. (2013), we address the variability of the length N across songs by sampling one 15-seconds long *patch* from each track, resulting in the fixed-size input to the CNN.

The deep model trained with these data is defined as follows: the CQT patches are fed to four convolutional layers with rectified linear units (ReLU) as activations. The four convolutions have the following number of filters, from first to last: 256, 512, 1024, and 1024. The convolutions are only applied to the time axis, leaving the frequencies fixed since the absolute and relative bin placement is important when aiming to capture particular sounds (as opposed to the irrelevance of *where* in time a certain sonic event occurs). Maxpooling of 4 units across the time axis is applied after each of the first three ReLUs, and 50% dropout is applied to all layers. The flattened output of the last layer has 4096 units, which becomes the vector embedding to later use in the multimodal approach described next.

8.5. Multimodal fusion

There are several approaches in the literature for multimodal feature learning (Ngiam et al., 2011; Srivastava & Salakhutdinov, 2012), and late fusion of multimodal feature vectors (Rouvier et al., 2015; Slizovskaia et al., 2017). In our approach, audio and text feature vectors are learned separately and then combined via late fusion in a multimodal network (see Figure 8.1).

Given the different nature of the artist and track embeddings, a normalization step is necessary. Normalized feature vectors are then fed to a feed forward neural network (a simple Multi Layer Perceptron, MLP). Two different architectures were explored: (i) each embedding vector is connected to an isolated dense layer of 512 hidden units with ReLU activations after a process of batch normalization (Ioffe & Szegedy, 2015). Then, both dense layers are connected to the output layer. The rationale behind this is that the isolated dense layers help the network learn non-linearities from each modality separately. (ii) each embedding vector is $L2$-normed and then concatenated into a single feature vector which is directly connected to the output layer, resulting in a linear model. Regularization is obtained by applying dropout with an empirically selected factor of 70% after the input layer for both architectures.

8.6. Experiments

8.6.1. Dataset

The Million Song Dataset (MSD) (McFee et al., 2012) is a collection of metadata and precomputed audio features for 1 million songs. Along with this dataset, the Echo Nest Taste Profile Subset (Bertin-Mahieux et al., 2011) provides play counts of 1 million users on more than 380,000 songs from the MSD. Starting from this subset, we gather biographies and social tags from Last.fm for all the artists that have at least one song in the dataset. When there are several artists with the same name, they are stored in the same page of Last.fm, which makes the biography and social tags ambiguous. We automatically removed all ambiguous artists by applying text processing on the biographies. The song features provided with the MSD are not generally suitable for deep learning, so we instead use audio previews between 7 and 30 seconds retrieved from `7digital.com`. After removing ambiguous artists and missing tracks, the final dataset consists of 328,821 tracks from 24,043 artists. Each track has at least 15 seconds of audio, each biography is at least 50 characters long, and each artist has at least 1 tag associated with it. All artist metadata, implicit feedback matrices, and splits are released as a new dataset called the MSD-A.

8.6.2. Artist recommendation

To investigate to what extent the different feature sets, data models and architectures influence the quality of the deep artist features, we evaluate the different approaches in an artist recommendation scenario. Given the matrix of implicit feedback R, and the set of artist and user factors obtained through matrix factorization (see Section 8.2), we predict the artist factors for the test set, and use them to compute a ranked list of recommended artists for every user. We use mean average precision (MAP) with a cut-off at 500 recommendations per user as our evaluation measure.

We compare four different approaches using the biography texts as input. (1) A pure text-based approach using a VSM and a feedforward network (A-TEXT). (2) Similar to (1) but with a semantically enriched version of the texts (A-SEM) (see Section 8.3.1). (3) A CNN approach based on word embeddings initialized with Google News vectors (A-W2V-GOO) (see Section 8.3.2). (4) Similar to (3) but initializing the embeddings with word vectors previously trained on the corpus of biographies (A-W2V). To properly frame the results, we compute two baselines and one competitor approach. The TAGS baseline approach uses artist social tags as input features, and TEXT-RF uses biography texts as input, but Random Forest Regression for the learning instead of a deep neural network. The former baseline is added to compare the potential of biography

Approach	Input	Data model	Arch	MAP
A-TEXT	Bio	VSM	FF	0.0161
A-SEM	**Sem Bio**	**VSM**	**FF**	**0.0201**
A-W2V-GOO	Bio	w2v-pretrain	CNN	0.0119
A-W2V	Bio	w2v-trained	CNN	0.0145
A-TAGS	Tags	VSM	FF	0.0314
TAGS-ITEMKNN	Tags	-	itemKnn	0.0161
TEXT-RF	Bio	VSM	RF	0.0089
RANDOM	-	-	-	0.0014
UPPER-BOUND	-	-	-	0.5528

Table 8.2: Artist Recommendation Results. Mean average precision (MAP) at 500 for the predictions of artist recommendations in 1M users. VSM refers to Vector Space Model, FF to Feedforward, RF to Random Forest, CNN to Convolutional Neural Network, and itemKnn to itemAttributeKnn approach. Bio refers to biography texts and Sem Bio to semantically enriched texts.

texts with respect to curated metadata, whilst the latter was added to study to which extent the deep network improves the results over other learning methods typically used in Natural Language Processing. There are few recommendation approaches able to deal with an extreme cold-start scenario like ours. Therefore, we select ItemAttributeKnn (Gantner et al., 2010) as the competitor approach (TAGS-ITEMKNN), using artist social tags as attribute data and computed using the MyMediaLite library[37]. We also show the scores achieved when the latent factor vectors are randomized (RANDOM), and when they are learned from feedback data using matrix factorization (UPPER-BOUND).

Results reported in Table 8.2 show that the semantic enrichment of the biographies (A-SEM) outperforms the pure text approach A-TEXT. As expected, the use of tags improves the results over the use of text. However, the addition of semantic features reduces the gap in performance between the use of tags and unstructured text. Moreover, the difference between A-TEXT and TEXT-RF shows that the use of deep learning with respect to random forest improves the results. We also note that a VSM model with a feedforward network outperforms the use of word embeddings with convolutions. Although, according to the literature, this latter approach has demonstrated its utility for simple tasks like binary classification with short texts, our task puts forward two challenges for this architecture: the greater length of the input texts, and the higher dimensionality of the output. Although we have shown that initializing the embedding layer with word vectors trained on the corpus itself (A-W2V) outperforms the use of Google News pre-trained vectors (A-W2V-GOO), further

[37] `http://www.mymedialite.net/`

Approach	Artist Input	Track Input	Arch	MAP
AUDIO	-	audio spec	CNN	0.0015
SEM-VSM	Sem Bio	-	FF	0.0032
SEM-EMB	A-SEM	-	FF	0.0034
MM-LF-LIN	**A-SEM**	**AUDIO emb**	**MLP**	**0.0036**
MM-LF-H1	**A-SEM**	AUDIO emb	MLP	0.0035
MM	Sem Bio	audio spec	CNN	0.0014
TAGS-VSM	Tags	-	FF	0.0043
TAGS-EMB	A-TAGS	-	FF	0.0049
RANDOM	rnd emb	-	FF	0.0002
UPPER-BOUND	-	-	-	0.1649

Table 8.3: Song Recommendation Results. Mean average precision (MAP) at 500 for the predictions of song recommendations in 1M users. AUDIO emb refers to the track embedding of AUDIO approach, SEM to artist embedding of SEM approach, TAGS to artist embedding of TAGS approach, spec to spectrogram, mm to multimodal, lf to late fusion, lin to linear, and h1 to one hidden layer.

work is necessary to properly optimize a convolutional architecture for this task. Finally, we observe that our approach A-TAGS outperforms the competitor approach TAGS-ITEMKNN using the same item attributes.

Once the network is trained, we predict the activations of the penultimate layer for the entire dataset of artists. Thus, we obtain a vector embedding of 2048 dimensions, which represents the artist deep features A'. From the evaluated approaches, we compute the artist embedding from the A-SEM and A-TAGS approaches.

8.6.3. Song recommendation

In this experiment, audio embeddings are obtained after training the convolutional network (see Section 8.4) with 260k patches of 15 seconds, corresponding to the 80% of the tracks described in Section 8.6.1. Patches are divided into training (80%), validation (10%) and test (10%) sets. Results reported in Table 8.3 are computed over the remaining 20% of tracks. As opposed to Van den Oord et al. (2013), no artist appears in more than one subset to avoid overfitting. Finally, multimodal approaches are computed on the same sets.

In our experiments, we want to measure the impact of the artist embeddings in the song recommendation problem, and also the potential of the multimodal approach. We experimented with two artist embedding approaches, SEM-EMB and TAGS-EMB, that exploit the data representations learned from the artists attributes (see Section 8.6.2), either based on biography texts (A-SEM) or artists tags (A-TAGS). To measure the potential of the artist embeddings, we also computed two approaches using as input the original artist attributes (SEM-VSM for semantically enriched texts and TAGS-VSM for tags). Results on Table 8.3 show that SEM-EMB and TAGS-EMB outperform SEM-VSM and TAGS-VSM, suggesting that using artist representations learned from the aggregated feedback data outperforms learning directly from the original artist attributes in song recommendation.

An approach based on the audio spectrograms was computed (AUDIO). From this latter approach, audio embeddings where obtained (AUDIO emb) and combined with A-SEM in a multimodal late fusion approach MM-LF-LIN (without hidden layers and $l2$-norm) and MM-LF-H1 (with one hidden layer after each feature vector and batch normalization) (see Section 8.5). We also tried with different combinations of hidden layers and normalization steps in the multimodal network but all of them yielded lower results than the ones reported for MM-LF-LIN and MM-LF-H1. We compared this network with a multimodal approach trained directly on the original features (semantically enriched text and audio spectrograms). Results on the combination of artist and track features show that the late fusion of artist and track embeddings (MM-LF-LIN) clearly outperforms the simultaneous training of artist and track initial features (MM). In addition, we observe that we achieve better results when no hidden layer is added to the multimodal network (MM-LF-LIN). Finally, we observe that the multimodal approach that combines text and audio features with late fusion (MM-LF-LIN) improves the results of pure text (SEM-EMB) or pure audio (AUDIO) approaches. All the differences between the approaches are statistically significant ($p < 0.01$) according to the paired t-test.

We also compared the results with an upper-bound approach obtained from the feedback data and an approach trained with random vector embeddings. Although results are in general far from the upper-bound, the comparative analysis of the proposed approaches gives some insights of the behavior of different feature representations and modalities in the cold-start recommendation problem.

8.7. Conclusions

In this chapter, a multimodal approach for cold-start music recommendation has been presented. The approach is divided into three steps. (1) Artist data representations are learned from text and semantic features in an artist recommendation scenario using a deep learning architecture. (2) Track data representations are learned from the audio spectrograms using convolutional neural networks. (3) Learned representations are combined in a multimodal network.

Results show that splitting the problem of music recommendation at artist and song levels improves the quality of recommendations. Learning artist data representations separately benefits from the aggregation of the information about the different songs of the same artist, yielding more robust artist features. Related to this, an approach for the semantic enrichment of artist metadata has been proposed, leading to a significant improvement in the results. In addition, we have shown the potential of exploiting artist biographies in music recommendation. Moreover, the deep learning architectures used have demonstrated their capacity to improve upon other learning models under the music recommendation framework.

Finally, we have shown how a multimodal approach, based on the late fusion of track and artist feature embeddings that are learned separately, outperforms end-to-end multimodal approaches where the different modalities are learned simultaneously. Moreover, results have shown that our multimodal approach achieves better results than pure text or audio approaches.

CHAPTER 9

Multi-label Music Genre Classification

9.1. Introduction

Music genres allow to categorize musical items that share common characteristics. However, almost all related work is concentrated in multi-class classification of music items into broad genres (e.g., Pop, Rock), assigning a single label per item. This is problematic since there may be hundreds of more specific music genres (Pachet & Cazaly, 2000), and these may not be necessarily mutually exclusive. In this chapter we aim to advance the field of music genre classification by framing it as multi-label genre classification of fine-grained genres.

To this end, we present *MuMu*, a new large-scale multimodal dataset for multi-label music genre classification. *MuMu* contains information of roughly 31k albums classified into one or more 250 genre classes. For every album we analyze its cover image, text reviews, and audio tracks, with a total number of approximately 147k audio tracks and 447k album reviews. Furthermore, we exploit this dataset with a novel deep learning approach to learn multiple genre labels for every album using different data modalities (i.e., audio, text, and image). Internal data representations of each modality are extracted from the neural networks used for classification. Next, we combine these representations to study how the different combinations behave.

Results show how representation learning using deep neural networks substantially surpasses traditional approaches based on handcrafted features, reducing the gap between text-based and audio-based classification (see Section 6.3.6). Moreover, an extensive comparative of different deep learning architectures for audio classification is provided, including the usage of a dimensionality reduction approach that yields improved results. Finally, we show how the late fusion of data representations learned from different modalities achieves better scores than each of them individually.

The rest of this chapter is structured as follows. First, in Section 9.2, we present the multimodal dataset collected for the experiments. Then, the multi-label classification problem is exposed (Section 9.3). The architectures for album genre classification are described next (Section 9.4). In Section 9.5 we describe the experiments performed. Finally, we conclude the chapter with a discussion about our findings (Section 9.6).

9.2. Multimodal dataset

To the best of our knowledge, there are no publicly available large-scale datasets that encompass audio, images, text, and multi-label genre annotations. Therefore, we present *MuMu*, a new Multimodal Music dataset with multi-label genre annotations that combines information from the Amazon Reviews dataset (McAuley et al., 2015b) and the Million Song Dataset (MSD) (Bertin-Mahieux et al., 2011). The former contains millions of album customer reviews and album metadata gathered from Amazon.com. The latter is a collection of metadata and precomputed audio features for a million songs.

To map the information from both datasets we use MusicBrainz. For every album in the Amazon dataset, we query MusicBrainz with the album title and artist name to find the best possible match. Matching is performed using the same methodology described in Section 5.3.1, following a pair-wise entity resolution approach based on string similarity. Following this approach, we were able to map 60% of the Amazon dataset. For all the matched albums, we obtain the MusicBrainz recording ids of their songs. With these, we use an available mapping from MSD to MusicBrainz[38] to obtain the subset of recordings present in the MSD. From the mapped recordings, we only keep those associated with a unique album. This process yields the final set of 147,295 songs, which belong to 31,471 albums.

As stated in Section 8.6.1, the song features provided by the MSD are not generally suitable for deep learning, so we also use in these experiments audio previews between 7 and 30 seconds retrieved from `7digital.com`. For the mapped set of albums, there are 447,583 customer reviews in the Amazon Dataset. In addition, the Amazon Dataset provides further information about each album, such as genre annotations, average rating, selling rank, similar products, cover image url, etc. We employ the provided image url to gather the cover art of all selected albums. The mapping between the three datasets (Amazon, MusicBrainz, and MSD), genre annotations, data splits, text reviews, and links to images are released as the *MuMu* dataset.

[38] `http://labs.acousticbrainz.org/million-song-dataset-echonest-archive`

Genre	% of albums	Genre	% of albums
Pop	84.38	Tributes	0.10
Rock	55.29	Harmonica Blues	0.10
Alternative Rock	27.69	Concertos	0.10
World Music	19.31	Bass	0.06
Jazz	14.73	European Jazz	0.06
Dance & Electronic	12.23	Piano Blues	0.06
Metal	11.50	Norway	0.06
Indie & Lo-Fi	10.45	Slide Guitar	0.06
R&B	10.10	East Coast Blues	0.06
Folk	9.69	Girl Groups	0.06

Table 9.1: Top-10 most and least represented genres.

9.2.1. Genre labels

Amazon has its own hierarchical taxonomy of music genres, which is up to four levels in depth. In the first level there are 27 genres, and almost 500 genres overall. In our dataset, we keep the 250 genres that satisfy the condition of having been annotated in at least 12 albums. Every album in Amazon is annotated with one or more genres from different levels of the taxonomy. The Amazon Dataset contains complete information about the specific branch from the taxonomy used to classify each album. For instance, an album annotated as Traditional Pop comes with the complete branch information *Pop / Oldies / Traditional Pop*. To exploit both the taxonomic and the co-occurrence information, we provide every item with the labels of all their branches. For example, an album classified as *Jazz / Vocal Jazz* and *Pop / Vocal Pop* is annotated in *MuMu* with the four labels: Jazz, Vocal Jazz, Pop, and Vocal Pop. There are in average 5.97 labels for each song (3.13 standard deviation).

The labels in the dataset are highly unbalanced, following a distribution that might align well with those found in real world scenarios. In Table 9.1 we list the top 10 most and least represented genres and the percentage of albums annotated with each label. The unbalanced character of the genre annotations poses an interesting challenge for music classification that we also aim to exploit. Among the multiple possibilities that this dataset may offer to the MIR community, in this chapter we focus on the multi-label classification problem, described next.

9.3. Multi-label classification

In multi-label classification, multiple target labels may be assigned to each classifiable instance. More formally: given a set of n labels $L = \{l_1, l_2, \dots, l_n\}$, and a set of m items $I = \{i_1, i_2, \dots, i_m\}$, we aim to model a function f able to associate a set of c labels to every item in I, where $c \in [1, n]$ varies for every item.

Deep learning approaches are well-suited for this problem, as these architectures allow to have multiple outputs in their final layer. The usual architecture for large multi-label classification using deep learning ends with a logistic regression layer with sigmoid activations evaluated with the cross-entropy loss, where target labels are encoded as high-dimensional sparse binary vectors (Szegedy et al., 2016). This method, which we refer as LOGISTIC, implies the assumption that the classes are statistically independent (which is not the case in music genres).

A more recent approach (Chollet, 2016), relies on matrix factorization to reduce the dimensionality of the target labels. This method makes use of the inter-relation between labels, embedding the high-dimensional sparse labels onto lower-dimensional vectors. In this case, the target of the network is a dense lower-dimensional vector, which can be learned using the cosine proximity loss, as these vectors tend to be $L2$-normalized. We denote this technique as COSINE, and we provide a more formal definition next.

9.3.1. Labels factorization

Let M be the binary matrix of items I and labels L where $m_{ij} = 1$ if i_i is annotated with label l_j and $m_{ij} = 0$ otherwise. Using M, we calculate the matrix X of Positive Pointwise Mutual Information (PPMI) for the set of labels L. Given L_i as the set of items annotated with label l_i, the PPMI between two labels is defined as:

$$X(l_i, l_j) = max\left(0, \log \frac{P(L_i, L_j)}{P(L_i)P(L_j)}\right) \tag{9.1}$$

where $P(L_i, L_j) = |L_i \cap L_j|/|I|$ and $P(L_i) = |L_i|/|I|$.

The PPMI matrix X is then factorized using Singular Value Decomposition (SVD) such that $X \approx U\Sigma V$, where U and V are unitary matrices, and Σ is a diagonal matrix of singular values. Let Σ_d be the diagonal matrix formed from the top d singular values, and let U_d be the matrix produced by selecting the corresponding columns from U, the matrix $C_d = U_d \cdot \sqrt{\Sigma_d}$ contains the label factors of d dimensions. Finally, we obtain the matrix of item factors F_d as $F_d = C_d \cdot M^T$. Further information on this technique may be found in Levy & Goldberg (2014).

Factors present in matrices C_d and F_d are embedded in the same space. Thus, a distance metric such as cosine distance can be used to obtain distance measures between items and labels. Similar labels are grouped in the space, and at the same time, items with similar sets of labels are near each other. These properties can be exploited in the label prediction problem.

9.3.2. Evaluation metrics

The evaluation of multi-label classification is not necessarily straightforward. Evaluation measures vary according to the output of the system. In this problem, we are interested in measures that deal with probabilistic outputs, instead of binary. The Receiver Operating Characteristic (ROC) curve is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied. Thus, the area under the ROC curve (AUC) is often taken as an evaluation measure to compare such systems. We selected this metric to compare the performance of the different approaches as it has been widely used for genre and tag classification problems (Choi et al., 2016a; Dieleman & Schrauwen, 2014).

The output of a multi-label classifier is a label-item matrix. Thus, it can be evaluated either from the labels or the items perspective. We can measure how accurate the classification is for every label, or how well the labels are ranked for every item. In this work, the former point of view is evaluated with the AUC measure, which is computed for every label and then averaged. We are interested in classification models that strengthen the diversity of label assignments. As the taxonomy is composed of broad genres that are over-represented in the dataset (see Table 9.1) and more specific subgenres (e.g., Vocal Jazz, Britpop), we want to measure whether the classifier is focusing only on over-represented genres, or on more fine-grained ones. To this end, we use aggregated diversity (Adomavicius & Kwon, 2012), also known as catalog coverage. ADiv@N measures the percentage of normalized unique labels present in the top N predictions across all test items (see Section 7.4). Values of $k = 1, 3, 5$ are typically employed in multi-label classification (Jain et al., 2016) .

9.4. Album genre classification

In this section we exploit the multimodal nature of the *MuMu* dataset to address the multi-label classification task. More specifically, and since each modality on this set (i.e., cover image, text reviews, and audio tracks) is associated with a music album, our task focuses on album classification.

In what follows, we define the architectures and methods for the prediction of genre labels at the album level from each modality using deep learning. Furthermore, we describe how to combine data representations learned from every modality in a single multimodal model, thus taking advantage of all available data in *MuMu*.

9.4.1. Audio-based approach

A music album is composed by a series of audio tracks, each of which may be associated with different genres. In order to learn the album genre from a set of audio tracks we split the problem into three steps: (1) track feature vectors are learned while trying to predict the genre labels of the album from every track in a deep neural network. (2) Track vectors of each album are averaged to obtain album feature vectors. (3) Album genres are predicted from the album feature vectors in a shallow network where the input layer is directly connected to the output layer.

We use a similar approach to the one described in Section 8.4 to learn track feature vectors using CNNs. Contant-Q (CQT) spectrograms with log-amplitude scaling are computed from the audio tracks, and patches of 15-seconds long for every track are fed to a CNN. To learn the genre labels we design a CNN with four convolutional layers and experiment with different number of filters, filter sizes, and output configurations (see Section 9.5.1). These networks are trained using mini batches of 32 items, randomly sampled from the training data to compute the gradient, and Adam (Kingma & Ba, 2014) is the optimizer used to train the models, with the default suggested learning parameters.

9.4.2. Text-based approach

In the presented dataset, each album has a variable number of customer reviews. We use an approach similar to the one described in Section 6.3 for genre classification from text, where all reviews from the same album are aggregated into a single text. The aggregated result is truncated at 1,000 characters, thus balancing the amount of text per album, as more popular artists tend to have a higher number of reviews. Then we apply a Vector Space Model approach (VSM) with tf-idf weighting (Zobel & Moffat, 1998) to create a feature vector for each album. Although word embeddings (Mikolov et al., 2013a) with CNNs are state of the art in many text classification tasks (Kim, 2014), a traditional VSM approach is used instead, as it seems to perform better when dealing with large texts (see Section 8.6.2). The vocabulary size is limited to 10k as it was a good balance of network complexity and accuracy.

Furthermore, a second approach is proposed based on the addition of semantic information via entity linking, similarly to the method described in Section 6.3. To semantically enrich the album texts, we adopted Babelfy (Moro et al., 2014b) via ELVIS (see Section 3.3). We take all the Wikipedia categories of entities identified by Babelfy in each document and add them at the end of the text as new words. Then a VSM with tf-idf weighting is applied to the semantically enriched texts, where the vocabulary is also limited to 10k terms. Note that either words or categories may be part of this vocabulary.

From this representation, a feed forward network with two dense layers of 2048 neurons and a Rectified Linear Unit (ReLU) after each layer is trained to predict the genre labels in both LOGISTIC and COSINE configurations. The network is trained also with mini batches of 32 items, and Adam as optimizer.

9.4.3. Image-based approach

Every album in the dataset has an associated cover art image. To perform music genre classification from these images, we use Deep Residual Networks (ResNets) (He et al., 2016). They are the state of the art in various image classification tasks like Imagnet (Russakovsky et al., 2015) and Microsoft COCO (Lin et al., 2014). ResNet is a common feed-forward CNN with *residual learning*, which consists on bypassing two or more convolution layers. We employ a slightly modified version of the original ResNet[39]: the scaling and aspect ratio augmentation are obtained from Szegedy et al. (2015), the photometric distortions from Howard (2013), and weight decay is applied to all weights and biases. The network we use is composed of 101 layers (ResNet-101), initialized with pretrained parameters learned on ImageNet. This is our starting point to fine-tune the network on the genre classification task. Our ResNet implementation has a logistic regression final layer with sigmoid activations and uses the binary cross entropy loss. The network is trained on the genre classification task with mini batches of 50 samples for 90 epochs, a learning rate of 0.0001, and with Adam as optimizer.

9.4.4. Multimodal approach

We aim to combine all of these different types of data into a single model. There are several works claiming that learning data representations from different modalities simultaneously outperforms systems that learn them separately (Ngiam et al., 2011; Dorfer et al., 2016). However, the experiments presented in Section 8.6.3 reflect the contrary. We have observed that deep networks are able to find an optimal minimum very fast from text data. However, the complexity of the audio signal can significantly slow down the training process. Simultaneous learning may under-explore one of the modalities, as the stronger modality may dominate quickly. Thus, learning each modality separately warrants that the variability of the input data is fully represented in each of the feature vectors.

Therefore, from each modality network described above, we separately obtain an internal feature representation for every album after training them on the genre classification task. Concretely, the activations of the last hidden layer of each network become the feature vector for its respective modality. Given

[39] `https://github.com/facebook/fb.resnet.torch/`

a set of feature vectors, $L2$-norm is applied on each of them. They are then concatenated into a single feature vector, which becomes the input to a simple Multi Layer Perceptron (MLP). The input layer of the MLP is directly connected to the output layer, in a similar way to the multimodal network used in Section 8.5. The output layer may have either a logistic or a cosine configuration.

Modality	Target	Settings	Params	Time	AUC	ADiv@1	ADiv@3
Audio	logistic	timbre-mlp	0.01M	1s	0.792	0.04	0.14
Audio	logistic	low-3x3	0.5M	390s	0.859	0.14	0.34
Audio	logistic	high-3x3	16.5M	2280s	0.840	0.20	0.43
Audio	logistic	low-4x96	0.2M	140s	0.851	0.14	0.32
Audio	logistic	high-4x96	5M	260s	0.862	0.12	0.33
Audio	logistic	low-4x70	0.35M	200s	0.871	0.05	0.16
Audio	logistic	high-4x70	7.5M	600s	0.849	0.08	0.23
Audio	cosine	low-3x3	0.33M	400s	0.864	0.26	0.47
Audio	cosine	high-3x3	15.5M	2200s	0.881	0.30	0.54
Audio	cosine	low-4x96	0.15M	135s	0.860	0.19	0.40
Audio	cosine	high-4x96	4M	250s	0.884	0.35	0.59
Audio	cosine	low-4x70	0.3M	190s	0.868	0.26	0.51
Audio (A)	**cosine**	**high-4x70**	**6.5M**	**590s**	**0.888**	**0.35**	**0.60**
Text	logistic	VSM	25M	11s	0.905	0.08	0.20
Text	logistic	VSM+Sem	25M	11s	0.916	0.10	0.25
Text	cosine	VSM	25M	11s	0.901	0.53	0.44
Text (T)	**cosine**	**VSM+Sem**	**25M**	**11s**	**0.917**	**0.42**	**0.70**
Image (I)	logistic	ResNet	1.7M	4009s	0.743	0.06	0.15
A + T	logistic	mlp	1.5M	2s	0.923	0.10	0.40
A + I	logistic	mlp	1.5M	2s	0.900	0.10	0.38
T + I	logistic	mlp	1.5M	2s	0.921	0.10	0.37
A + T + I	**logistic**	**mlp**	**2M**	**2s**	**0.936**	**0.11**	**0.39**
A + T	cosine	mlp	0.3M	2s	0.930	0.43	0.74
A + I	cosine	mlp	0.3M	2s	0.896	0.32	0.57
T + I	cosine	mlp	0.3M	2s	0.919	0.43	0.74
A + T + I	cosine	mlp	0.4M	2s	0.931	0.42	0.72

Table 9.2: Results for Multi-label Music Genre Classification of Albums. Number of network hyperparameters, epoch training time, AUC-ROC, and aggregated diversity at $N = 1, 3$ for different settings and modalities.

9.5. Experiments

We apply the architectures defined in the previous section to the *MuMu* dataset. The dataset is divided as follows: 80% for training, 10% for validation, and 10% for test. All sets contain albums of different artists, to avoid overfitting (Flexer, 2007). We first evaluate every modality in isolation in the multi-label genre classification task. Then, from each modality, a deep feature vector is obtained for the best performing approach in terms of AUC. Finally, the three modality vectors are combined in a multimodal network. All results are reported in Table 9.2. Performance of the classification is reported in terms of AUC score and ADiv@N with $N = 1, 3$. The training speed per epoch and number of network hyperparameters are also reported.

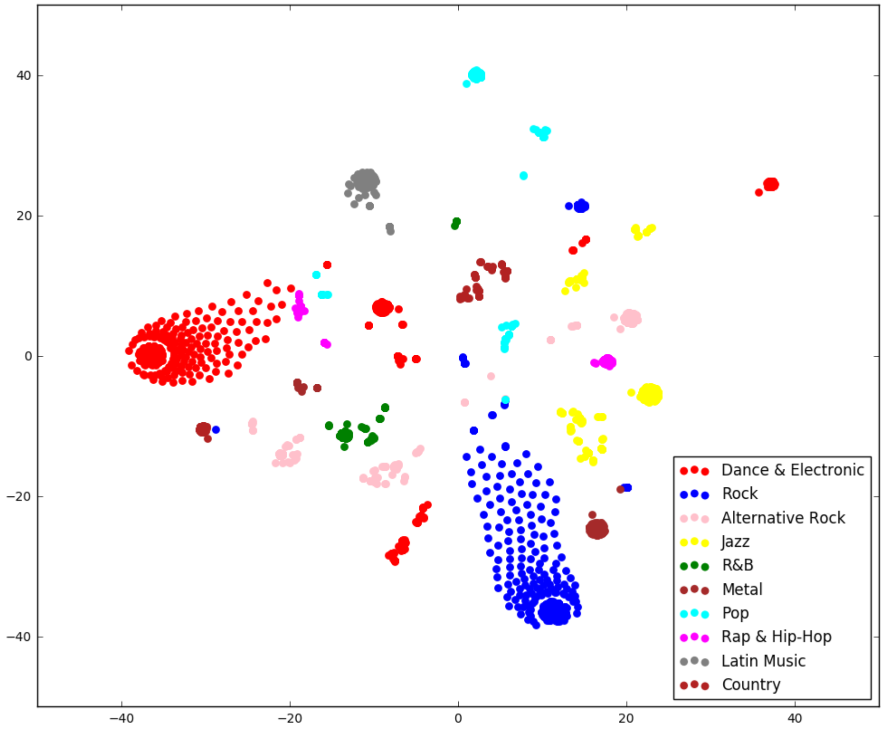


Figure 9.1: t-SNE of album factors.

The matrix of album genre annotations of the training and validation sets is factorized using the approach described in Section 9.3.1, with a value of $d = 50$ dimensions. From the set of album factors, those annotated with a single label from the top level of the taxonomy are plotted in Figure 9.1 using t-SNE dimensionality reduction (Maaten & Hinton, 2008). It can be seen how the different albums are properly clustered in the factor space according to their genre.

9.5.1. Audio classification

We explore three network design parameters: convolution filter size, number of filters per convolutional layer, and target layer. For the filter size we compare three approaches: square 3x3 filters as in Choi et al. (2016a), a filter of 4x96 that convolves only in time (Van den Oord et al., 2013), and a musically motivated filter of 4x70, which is able to slightly convolve in the frequency domain (Pons et al., 2016). To study the width of the convolutional layers we try with two different settings: HIGH with 256, 512, 1024, and 1024 in each layer respectively, and LOW with 64, 128, 128, 64 filters. Max pooling is applied after each convolutional layer. Finally, we use the two different network targets defined in Section 9.3, LOGISTIC and COSINE. We empirically observed that dropout regularization only helps in the HIGH plus COSINE configurations. Therefore we applied dropout with a factor of 0.5 to these configurations, and no dropout to the others.

Apart from these configurations, a baseline approach is added. This approach consists in a traditional audio-based approach for genre classification based on the audio descriptors present in the MSD (Bertin-Mahieux et al., 2011). More specifically, for each song we aggregate four different statistics of the 12 timbre coefficient matrices: mean, max, variance, and $L2$-norm. The obtained 48-dimensional feature vectors are fed into a feed forward network as the one described in Section 9.4.4 with LOGISTIC output. This approach is denoted as TIMBRE-MLP.

The results show that CNNs applied over audio spectrograms clearly outperform traditional approaches based on handcrafted features. We observe that the TIMBRE-MLP approach achieves 0.792 of AUC, contrasting with the 0.888 from the best CNN approach. We note that the LOGISTIC configuration obtains better results when using a lower number of filters per convolution (LOW). Configurations with fewer filters have fewer parameters to optimize, and their training processes are faster. On the other hand, in COSINE configurations we observe that the use of a higher number of filters tends to achieve better performance. It seems that the fine-grained regression of the factors benefits from wider convolutions. Moreover, we observe that 3x3 square filter settings have lower performance, need more time to train, and have a higher number of parameters to optimize. By contrast, networks using time convolutions only (4X96) have a lower number of parameters, are faster to train, and achieve comparable performance. Furthermore, networks that slightly convolve across the frequency bins (4X70) achieve better results with only a slightly higher number of parameters and training time. Finally, we observe that the COSINE regression approach achieves better AUC scores in most configurations, and also their results are more diverse in terms of aggregated diversity.

9.5.2. Text classification

For text classification, we obtain two feature vectors as described in Section 9.4.2: one built from the texts (VSM), and another built from the semantically enriched texts (VSM+SEM). Both feature vectors are trained in the multi-label genre classification task using the two output configurations LOGISTIC and COSINE.

Results show that the semantic enrichment of texts clearly yields better results in terms of AUC and diversity. Furthermore, we observe that the COSINE con-

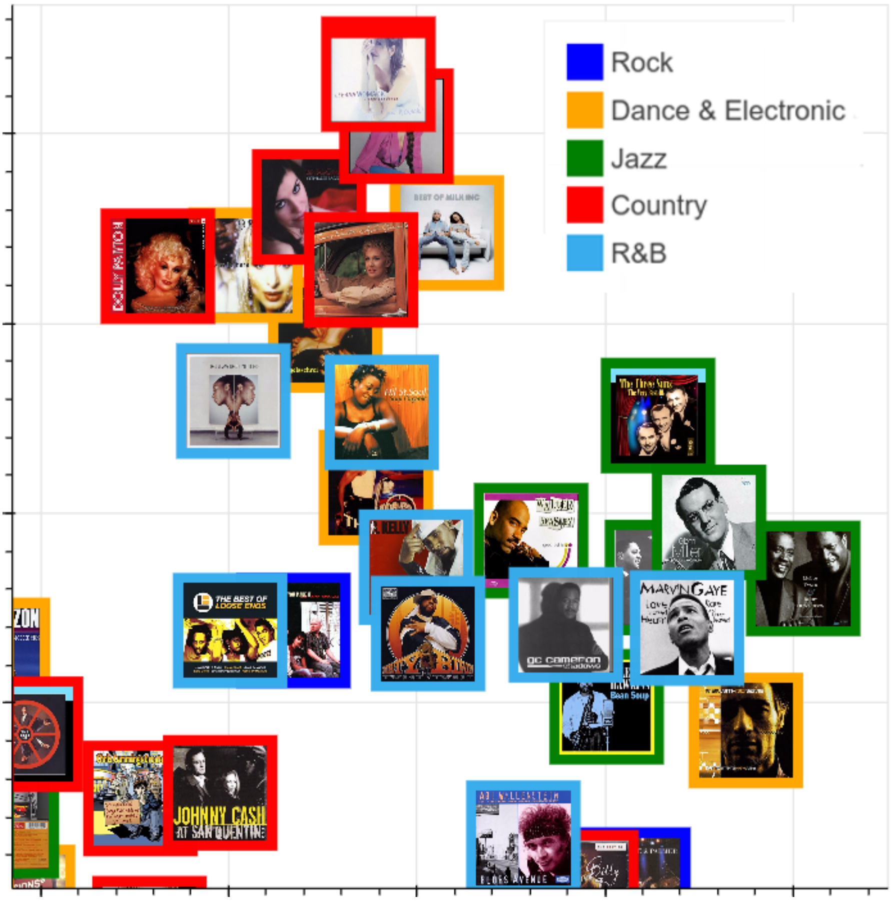


Figure 9.2: Particular of the t-SNE of randomly selected image vectors from five of the most frequent genres.

figuration slightly outperforms LOGISTIC in terms of AUC, and greatly in terms of aggregated diversity. The text-based results are overall slightly superior to the audio-based ones.

We also studied the information gain of words in the different genres. We observed that genre labels present inside the texts have high information gain values. It is also remarkable that *band* is a very informative word for Rock, *song* for Pop, and *dope*, *rhymes*, and *beats* are discriminative features for Rap albums. Place names have also important weights, as *Jamaica* for Reggae, *Nashville* for Country, or *Chicago* for Blues.

9.5.3. Image classification

Results show that genre classification from images has lower performance in terms of AUC and aggregated diversity compared to the other modalities. Due to the use of an already pre-trained network with a logistic output (ImageNet Russakovsky et al. (2015)) as initialization of the network, it is not straightforward to apply the COSINE configuration. Therefore, we only report results for the LOGISTIC configuration.

In Figure 9.2 a set of cover images of five of the most frequent genres in the dataset is shown using t-SNE over the obtained image feature vectors. In the left top corner the ResNet recognizes women faces on the foreground, which seems to be common in Country albums (red). The jazz albums (green) on the right are all clustered together probably thanks to the uniform type of clothing worn by the people of their covers. Therefore, the visual style of the cover seems to be informative when recognizing the album genre. For instance, many classical music albums include an instrument in the cover, and Dance & Electronics covers are often abstract images with bright colors, rarely including human faces.

9.5.4. Multimodal classification

From the best performing approaches in terms of AUC of each modality (i.e., AUDIO/COSINE/HIGH-4X70, TEXT/COSINE/VSM+SEM, and IMAGE/LOGISTIC/RESNET), an internal feature representation is obtained as described in Section 9.4.4. Then, these three feature vectors are aggregated in all possible combinations, and genre labels are predicted using the MLP network described in Section 9.4.4. Both output configurations LOGISTIC and COSINE are used in the learning phase, and dropout of 0.7 is applied in the COSINE configuration.

Results, as shown at the bottom of Table 9.2, suggest that the combination of modalities outperforms single modality approaches. As image features are learned using a LOGISTIC configuration, they seem to improve multimodal approaches with LOGISTIC configuration only. Multimodal approaches that include text features tend to achieve better results. Nevertheless, the best approaches are those that exploit the three modalities of *MuMu*. COSINE approaches have similar AUC than LOGISTIC approaches but a much better aggregated diversity, thanks to the spatial properties of the factors space.

9.6. Conclusions

An approach for multi-label music genre classification using deep learning architectures has been proposed. The approach was applied to audio, text, and image data, and to the combination of learned data representations. For its assessment, *MuMu*, a new multimodal music dataset with over 31k albums and 147k songs has been gathered. We showed how representation learning approaches for audio classification outperform traditional handcrafted feature based approaches. Moreover, we compared the effect of different design parameters of CNNs in audio classification. Text-based approaches seem to outperform other modalities, and benefit from the semantic enrichment of texts via entity linking. While the image-based classification yielded the lowest performance, it helped to improve the results when combined with other modalities. Multimodal approaches appear to outperform single modality approaches, and the aggregation of the three modalities achieved the best results. Furthermore, the dimensionality reduction of target labels led to better results, not only in terms of accuracy, but also in terms of aggregated diversity.

The work in this chapter is an initial attempt to study the multi-label classification problem of music genres from different perspectives and using different data modalities. In addition, the release of the *MuMu* dataset opens up a number of unexplored research possibilities.

CHAPTER 10

Summary and future perspectives

10.1. Introduction

When the work for this thesis started, there was almost no published literature related with the extraction of high-level semantic representations from unstructured music-related texts. Nevertheless, in the context of MIR a growing number of research works had been published exploiting either user-generated texts (Celma et al., 2006; Lamere, 2008; Whitman & Lawrence, 2002; Knees & Schedl, 2013) or knowledge bases (Sordo et al., 2010; Celma, 2006; Passant & Decker, 2010; Ostuni et al., 2013). Initial attempts to apply knowledge extraction techniques to the music domain (Tata & Di Eugenio, 2010; Knees & Schedl, 2011; Sordo et al., 2012), showed the epistemic potential of text for music applications. In this thesis we have followed these ideas, deepening in the linguistic processing applied to extract the information, and proposing new approaches that exploit the extracted information in MIR tasks such as music recommendation and classification. In addition, we have combined extracted semantic information with content from other data modalities such as audio and images using deep neural networks. New data representations learned from the different data modalities and their combination have shown to outperform traditional handcrafted audio features and single modality approaches.

We started this thesis motivating and framing the work carried out with an introduction to knowledge extraction and representation learning in the context of Music Information Retrieval (MIR). In addition, we introduced the music recommendation and classification tasks (Chapter 1). We continued by illustrating some background concepts related to Natural Language Processing (NLP), and summarizing the existing literature on text-based, knowledge-based, and deep learning approaches for Recommender Systems and MIR. Then, we described a framework for entity linking and the creation of a large corpus of annotated musical entities (Chapter 3). Next, we proposed a method for extracting semantic relations between musical entities present in unstructured texts, and we evaluated the suitability of extracted knowledge to provide explanations for music recommendations (Chapter 4). Two use cases on the applications of knowledge extraction for musicological studies were described next (Chapter 5). Then, we presented an approach for the semantic enrichment of musical texts via entity linking, which was applied to artist similarity and music genre classification (Chapter 6). Next, a similar idea was further developed and combined with user feedback data in the context of a hybrid music recommendation approach (Chapter 7). Finally, we described an approach to learn novel data representations from multimodal content using deep neural networks. This approach was then applied to the problems of cold-start music recommendation (Chapter 8), and multi-label music genre classification (Chapter 9).

In each chapter, we provided a summary of the conclusions and relevant results of the corresponding work. In what follows, we enumerate the main contribution of this thesis. Finally, we end this thesis with a discussion about future research directions.

10.2. Summary of contributions

In this thesis, we have focused on the problem of recommending and classifying musical items in large music collections applying two different approaches: (i) an approach based on the extraction of structured knowledge from unstructured texts and its further enrichment using semantic information coming from online knowledge repositories, (ii) an approach based on representation learning from multimodal data using deep learning architectures. We now present a summary of the main contributions of this thesis.

10.2.1. Scientific contributions

1. A comprehensive review of current approaches in Natural Language Processing, Recommender Systems, and Music Information Retrieval, with a special focus on entity linking, knowledge base creation, relation extraction, artist similarity, music classification, and music recommendation (Chapter 2).

2. An approach for the automatic creation of music knowledge bases from unstructured texts, which encodes semantic relations among musical entities by leveraging syntactic and semantic information (Chapter 4). The approach has the following advantages:

   a) It is able to capture a highly precise and compact set of weighted triples thanks to a clustering method and a novel scoring metric.
   b) Given a proper text corpora, it is able to extract knowledge not present in other knowledge bases, both general and domain-specific.
   c) The extracted knowledge base is suitable to provide explanations for music recommendations.

3. An exploratory study on how knowledge extraction techniques may impact musicological studies (Chapter 5), which has produced the following outcomes:

   a) An approach for the creation of culture-specific music knowledge bases, which combines structured information coming from different data sources and information extracted from unstructured texts.
   b) A methodology to build knowledge graphs from unstructured texts suitable for computing artist's relevance.
   c) A method to extract and analyze the sentiment polarity expressed in music reviews, which is used to study the evolution of music genres and affective language.

4. A methodology for the semantic enrichment of unstructured text documents with information present in online knowledge repositories. Enriched text representations are further exploited in artist similarity and music classification tasks, outperforming traditional text-based approaches (Chapter 6).

5. An extension of the previous contribution for the creation of knowledge graphs from tags and items descriptions leveraging semantic information. These graphs are in turn exploited together with user feedback information in a hybrid recommendation approach. An extensive evaluation shows improvements with respect to state-of-the-art collaborative filtering algorithms, in terms of prediction accuracy, novelty, and aggregated diversity (Chapter 7).

6. An approach for providing recommendations of novel artists and songs, combining audio tracks, semantically enriched artist biographies, and user feedback information using deep neural networks (Chapter 8). The proposed approach benefits from the late fusion of data representations learned separately.

7. A methodology for the classification of musical items with multiple genre labels using audio, text, semantic information, and images, where novel data representations are learned using deep neural networks and further

combined in a multimodal approach. Moreover, classification accuracy and aggregated diversity are improved by applying dimensionality reduction of target labels through matrix factorization techniques (Chapter 9).

10.2.2. Datasets

Due to the fact that appropriate datasets for the evaluation of our methods have not been always available, we have dedicated an important effort in gathering and curating new datasets. These are our contributions in terms of datasets.

1. Novel dataset of ∼13k documents and almost 150k annotated musical entities, which are linked to DBpedia and MusicBrainz. From this corpus, a gold standard dataset of 200 documents with manually annotated entities is also created (Section 3.4).

2. Large dataset of about 64k albums with customer reviews, acoustic features per track, metadata, and single-label genre annotations (Sections 5.3.1 and 6.3.1).

3. Two datasets of 188 and 2,336 artist biographies respectively, together with artist similarity ground truth data (Section 6.2.4).

4. Two datasets of tags and text descriptions about musical items, together with user feedback information on those items. A dataset of sounds with ∼21k items and 20k users, and a dataset of songs with ∼8.5k items and ∼5k users (Section 7.4.1).

5. Dataset of ∼24k artist biographies linked to the artists present in the Million Song Dataset (Section 8.6.1).

6. Large dataset of about ∼31k albums, with ∼450k customer reviews, ∼147k audio tracks, cover artworks, and multi-label genre annotations (Section 9.2).

10.2.3. Knowledge bases

1. Knowledge base of popular music extracted from a corpus of ∼32k documents with stories about songs (Section 4.3.2).

2. Knowledge base of flamenco music, created by combining data from 7 different data sources, and enriched with information extracted from ∼1k artist biographies (Section 5.2.2).

10.2.4. Software

1. A system that integrates different entity linking tools, enriching their output and providing high confident entity disambiguations.

2. A system to perform and evaluate deep learning experiments on classification and recommendation from different data modalities and their combination.

10.2.5. Publications

The research carried out in this thesis has been published in several peer-reviewed journals and top international conferences. Parts of the research presented in Chapter 3 have been published in a conference paper (Oramas et al., 2016b). The work described in Chapter 4 has been published in a conference and a journal paper (Oramas et al., 2015b, 2016c). The parts of the research presented in Chapter 5 related with the creation of domain-specific knowledge bases have been published in a conference and a journal paper (Oramas et al., 2015a; Oramas & Sordo, 2016), and those related with the diachronic study of music reviews were published in another conference paper (Oramas et al., 2016a). Similarly, the parts of the research presented in Chapter 6 related with artist similarity have been published in a conference paper (Oramas et al., 2015c), and those related with music genre classification have been published also in Oramas et al. (2016a). Furthermore, the outcomes of Chapter 7 have been published in a journal paper (Oramas et al., 2016d). Finally, the work described in Chapter 8 has been published in a conference paper (Oramas et al., 2017b), and the outcomes of the research carried out in Chapter 9 have been published in another conference paper (Oramas et al., 2017a). The full list of author's publications related to the work presented in this thesis is available in Appendix A, and the full list of released datasets, knowledge bases, and software is available in Appendix B.

10.3. Directions for future research

In the present thesis we have tried to help machines to better understand what people say about music, and we have shown how to combine semantic knowledge, texts, user feedback, audio, and images in the context of MIR and computational musicology. This is an exploratory work that opens up a number of research possibilities for text-based and multimodal approaches in the music domain. In what follows, we enumerate a series of ideas for future work related with the different tasks addressed in this thesis.

Entity linking As observed in Chapters 3 and 4, the identification and classification of music entities in text is a problem far from being solved. Current systems make an important number of mistakes and do not operate on music-specific knowledge bases. Instead, current systems use general-purpose ones such as DBpedia or BabelNet. As availability of music entities in these knowledge bases is scarce, there is a need for an entity linking system able to recognize and disambiguate musical entities to a music knowledge base (e.g., MusicBrainz). We envision that splitting the problem into recognition and disambiguation may improve the precision. An entity recognizer trained with music-specific corpora would benefit from the textual context of the entities to properly classify them. Then, categories identified by the recognizer would be used in the disambiguation step to improve the precision of linking. To this end, the creation of large datasets of annotated musical entities is a necessary step.

Construction of knowledge bases In Chapter 4 we have explored the automatic creation of music knowledge bases using an approach based on the combination of open information extraction and entity linking. However, other approaches may be used for relation extraction. In the MusicBrainz database, a large number of relations between entities are encoded together with information about the lexicalization of these relations. This is a highly valuable resource that can be exploited, for instance, in distant supervision approaches. Additionally, the creation of an open music knowledge base that constantly reads from the web, like the Never-Ending Language Learning system (NELL) (Carlson et al., 2010a), would create a highly valuable resource, not only for research, but also for commercial applications.

Other NLP techniques In this work we have explored the application of several NLP techniques and tasks to the music domain. Nevertheless, among the tasks not explored in this thesis, we may highlight Question & Answering, which is a challenging problem that also deals with semantic representations of text. Question & Answering systems or chat bots may have several applications to the music domain, such as knowledge dissemination, promotion of artists, or music recommendation. Big companies are currently working on their own conversational systems. Knowledge bases have been traditionally exploited by these systems, and more recently, deep learning approaches using RNNs and memory networks have shown promising results learning directly from conversations (Sukhbaatar et al., 2015). Moreover, other deep learning techniques such as reinforcement learning have shown the potential of combining knowledge bases and deep learning in conversational systems (Andreas et al., 2016).

Musicology In Chapter 5 we left some hypotheses open about the evolution of the language used in music reviews. To demonstrate any of these hypotheses is a challenging problem. In addition, a more thorough study on the evolution of music genres could be done over the compiled dataset. We have shown how knowledge extraction techniques may facilitate musicologists' work. Therefore, the creation of specific tools to process large amounts of musicological documents, either in music digital libraries or private collections, is an open research direction.

Deep learning for text Word vector embeddings have been shown to be very useful in most NLP tasks, and they have been widely exploited in deep learning approaches (Collobert et al., 2011). Hence, further exploration on architectures that exploit the potential of these word representations in MIR tasks is a clear research direction. Additionally, combining lexical semantics encoded in word vectors and explicit semantics encoded in knowledge bases is another open research direction. Novel techniques, such as retrofitting (Faruqui et al., 2014), go in this direction. In addition, recent developments, such as attention-based networks (Lin et al., 2017), could be also applied in the context of our research.

Multimodal deep learning The multimodal deep learning approach presented in this thesis, is based on the late fusion of learned data representations. We have shown how this approach outperforms simultaneous learning from text and audio. However, an intermediate way would be to learn data representations separately and try to fine-tune the whole multimodal network in the final task, becoming a fully end-to-end learning approach.

Music classification In Chapter 9 we have shown how data representations can be learned in a multi-label genre classification task. Given the high granularity of the genre annotations, learned features encode fine-grained properties of the data. Therefore, they might be exploited in other applications via transfer learning, such as music similarity or music recommendation.

Music recommendation A neural model may be learned to embed data representations from different modalities in a new multimodal space that better optimizes their similarity. Mapping multimodal data representations in a common space may be useful to pass from one modality to another. One application of this may be, for instance, going from a text description or a photo to a set of audio tracks, giving rise to new ways for playlist generation.

Generative models Another interesting line of research we envision are generative models based on multimodal data. Generation of audio from text descriptions, text descriptions from audio, or album cover artwork from album tracks are some of the possible applications. Similar approaches have already been developed between texts and images. However, the generation of/from audio has received less attention.

All in all, the writing of this thesis has been an exciting path through different ways of incorporating human and machine representations of musical knowledge into computational systems.

Sergio Oramas Martín, Barcelona, 21 September 2017.

Bibliography

Abdollahpouri, H., Burke, R., & Mobasher, B. (2017). Controlling popularity bias in learning to rank recommendation. In *Proceedings of the 11th ACM Conference on Recommender Systems.*

Adomavicius, G. & Kwon, Y. (2012). Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering*, *24*(5), 896–911.

Alcorta, C. S., Sosis, R., & Finkel, D. (2008). Ritual harmony: Toward an evolutionary theory of music. *Behavioral and Brain Sciences*, *31*(5), 576–577.

Alleyne, M. R. & Dunbar, S. (2012). *The Encyclopedia of Reggae: The Golden Age of Roots Reggae.* Sterling.

Anand, S. S., Kearney, P., & Shapcott, M. (2007). Generating semantically enriched user profiles for Web personalization. *ACM Transactions on Internet Technology*, *7*(4), 22.

Anderson, C. (2006). *The long tail: Why the future of business is selling less of more.* Hachette Books.

Andreas, J., Rohrbach, M., Darrell, T., & Klein, D. (2016). Learning to compose neural networks for question answering. *arXiv preprint arXiv:1601.01705.*

Bach, N. & Badaskar, S. (2007). A review of relation extraction. *Literature review for Language and Statistics II*, *2*.

Ballesteros, M. & Nivre, J. (2013). Going to the roots of dependency parsing. *Computational Linguistics*, *39*(1), 5–13.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open information extraction for the web. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, vol. 7, pp. 2670–2676.

Bansal, T., Belanger, D., & McCallum, A. (2016). Ask the gru: Multi-task learning for deep text recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pp. 107–114. ACM.

Baral, C. & De Giacomo, G. (2015). Knowledge representation and reasoning: What's hot. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 4316–4317.

Beliakov, G., Calvo, T., & James, S. (2015). Aggregation functions for recommender systems. In F. Ricci, L. Rokach, & B. Shapira (Eds.) *Recommender Systems Handbook*, pp. 777–808. Springer US.

Bello, J. P. & Pickens, J. (2005). A robust mid-level representation for harmonic content in music signals. In *Proceedings of the 6th International Society for Music Information Retrieval Conference ISMIR*, pp. 304–311.

Bellogín, A., Cantador, I., & Castells, P. (2010). A Study of Heterogeneity in Recommendations for a Social Music Service. In *Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, HetRec '10, pp. 1–8. ACM.

Bellomi, F. & Bonato, R. (2005). Network analysis for Wikipedia. In *Proceedings of Wikimania.*

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798–1828.

Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A neural probabilistic language model. *The Journal of Machine Learning Research*, *3*, 1137–1155.

Bertin-Mahieux, T., Eck, D., Maillet, F., & Lamere, P. (2008). Autotagger: A model for predicting social tags from acoustic features on large music databases. *Journal of New Music Research*, *37*(2), 115–135.

Bertin-Mahieux, T., Ellis, D. P. W., Whitman, B., & Lamere, P. (2011). The Million Song Dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR).*

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., & Hellmann, S. (2009). DBpedia - A crystallization point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, *7*(3), 154–165.

Blas Vega, J. & Ríos Ruiz, M. (1988). *Diccionario enciclopédico ilustrado del flamenco.* Madrid: Cinterco.

Bogdanov, D., Haro, M., Fuhrmann, F., Xambó, A., Gómez, E., & Herrera, P. (2013a). Semantic audio content-based music recommendation and visualization based on user preference examples. *Information Processing & Management*, *49*, 13–33.

Bogdanov, D. & Herrera, P. (2011). How much metadata do we need in music recommendation? a subjective evaluation using preference sets. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*.

Bogdanov, D., Porter, A., Herrera, P., & Serra, X. (2016). Cross-collection evaluation for music classification tasks. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*.

Bogdanov, D., Wack, N., & Others (2013b). ESSENTIA: An open-source library for sound and music analysis. In *ACM International Conference on Multimedia*, pp. 855–858.

Bohnet, B. (2010). Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pp. 89–97. Association for Computational Linguistics.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008). Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pp. 1247–1250. ACM.

Bonnin, G. & Jannach, D. (2014). Automated generation of music playlists: survey and experiments. *ACM Comput. Surv.*, *47*(2), 26:1–26:35.

Bouayad-Agha, N., Burga, A., Casamayor, G., Codina, J., Nazar, R., & Wanner, L. (2014). An exercise in reuse of resources: Adapting general discourse coreference resolution for detecting lexical chains in patent documentation. In *Proccedings of the Language Resources and Evaluation 775 Conference (LREC)*, pp. 3214–3221.

Bovi, C. D., Espinosa-Anke, L., & Navigli, R. (2015a). Knowledge base unification via sense embeddings and disambiguation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 726–736.

Bovi, C. D., Telesca, L., & Navigli, R. (2015b). Large-scale information extraction from textual definitions through deep syntactic and semantic analysis. *Transactions of the Association for Computational Linguistics (TACL)*, *3*, 529–543.

Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, *30*, 107–117.

Bunescu, R. & Pasca, M. (2006). Using encyclopedic knowledge for named entity disambiguation. In *Proceesings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06)*, pp. 9–16.

Bunescu, R. C. & Mooney, R. J. (2005). A Shortest Path Dependency Kernel for Relation Extraction. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pp. 724–731.

Bunke, H. & Shearer, K. (1998). A graph distance metric based on the maximal common subgraph. *Pattern Recognition Letters*, *19*(3-4), 255–259.

Burke, R. (2002). Hybrid recommender systems: survey and experiments. *User Modeling and User-Adapted Interaction*, *12*(4), 331–370.

Cambria, E. & White, B. (2014). Jumping NLP curves: A review of natural language processing research. *Computational Intelligence Magazine, IEEE*, *9*(2), 48–57.

Cantador, I., Bellogín, A., & Castells, P. (2008). A multilayer ontology-based hybrid recommendation model. *AI Communications, Special Issue on Recommender Systems*, *21*(2-3), 203–210.

Carlson, A., Betteridge, J., & Kisiel, B. (2010a). Toward an architecture for never-ending language learning. *In Proceedings of the Conference on Artificial Intelligence (AAAI) (2010)*, pp. 1306–1313.

Carlson, A., Betteridge, J., Wang, R. C., Hruschka Jr, E., & Mitchell, T. M. (2010b). Coupled Semi-Supervised Learning for Information Extraction. In *Proceedings of the third ACM International Conference on Web Search and Data Mining (WSDM)*, pp. 101–110.

Casey, M. A., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., & Slaney, M. (2008). Content-based music information retrieval: Current directions and future challenges. *Proceedings of the IEEE*, *96*(4), 668–696.

Celma, Ò. (2006). Foafing the music: Bridging the semantic gap in music recommendation. In *Proceedings of 5th International Semantic Web Conference*, pp. 927–934.

Celma, Ò. (2010). *Music Recommendation and Discovery - The Long Tail, Long Fail, and Long Play in the Digital Music Space.* Springer.

Celma, Ò., Cano, P., & Herrera, P. (2006). Search Sounds: An audio crawler focused on weblogs. In *7th International Conference on Music Information Retrieval (ISMIR).*

Celma, Ò. & Herrera, P. (2008). A new approach to evaluating novel recommendations. In *Proceedings of the 2008 ACM conference on Recommender systems*, pp. 179–186. ACM.

Chim, H. & Deng, X. (2008). Efficient phrase-based document similarity for clustering. *Knowledge and Data Engineering, IEEE Transactions on*, *20*(9), 1217–1229.

Choi, K., Fazekas, G., & Sandler, M. (2016a). Automatic tagging using deep convolutional neural networks. *Proceedings of the 17th International Society for Music Information Retrieval Conference*, pp. 805–811.

Choi, K., Fazekas, G., Sandler, M., & Cho, K. (2016b). Convolutional recurrent neural networks for music classification. *arXiv preprint arXiv:1609.04243*.

Choi, K., Lee, J. H., & Downie, J. S. (2014). What is this song about anyway?: Automatic classification of subject using user interpretations and lyrics. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, pp. 453–454.

Chollet, F. (2016). Information-theoretical label embeddings for large-scale image classification. *CoRR*, pp. 1–10.

Cohen, W. W. & Fan, W. (2000). Web-collaborative filtering: recommending music by crawling the Web. *Computer Networks*, *33*, 685–698.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, *12*, 2493–2537.

Cornolti, M., Informatica, D., Ferragina, P., & Ciaramita, M. (2013). A framework for benchmarking entity-annotation systems. *Proceedings of the International World Wide Web Conference (WWW) (Practice & Experience Track), ACM (2013)*.

Corona, H. & O'Mahony, M. P. (2015). An exploration of mood classification in the million songs dataset. In *Proceedings of the 12th Sound and Music Computing Conference*.

Cowie, J. & Lehnert, W. (1996). Information extraction. *Communications of the ACM*, *39*(1), 80–91.

Crawford, T., Fields, B., Lewis, D., & Page, K. (2014). Explorations in Linked Data practice for early music corpora. In *Proceedings of the 14th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 309–312.

Culotta, A. & Sorensen, J. (2004). Dependency tree kernels for relation extraction. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL).*

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. (1990). Indexing by latent semantic analysis. *JAsIs*, *41*(6), 391–407.

Di Noia, T., Mirizzi, R., Ostuni, V. C., & Romito, D. (2012a). Exploiting the web of data in model-based recommender systems. In *Proceedings of the Sixth ACM Conference on Recommender Systems*, RecSys '12, pp. 253–256. ACM.

Di Noia, T., Mirizzi, R., Ostuni, V. C., Romito, D., & Zanker, M. (2012b). Linked open data to support content-based recommender systems. In *Proceedings of the 8th International Conference on Semantic Systems*, I-SEMANTICS '12, pp. 1–8. ACM.

Dieleman, S., Brakel, P., & Schrauwen, B. (2011). Audio-based music classification with a pretrained convolutional network. In *12th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 669–674.

Dieleman, S. & Schrauwen, B. (2014). End-to-end learning for music audio. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 6964–6968. IEEE.

Dong, R., O'Mahony, M. P., & Smyth, B. (2014). Further experiments in opinionated product recommendation. In *Proceedings of the 22th International Conference on Case-Based Reasoning ICCBR'14*, pp. 110–124.

Dong, R., Schaal, M., O'Mahony, M. P., & Smyth, B. (2013). Topic extraction from online reviews for classification and recommendation. Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), pp. 1310–1316.

Dorfer, M., Arzt, A., & Widmer, G. (2016). Towards score following in sheet music images. *Proceedings of the 17th International Society of Music Information Retrieval Conference (ISMIR).*

Downie, J. S. & Hu, X. (2006). Review mining for music digital libraries: phase II. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries*, p. 196.

Eck, D., Lamere, P., Bertin-Mahieux, T., & Green, S. (2008). Automatic generation of social tags for music recommendation. In *Advances in Neural Information Processing Systems 20*, pp. 385–392. MIT Press.

Ellis, D. P. W., Ellis, D. P., Whitman, B., Berenzweig, A., & Lawrence, S. (2002). The quest for ground truth in musical artist similarity. In *Proceedings International Symposium on Music Information Retrieval (ISMIR 2002)*, pp. 170–177.

Esuli, A. & Sebastiani, F. (2006). SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, vol. 6, pp. 417–422. Citeseer.

Fader, A., Soderland, S., & Etzioni, O. (2011). Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing EMNLP '11*, pp. 1535–1545.

Faruqui, M., Dodge, J., Jauhar, S. K., Dyer, C., Hovy, E., & Smith, N. A. (2014). Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*.

Fernández, L. (2004). *Teoría musical del flamenco*. Madrid: Acordes Concert.

Fernández-Tobías, I., Cantador, I., Kaminskas, M., & Ricci, F. (2011). A generic semantic-based framework for cross-domain recommendation. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, HetRec '11, pp. 25–32. ACM.

Ferragina, P. & Scaiella, U. (2010). Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pp. 1625–1628. ACM.

Ferragina, P. & Scaiella, U. (2012). Fast and accurate annotation of short texts with Wikipedia pages. *Software, IEEE*, *29*(1).

Fields, B., Casey, M. A., Jacobson, K., Sandler, M. B. et al. (2008). Do you sound like your friends? exploring artist similarity via artist social network relationships and audio signal processing. In *Proceedings of the International Computer Music Conference (ICMC)*.

Finkel, J. R., Grenager, T., & Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pp. 363–370.

Flexer, A. (2007). A closer look on artist filters for musical genre classification. In *Proceedings of the 8th International Society for Music Information Retrieval Conference*.

Font, F. (2015). Tag recommendation using folksonomy information for online sound sharing platforms. *PhD dissertation, Universitat Pompeu Fabra.*

Font, F., Roma, G., Herrera, P., & Serra, X. (2012). Characterization of the Freesound online community. In *Proceedings of the 3rd International Workshop on Cognitive Information Processing (CIP)*, pp. 1–6.

Gale, W. A., Church, K. W., & Yarowsky, D. (1992). One sense per discourse. In *Proceedings of the workshop on Speech and Natural Language*, pp. 233–237. Association for Computational Linguistics.

Gamallo, P., Garcia, M., & Fernández-Lanza, S. (2012). Dependency-based Open Information Extraction. In *Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP*, ROBUS-UNSUP '12, pp. 10–18.

Gamboa, J. M. (2005). *Una historia del flamenco*. Madrid: Espasa-Calpe.

Gangemi, A. (2013). A comparison of knowledge extraction tools for the semantic web. In *Extended Semantic Web Conference*, pp. 351–366. Springer.

Gantner, Z., Drumond, L., Freudenthaler, C., Rendle, S., & Schmidt-Thieme, L. (2010). Learning attribute-to-feature mappings for cold-start recommendations. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, ICDM '10, pp. 176–185.

Garcia-Silva, A., Corcho, O., Alani, H., & Gomez-Perez, A. (2012). Review of the state of the art: discovering and associating semantics to tags in folksonomies. *The Knowledge Engineering Review*, *27*(01), 57–85.

Getoor, L. (2012). Entity Resolution : Theory , Practice & Open Challenges. *Tutorial at AAAI-12*, pp. 2018–2019.

Gracy, K. F., Zeng, M. L., & Skirvin, L. (2013). Exploring Methods To Improve Access to Music Resources by Aligning Library Data With Linked Data : A Report of Methodologies and Preliminary Findings. *Journal of the Association for Information Science and Technology (JASIST)*, *64*(10), 2078–2099.

Gruhl, D., Nagarajan, M., Pieper, J., Robson, C., & Sheth, A. (2009). Context and domain knowledge enhanced entity spotting in informal text. In *The Semantic Web-ISWC*, pp. 260–276. Springer.

Gülçehre, Ç. & Bengio, Y. (2016). Knowledge matters: Importance of prior information for optimization. *Journal of Machine Learning Research*, *17*(8), 1–32.

Hauger, D., Schedl, M., Košir, A., & Tkalcic, M. (2013). The million musical tweets dataset: What can we learn from microblogs. In *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013)*.

Havasi, C., Speer, R., & Alonso, J. (2007). ConceptNet 3: A flexible, multilingual semantic network for common sense knowledge. In *Proceedings of Recent Advances in Natural Language Processing*, pp. 27–29. Citeseer.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778.

Heitmann, B. & Hayes, C. (2010). Using Linked Data to Build Open, Collaborative Recommender Systems. In *AAAI Spring Symposium: Linked Data Meets Artificial Intelligence*.

Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2015). Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

Ho, C.-H. & Lin, C.-J. (2012). Large-scale linear support vector regression. *Journal of Machine Learning Research*, *13*, 3323–3348.

Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., & Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. *Network*, pp. 541–550.

Howard, A. G. (2013). Some improvements on deep convolutional neural network based image classification. *arXiv preprint arXiv:1312.5402*.

Hu, M. & Liu, B. (2004). Mining opinion features in customer reviews. In *AAAI'04*, pp. 755–760.

Hu, X. & Downie, J. (2006). Stylistics in customer reviews of cultural objects. *SIGIR Forum*, pp. 49–51.

Hu, X., Downie, J., West, K., & Ehmann, A. (2005). Mining music reviews: Promising preliminary results. In *Proceedings of the 6th International Society of Music Information Retrieval Conference (ISMIR)*, pp. 536–539.

Hu, X., Zhang, X., Lu, C., Park, E. K., & Zhou, X. (2009). Exploiting Wikipedia as external knowledge for document clustering. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 389–396. ACM.

Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pp. 263–272.

Humphrey, E. J., Bello, J. P., & LeCun, Y. (2012). Moving beyond feature design: Deep architectures and automatic feature learning in music informatics. In *Proceedings of the 13th International Society for Music Information Retrieval Conference ISMIR*, pp. 403–408.

Iacobacci, I., Pilehvar, M. T., & Navigli, R. (2015). SensEmbed: Learning sense embeddings for word and relational similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Ioffe, S. & Szegedy, C. (2015). Batch Normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, *abs/1502.03167*.

Jain, H., Prabhu, Y., & Varma, M. (2016). Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 935–944. ACM.

Jeh, G. & Widom, J. (2002). SimRank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 538–543. ACM.

Jiang, J. & Zhai, C. (2007). A systematic exploration of the feature space for relation extraction. In *HLT-NAACL*, pp. 113–120.

Juslin, P. N. & Västfjäll, D. (2008). Emotional responses to music: the need to consider underlying mechanisms. *The Behavioral and brain sciences*, *31*(5), 559–621.

Kaminskas, M. & Ricci, F. (2012). Contextual music information retrieval and recommendation: State of the art and challenges. *Computer Science Review*, *6*(2-3), 89–119.

Kim, J. H., Tomasik, B., & Turnbull, D. (2009). Using artist similarity to propagate semantic information. In *Proceedings of the 10th International Society of Music Information Retrieval Conference (ISMIR)*, vol. 9, pp. 375–380.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pp. 1746–1751.

Kingma, D. P. & Ba, J. (2014). Adam: A Method for Stochastic Optimization. *CoRR*, *abs/1412.6980*.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, *46*, 604–632.

Knees, P. & Schedl, M. (2011). Towards Semantic Music Information Extraction from the Web Using Rule Patterns and Supervised Learning. p. 18.

Knees, P. & Schedl, M. (2013). A survey of music similarity and recommendation from music context data. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, *10*(1).

Koduri, G. K. (2014). Culture-aware approaches to modeling and description of intonation using multimodal data. In *EKAW (Satellite Events)*, pp. 209–217.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems. *Computer*, *42*(8), 42–49.

Lamere, P. (2008). Social tagging and music information retrieval. *Journal of new music research*, *37*(2), 101–114.

Larochelle, H., Bengio, Y., Louradour, J., & Lamblin, P. (2009). Exploring strategies for training deep neural networks. *Journal of Machine Learning Research*, *10*(Jan), 1–40.

Laurier, C., Grivolla, J., & Herrera, P. (2008). Multimodal music mood classification using audio and lyrics. In *Machine Learning and Applications, 2008. ICMLA'08. Seventh International Conference on*, pp. 688–693. IEEE.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., & Bizer, C. (2014). {DBpedia} - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*.

Levy, O. & Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*, pp. 2177–2185.

Libeks, J. & Turnbull, D. (2011). You can judge an artist by an album cover: Using images for music annotation. *IEEE MultiMedia*, *18*(4), 30–37.

Liem, C., Müller, M., Eck, D., Tzanetakis, G., & Hanjalic, A. (2011). The need for music information retrieval with user-centered and multimodal strategies. In *Proceedings of the 1st international ACM workshop on Music information retrieval with user-centered and multimodal strategies*, pp. 1–6. ACM.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pp. 740–755. Springer.

Lin, Z., Feng, M., Santos, C. N. d., Yu, M., Xiang, B., Zhou, B., & Bengio, Y. (2017). A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130.*

Liu, H. & Wang, P. (2014). Assessing Text Semantic Similarity Using Ontology. *Journal of Software*, *9*(2), 490–497.

Logan, B. & Ellis, D. P. W. (2003). Toward Evaluation Techniques for Music Similarity. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 7–11.

Logan, B. & Others (2000). Mel Frequency Cepstral Coefficients for Music Modeling. In *Proceedings of the International Society of Music Information Retrieval Conference (ISMIR).*

Lux, M. & Granitzer, M. (2005). A Fast and Simple Path Index Based Retrieval Approach for Graph Based Semantic Descriptions. *In Proceedings of the Second International Workshop on Text-Based Information Retrieval.*

Maaten, L. v. d. & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, *9*(Nov), 2579–2605.

Mausam, Schmitz, M., Bart, R., Soderland, S., & Etzioni, O. (2012). Open language learning for information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL).*

McAuley, J., Pandey, R., & Leskovec, J. (2015a). Inferring networks of substitutable and complementary products. In *Proceedings of the 21st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'15)*, p. 12.

McAuley, J., Targett, C., Shi, Q., & Van Den Hengel, A. (2015b). Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 43–52.

McFee, B., Bertin-Mahieux, T., Ellis, D. P., & Lanckriet, G. R. (2012). In *Proceedings of the 21st international conference companion on World Wide Web - WWW '12 Companion*, p. 909.

McFee, B. & Lanckriet, G. R. (2009). Heterogeneous embedding for subjective artist similarity. In *Proceedings of the 10th International Society of Music Information Retrieval Conference (ISMIR)*, pp. 513–518.

Mcfee, B., Raffel, C., Liang, D., Ellis, D. P. W., Mcvicar, M., Battenberg, E., & Nieto, O. (2015). librosa: Audio and music signal analysis in Python. *Proceedings of the 14th Python in Science Conference*, (Scipy), 1–7.

McKay, C. & Fujinaga, I. (2006). Musical genre classification: Is it worth pursuing and how can it be improved? In *Proceedings of the 7th International Society of Music Information Retrieval Conference*, pp. 101–106.

McKay, C. & Fujinaga, I. (2008). Combining features extracted from audio, symbolic and cultural sources. In *Proceedings of the 9th International Society of Music Information Retrieval Conference*, pp. 597–602.

McNee, S. M., Riedl, J., & Konstan, J. A. (2006). Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, pp. 1097–1101.

Mendes, P. N., Jakob, M., García-Silva, A., & Bizer, C. (2011). DBpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pp. 1–8. ACM.

Middleton, S. E., Roure, D. D., & Shadbolt, N. R. (2009). Ontology-Based Recommender Systems. *Handbook on Ontologies*, *32*(6), 779–796.

Mihalcea, R. & Csomai, A. (2007). Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pp. 233–242. ACM.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013a). Distributed Representations of Words and Phrases and their Compositionality. *Advances in neural information processing systems*, pp. 3111–3119.

Mikolov, T., Yih, W.-t., & Zweig, G. (2013b). Linguistic Regularities in Continuous Space Word Representations. In *HLT-NAACL*, pp. 746–751.

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Commun. ACM*, *38*(11), 39–41.

Mobasher, B., Jin, X., & Zhou, Y. (2004). Semantically Enhanced Collaborative Filtering on the Web. In B. Berendt, A. Hotho, D. Mladenic, M. Someren, M. Spiliopoulou, & G. Stumme (Eds.) *Web Mining: From Web to Semantic Web*, *Lecture Notes in Computer Science*, vol. 3209, pp. 57–76. Springer Berlin Heidelberg.

Moïsi, D. (2010). *The Geopolitics of Emotion: How Cultures of Fear, Humiliation, and Hope are Reshaping the World.* Anchor Books.

Montero, C. S., Munezero, M., & Kakkonen, T. (2014). Investigating the Role of Emotion-Based Features in Author Gender Classification of Text. In *Computational Linguistics and Intelligent Text Processing*, pp. 98–114. Springer.

Mooney, R. J. & Roy, L. (1999). Content-Based Book Recommending. *Proceedings of the SIGIR-99 Workshop on Recommender Systems: Algorithms and Evaluation.*

Moro, A., Cecconi, F., & Navigli, R. (2014a). Multilingual word sense disambiguation and entity linking for everybody. In *Proceedings of the 13th Internation Conference on Semantic Web (P&D).*

Moro, A. & Navigli, R. (2012). WiSeNet: Building a Wikipedia-based semantic network with ontologized relations. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 1672–1676.

Moro, A. & Navigli, R. (2013). Integrating syntactic and semantic analysis into the open information extraction paradigm. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2148–2154. AAAI Press.

Moro, A., Raganato, A., & Navigli, R. (2014b). Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics (TACL)*, *2*, 231–244.

Musto, C., Semeraro, G., Lops, P., & de Gemmis, M. (2014). Combining distributional semantics and entity linking for context-aware content-based recommendation. In *Proceedings of the User Modeling, Adaptation, and Personalization 22nd International Conference (UMAP)*, pp. 381–392.

Nadeau, D. & Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, *30*(1), 3–26.

Nakashole, N., Weikum, G., & Suchanek, F. M. (2012). PATTY: A Taxonomy of Relational Patterns with Semantic Types. *EMNLP-CoNLL*, (July), 1135–1145.

Navarro, J. L. & Ropero, M. (1995). *Historia del flamenco.* Sevilla: Ed. Tartessos.

Navigli, R. & Ponzetto, S. P. (2010). BabelNet : Building a very large multilingual semantic network. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, (July), 216–225.

Navigli, R. & Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, *193*, 217–250.

Neumayer, R. & Rauber, A. (2007). Integration of text and audio features for genre classification in music information retrieval. In *European Conference on Information Retrieval*, pp. 724–727. Springer.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., & Ng, A. Y. (2011). Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 689–696.

Nguyen, K. A., Schulte im Walde, S., & Vu, N. T. (2016). Neural-based Noise Filtering from Word Embeddings. *Coling-2016*, pp. 2699–2707.

Ning, X. & Karypis, G. (2012). Sparse linear methods with side information for top-n recommendations. In *Proceedings of the Sixth ACM Conference on Recommender Systems*, RecSys '12, pp. 155–162.

Oramas, S., Espinosa-Anke, L., Lawlor, A., & Others (2016a). Exploring customer reviews for music genre classification and evolutionary studies. In *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR 2016).*

Oramas, S., Espinosa-Anke, L., Sordo, M., Saggion, H., & Serra, X. (2016b). ELMD: An automatically generated entity linking gold standard dataset in the music domain. In *Proceedings of the 10th International Conference on Language Resources and Evaluation, LREC 2016.*

Oramas, S., Espinosa-Anke, L., Sordo, M., Saggion, H., & Serra, X. (2016c). Information extraction for knowledge base construction in the music domain. *Data and Knowledge Engineering*, *106*, 70–83.

Oramas, S., Gómez, F., Gómez, E., & Mora, J. (2015a). Flabase: Towards the creation of a flamenco music knowledge base. In *Proceedings of the 16th International Society for Music Information Retrieval Conference.*

Oramas, S., Nieto, O., Barbieri, F., & Serra, X. (2017a). Multi-label music genre classification from audio, text, and images using deep features. In *Proceedings of the 18th International Society of Music Information Retrieval Conference ISMIR 2017.*

Oramas, S., Nieto, O., Sordo, M., & Serra, X. (2017b). A deep multimodal approach for cold-start music recommendation. In *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems, co-located with RecSys 2017*.

Oramas, S., Ostuni, V. C., Di Noia, T., Serra, X., & Di Sciascio, E. (2016d). Sound and music recommendation with knowledge graphs. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *8*(2), 21.

Oramas, S. & Sordo, M. (2016). Knowledge is out there: A new step in the evolution of music digital libraries. *Fontes Artis Musicae*, *63*(4), 285–298.

Oramas, S., Sordo, M., & Espinosa-Anke, L. (2015b). A rule-based approach to extracting relations from music tidbits. In *Proceeding of the 2nd Workshop in Knowledge Extraction from Text, WWW'15*.

Oramas, S., Sordo, M., Espinosa-Anke, L., & Serra, X. (2015c). A semantic-based approach for artist similarity. *Proceedings of 16th the International Society for Music Information Retrieval Conference (ISMIR)*.

Ostuni, V. C., Di Noia, T., Di Sciascio, E., & Mirizzi, R. (2013). Top-N recommendations from implicit feedback leveraging Linked Open Data. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pp. 85–92.

Ostuni, V. C., Di Noia, T., Mirizzi, R., & Di Sciascio, E. (2014). A Linked Data recommender system using a neighborhood-based graph kernel. In *The 15th International Conference on Electronic Commerce and Web Technologies*, Lecture Notes in Business Information Processing. Springer-Verlag.

Pachet, F. & Cazaly, D. (2000). A taxonomy of musical genres. In *Content-Based Multimedia Information Access-Volume 2*, pp. 1238–1245.

Passant, A. (2010). dbrec: music recommendations using DBpedia. In *Proceedings of 9th International Semantic Web Conference*, ISWC'10, pp. 209–224.

Passant, A. & Decker, S. (2010). Hey! Ho! Let's Go! Explanatory Music Recommendations with dbrec. In *Proceedings of the 7th Extended Semantic Web Conference (ESWC)*, pp. 411–415.

Passant, A. & Raimond, Y. (2008). Combining Social Music and Semantic Web for music-related recommender systems. In *Social Data on the Web Workshop*.

Pattuelli, M. C., Miller, M., Lange, L., Fitzell, S., & Li-Madeo, C. (2013). Crafting Linked Open Data for Cultural Heritage: Mapping and Curation Tools for the Linked Jazz Project. *Code4Lib Journal*, p. 4.

Pereira, B. (2014). Entity linking with multiple knowledge bases: An ontology modularization approach. In *International Semantic Web Conference*, pp. 513–520. Springer.

Pilászy, I. & Tikk, D. (2009). Recommending new movies: even a few ratings are more valuable than metadata. In *Proceedings of the third ACM conference on Recommender systems*, pp. 93–100. ACM.

Pons, J., Lidy, T., & Serra, X. (2016). Experimenting with musically motivated convolutional neural networks. In *Content-Based Multimedia Indexing (CBMI), 2016 14th International Workshop on*, pp. 1–6. IEEE.

Porter, A., Bogdanov, D., Kaye, R., Tsukanov, R., Serra, X., Group, M. T., Fabra, U. P., & Foundation, M. (2015). Acousticbrainz: a community platform for gathering music information obtained from audio. *16th International Society for Music Information Retrieval Conference (ISMIR 2015)*, pp. 786–792.

Raimond, Y., Abdallah, S. A., Sandler, M. B., & Giasson, F. (2007). The music ontology. In *Proceedings of the International Society of Music Information Retrieval Conference (ISMIR)*, vol. 422.

Raimond, Y., Sutton, C., & Sandler, M. B. (2008). Automatic interlinking of music datasets on the semantic web. *LDOW*, *369*.

Ratcliff, J. W. & Metzener, D. (1988). Pattern matching: The gestalt approach. *Dr. Dobb's Journal*, *13*, 46–72.

Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pp. 452–461. AUAI Press.

Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In F. Ricci, L. Rokach, & B. Shapira (Eds.) *Recommender systems handbook*, pp. 1–35. Springer US.

Rizzo, G., van Erp, M., & Troncy, R. (2014). Benchmarking the extraction and disambiguation of named entities on the semantic web. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Rorvig, M. (1999). Images of similarity: A visual exploration of optimal similarity metrics and scaling properties of TREC topic-document sets. *Journal of the American Society for Information Science*, *50*(8), 639–651.

Rose, S. & Tuppen, S. (2014). Prospects for a big data history of music. In *Proceedings of the International Workshop on Digital Libraries for Musicology.*

Rouvier, M., Delecraz, S., Favre, B., Bendris, M., & Bechet, F. (2015). Multimodal embedding fusion for robust speaker role recognition in video broadcast. *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU).*

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, *115*(3), 211–252.

Saggion, H. & Gaizauskas, R. (2004). Multi-document summarization by cluster/profile relevance and redundancy removal. In *Proceedings of the Document Understanding Conference*, pp. 6–7.

Salakhutdinov, R., Mnih, A., & Hinton, G. (2007). Restricted boltzmann machines for collaborative filtering. In *Proceedings of the 24th International Conference on Machine Learning*, pp. 791–798. ACM.

Sanden, C. & Zhang, J. Z. (2011). Enhancing multi-label music genre classification through ensemble techniques. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pp. 705–714.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pp. 285–295.

Saveski, M. & Mantrach, a. (2014). Item cold-start recommendations: learning local collective embeddings. *RecSys '14 Proceedings of the 8th ACM Conference on Recommender systems*, pp. 89–96.

Schedl, M. (2008). *Automatically extracting, analyzing, and visualizing information on music artists from the World Wide Web.* PhD Dissertation, Johannes Kepler University Linz.

Schedl, M., Gómez, E., & Urbano, J. (2014). Music Information Retrieval: Recent Developments and Applications. *Foundations and Trends® in Information Retrieval*, *8*(2–3), 127–261.

Schedl, M. & Hauger, D. (2012). Mining microblogs to infer music artist similarity and cultural listening patterns. In *Proceedings of the 21st International Conference on World Wide Web*, pp. 877–886. ACM.

Schedl, M., Hauger, D., & Urbano, J. (2013). Harvesting microblogs for contextual music similarity estimation: a co-occurrence-based framework. *Multimedia Systems*, *20*(6), 693–705.

Schedl, M., Knees, P., & Widmer, G. (2005). A Web-Based Approach to Assessing Artist Similarity using Co-Occurrences. In *Proceedings of the 4th International Workshop on Content-Based Multimedia Indexing.*

Schindler, A. & Rauber, A. (2015). An audio-visual approach to music genre classification through affective color features. In *European Conference on Information Retrieval*, pp. 61–67. Springer.

Schörkhuber, C. & Klapuri, A. (2010). Constant-Q transform toolbox for music processing. *7th Sound and Music Computing Conference*, (JANUARY), 3–64.

Serra, X. (2014). Creating Research Corpora for the Computational Study of Music : the case of the CompMusic Project. *53rd International Conference: Semantic Audio (January 2014)*, pp. 1–9.

Seyerlehner, K., Schedl, M., Pohle, T., & Knees, P. (2010a). Using block-level features for genre classification, tag classification and music similarity estimation. *Submission to Audio Music Similarity and Retrieval Task of MIREX, 2010.*

Seyerlehner, K., Widmer, G., Schedl, M., & Knees, P. (2010b). Automatic music tag classification based on block-level. In *Proceedings of Sound and Music Computing 2010.*

Shawe-Taylor, J. & Cristianini, N. (2004). *Kernel Methods for Pattern Analysis.* New York, NY, USA: Cambridge University Press.

Slizovskaia, O., Gómez, E., & Haro, G. (2017). Musical instrument recognition in user-generated videos using a multimodal convolutional neural network architecture. In *ICMR '17: Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval.*

Smith, T. (2009). The social media revolution. *International journal of market research*, *51*(4), 559–561.

Soon, W. M., Ng, H. T., & Lim, D. C. Y. (2001). A Machine Learning Approach to Coreference Resolution of Noun Phrases. *Computational linguistics*, *27*(4), 521–544.

Sordo, M. (2012). Semantic annotation of music collections: A computational approach. In *PhD Dissertation, Universitat Pompeu Fabra.*

Sordo, M., Gouyon, F., & Sarmento, L. (2010). A method for obtaining semantic facets of music tags. In *1st Workshop On Music Recommendation And Discovery, ACM RecSys 2010.*

Sordo, M., Gouyon, F., Sarmento, L., Celma, Ò., & Serra, X. (2013). Inferring semantic facets of a music folksonomy with wikipedia. *Journal of New Music Research*, *42*(4), 346–363.

Sordo, M., Serrà, J., & Serra, X. (2012). A method for extracting semantic information from on-line art music discussion forums. In *2nd CompMusic Workshop*, pp. 55–60.

Speck, R., Röder, M., Oramas, S., Espinosa-Anke, L., & Ngonga Ngomo, A.-C. (2017). Open knowledge extraction challenge 2017. In *Semantic Web Challenges: Fourth SemWebEval Challenge at ESWC 2017*, Communications in Computer and Information Science. Springer International Publishing.

Srivastava, N. & Salakhutdinov, R. (2012). Learning representations for multimodal data with deep belief nets. In *International conference on machine learning workshop.*

Steck, H. (2013). Evaluation of recommendations: rating-prediction and ranking. In *RecSys*, pp. 213–220.

Stevenson, M. & Wilks, Y. (2003). Word sense disambiguation. *The Oxford Handbook of Comp. Linguistics*, pp. 249–265.

Sturm, B. L. (2012). A survey of evaluation in music genre recognition. In *International Workshop on Adaptive Multimedia Retrieval*, pp. 29–66. Springer.

Suchanek, F. M., Kasneci, G., & Weikum, G. (2007). Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, pp. 697–706. ACM.

Sukhbaatar, S., Weston, J., Fergus, R. et al. (2015). End-to-end memory networks. In *Advances in neural information processing systems*, pp. 2440–2448.

Sutcliffe, R., Crawford, T., Fox, C., Root, D. L., & Hovy, E. (2015). Relating natural language text to musical passages. *Proceedings of the 16th International Society for Music Information Retrieval Conference.*

Sutcliffe, R. F., Collins, T., Hovy, E. H., Lewis, R., Fox, C., & Root, D. L. (2016). The c@merata task at mediaeval 2016: Natural language queries derived from exam papers, articles and other sources against classical music scores in musicxml. In *Proceedings of the MediaEval 2016 Workshop.*

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826.

Tata, S. & Di Eugenio, B. (2010). Generating fine-grained reviews of songs from album reviews. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 1376–1385.

Tesnière, L. (1959). *Elements de Syntaxe Structurale.* Editions Klincksieck.

Tsoumakas, G. & Katakis, I. (2006). Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, *3*(3).

Turnbull, D., Barrington, L., & Lanckriet, G. (2008a). Five approaches to collecting tags for music. *Proceedings of 9th the International Society for Music Information Retrieval Conference*, pp. 225–230.

Turnbull, D., Barrington, L., Torres, D., & Lanckriet, G. (2008b). Semantic annotation and retrieval of music and sound effects. *IEEE Transactions on Audio, Speech, and Language Processing*, *16*(2), 467–476.

Tzanetakis, G. & Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, *10*(5), 293–302.

Usbeck, R., Röder, M., Ngomo, A.-C. N., Baron, C., Both, A., Brümmer, M., Ceccarelli, D., Cornolti, M., Cherix, D., Eickmann, B., Ferragina, P., Lemke, C., Moro, A., Navigli, R., Piccinno, F., Rizzo, G., Sack, H., Speck, R., Troncy, R., Waitelonis, J., & Wesemann, L. (2015). GERBIL – General Entity Annotator Benchmarking Framework. *Proceedings of the 24th International Conference on World Wide Web, WWW 2015*, pp. 1133–1143.

Vall, A., Eghbal-zadeh, H., Dorfer, M., Schedl, M., & Widmer, G. (2017). Music playlist continuation by learning from hand-curated examples and song features: Alleviating the cold-start problem for rare and out-of-set songs. In *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems*, pp. 46–54. ACM.

Van den Oord, A., Dieleman, S., & Schrauwen, B. (2013). Deep content-based music recommendation. In *Advances in neural information processing systems*, pp. 2643–2651.

Vigliensoni, G. & Fujinaga, I. (2014). Identifying time zones in a large dataset of music listening logs. In *Proceedings of the First International Workshop on Social Media Retrieval and Analysis*, SoMeRA '14, pp. 27–32. ACM.

Voskarides, N. & Meij, E. (2015). Learning to explain entity relationships in knowledge graphs. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pp. 564–574.

Wang, F., Wang, X., Shao, B., Li, T., & Ogihara, M. (2009). Tag integrated multi-label music style classification with hypergraph. In *Proceedings of the 10th International Society of Music Information Retrieval Conference ISMIR*, pp. 363–368.

Wang, H., Wang, N., & Yeung, D.-Y. (2015). Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1235–1244. ACM.

Wang, M. (2008). A re-examination of dependency path kernels for relation extraction. In *International Joint Conference on Natural Language Processing (IJCNLP)*, pp. 841–846.

Whitman, B. & Ellis, D. P. W. (2004). Automatic record reviews. *Proceedings of 5th International Conference on Music Information Retrieval.*

Whitman, B. & Lawrence, S. (2002). Inferring descriptions and similarity for music from community metadata. In *Proceedings of the 2002 International Computer Music Conference*, pp. 591–598.

Zhang, X., Liu, Z., Qiu, H., & Fu, Y. (2009). A hybrid approach for Chinese named entity recognition in music domain. *8th IEEE International Conference on Dependable, Autonomic and Secure Computing*, pp. 677–681.

Zhou, G. & Su, J. (2002). Named entity recognition using an HMM-based chunk tagger. In *proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 473–480. Association for Computational Linguistics.

Ziegler, C.-N., Lausen, G., & Schmidt-Thieme, L. (2004). Taxonomy-driven computation of product recommendations. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, CIKM '04, pp. 406–415. ACM.

Zobel, J. & Moffat, A. (1998). Exploring the similarity space. *ACM SIGIR Forum*, *32*(1), 18–34.

Appendix A: Publications by the author

Journal papers

Oramas S., Espinosa-Anke L., Sordo M., Saggion H. & Serra X. (2016). Information Extraction for Knowledge Base Construction in the Music Domain. *Data & Knowledge Engineering, Volume 106*, Pages 70-83.

Oramas S., Ostuni V. C., Di Noia T., Serra, X., & Di Sciascio E. (2016). Music and Sound Recommendation with Knowledge Graphs. *ACM Transactions on Intelligent Systems and Technology, Volume 8*, Issue 2, Article 21.

Oramas S., & Sordo M. (2016). Knowledge is Out There: A New Step in the Evolution of Music Digital Libraries. *Fontes Artis Musicae, Vol 63, no. 4.*

Conference papers

Oramas, S., Nieto O., Barbieri F., & Serra X. (2017). Multi-label Music Genre Classification from Audio, Text and Images Using Deep Features. *In Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR 2017).*

Oramas, S., Nieto O., Sordo M., & Serra X. (2017). A Deep Multimodal Approach for Cold-start Music Recommendation. *In Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems, co-located with RecSys 2017.*

Espinosa-Anke, L., Oramas S., Saggion H., & Serra X. (2017). ELMDist: A vector space model with words and MusicBrainz entities. *In Proceedings of the 1st Workshop on Semantic Deep Learning, co-located with ESWC 2017.*

Oramas S., Espinosa-Anke L., Lawlor A., Serra X., & Saggion H. (2016). Exploring Music Reviews for Music Genre Classification and Evolutionary Studies. *In Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR 2016).*

Oramas S., Espinosa-Anke L., Sordo M., Saggion H., & Serra X. (2016). ELMD: An Automatically Generated Entity Linking Gold Standard in the

Music Domain. *In Proceedings of the 10th Conference on Language Resources and Evaluation (LREC 2016).*

Espinosa-Anke, L., Oramas S., Camacho-Collados J., & Saggion H. (2016). Finding and Expanding Hypernymic Relations in the Music Domain. *In Proceedings of the 19th International Conference of the Catalan Association for Artificial Intelligence (CCIA 2016).*

Oramas S., Sordo M., Espinosa-Anke L., & Serra X. (2015). A Semantic-based approach for Artist Similarity. *In Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR 2015).*

Oramas S., Gómez F., Gómez E., & Mora J. (2015). FlaBase: Towards the creation of a Flamenco Music Knowledge Base. *In Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR 2015).*

Ostuni V. C., Oramas S., Di Noia T., Serra, X., & Di Sciascio E. (2015). A Semantic Hybrid Approach for Sound Recommendation. *In Proceedings of the 24th International World Wide Web Conference (WWW 2015, Poster track).*

Oramas S., Sordo M., & Espinosa-Anke L. (2015). A Rule-based Approach to Extracting Relations from Music Tidbits. *In Proceedings of the 2nd Workshop on Knowledge Extraction from Text (KET 2015).*

Sordo, M., Oramas S., & Espinosa-Anke L. (2015). Extracting Relations from Unstructured Text Sources for Music Recommendation. *In Proceedings of the 20th International Conference on Applications of Natural Language to Information Systems (NLDB 2015).*

Oramas S., Sordo M., & Serra X. (2014). Automatic Creation of Knowledge Graphs from Digital Musical Document Libraries. *In Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM 2014).*

Oramas S. (2014). Harvesting and Structuring Social Data in Music Information Retrieval. *In Proceedings of the 11th Extended Semantic Web Conference (ESWC 2014, PhD Symposium).*

Font, F., Oramas, S., Fazekas, G., & Serra, X. (2014). Extending Tagging Ontologies with Domain Specific Knowledge. *In Proceedings of the International Semantic Web Conference (ISWC 2014, Poster track).*

Tutorials and Challenges

Oramas S., Espinosa-Anke L., Zhang S., Saggion H., & Serra X. (2016). Natural Language Processing for Music Information Retrieval. *17th International*

Society for Music Information Retrieval Conference (ISMIR 2016).

Speck, R., Röder, M., Oramas, S., Espinosa-Anke, L., & Ngomo, A. C. N. (2017). Open Knowledge Extraction Challenge 2017. *14th Extended Semantic Web Conference (ESWC 2017).*

Conference presentations

Oramas, S. (2017). Knowledge Extraction and Feature Learning for Music Recommendation in the Long Tail. *5th Large Scale Recommendation Systems Workshop, co-located with RecSys 2017*, Como, Italy.

Oramas, S. (2017). Discovering Similarities and Relevance Ranking of Renaissance Composers. *The 63rd Annual Meeting of the Renaissance Society of America (RSA)*, Chicago.

Oramas S. (2015). Information Extraction for the Music Domain. *The 2nd International Workshop on Human History Project: Natural Language Processing and Big Data*, CIRMMT, Montreal.

Oramas, S., & Sordo M. (2015). Knowledge Acquisition from Music Digital Libraries. *The International Association of Music Libraries and International Musicological Society Conference (IAML/IMS 2015)*, New York.

Appendix B: Datasets, Knowledge Bases, and Software

Datasets

ELMD Dataset of ∼13k documents and almost 150k annotated musical entities, which are linked to DBpedia and MusicBrainz. From this corpus, a gold standard dataset of 200 documents with manually annotated entities is also created (Section 3.4). `http://mtg.upf.edu/download/datasets/elmd`

MARD Large dataset of about 64k albums with customer reviews, acoustic features per track, metadata, and single-label genre annotations (Sections 5.3.1 and 6.3.1). `http://mtg.upf.edu/download/datasets/mard`

SAS Two datasets of 188 and 2,336 artist biographies respectively, together with artist similarity ground truth data (Section 6.2.4). `http://mtg.upf.edu/download/datasets/semantic-similarity`

KG-Rec Two datasets of tags and text descriptions about musical items, together with user feedback information on those items. A dataset of sounds with ∼21k items and 20k users, and a dataset of songs with ∼8.5k items and ∼5k users (Section 7.4.1). `http://mtg.upf.edu/download/datasets/knowledge-graph-rec`

MSD-A Dataset of ∼24k artist biographies linked to the artists present in the Million Song Dataset (Section 8.6.1). `http://mtg.upf.edu/download/datasets/msd-a`

MuMu Large dataset of about ∼31k albums, with ∼450k customer reviews, ∼147k audio tracks, cover artworks, and multi-label genre annotations (Section 9.2). `https://www.upf.edu/web/mtg/mumu`

Knowledge bases

KBSF Knowledge base of popular music extracted from a corpus of ∼32k documents with stories about songs (Section 4.3.2). `http://mtg.upf.edu/download/datasets/kbsf`

FlaBase Knowledge base of flamenco music, created by combining data from 7 different data sources, and enriched with information extracted from ∼1k artist biographies (Section 5.2.2). `http://mtg.upf.edu/download/datasets/flabase`

Software

ELVIS System that integrates different entity linking tools, enriching their output and providing high confident entity disambiguations. `https://github.com/sergiooramas/elvis`

Tartarus System to perform and evaluate deep learning experiments on classification and recommendation from different data modalities and their combination. `https://github.com/sergiooramas/tartarus`