Master thesis on Intelligent Interactive Systems

Universitat Pompeu Fabra

Information Extraction from User-generated Content in the Classical Music Domain

Lorenzo Porcaro

Supervisor: Horacio Saggion

September 2018

Table of Contents

Acknowledgments

The work presented in this thesis was made possible with the help of several people. First of all, I would like to thank Dr. Horacio Saggion for giving me the possibility to work on this topic and for the valuable advices and suggestions during the last months. Thanks to Dr. Francesco Ronzano and Dr. Frederic Font, respectively from the TALN and the MTG research groups of the UPF, for the feedbacks given during the presentation, which have been very constructive and that I have appreciated very much. Thanks to Dr. Pedro Cano who, before starting this work, suggested me several ideas for my master thesis while I was working at BMAT. Thanks to Mireia Farrús who as coordinator of the master in Intelligent Interactive System has helped me and the other students during the last two years.

Lastly, thanks to my family and to my friends, who even if from afar, have always supported me.

Abstract

The applications of Information Extraction (IE) on User-generated Content (UGC) have widely benefited from the emergence of microblogging services in the last decade. In particular, *Twitter* has been at the center of attention of many studies because of its widespread use and easy accessibility. Among the several fields which have benefited from this source, in particular Named Entity Recognition (NER) has demonstrated how challenging can be obtaining useful information from the noisy space of *tweets*. From another perspective, recently in the field of Music Information Retrieval (MIR) researches have shown how NLP techniques such as IE and NER can be an important resource to improve accuracy and precision in tasks like Music Recommendation, Artist Similarity or Genre Classification. The objective of this thesis is to investigate methods to extract information from user-generated content in a specific channel related to Classical Music, *BBC Radio 3*, through the use of NER techniques. We investigate how state-of-the-art methods in NER can be applied to detect entities in the music domain, and how contextual information can contribute with NER in this particular case.

Keywords: Information Extraction; Named Entity Recognition; Music Information Retrieval; User-generated content.

1. Introduction

The increasing use of social media and microblogging services has broken new ground in the field of Information Extraction from user-generated content. Both the scientific community and private companies have invested many resources on developing new techniques to perform linguistic and semantic analysis of raw texts coming from the web, which nowadays can be considered as one of the most representative source of personalized content publicly available.

Understanding the information contained in this kind of content has become one of the main goal for many applications, due to the uniqueness and the variety of this data. However, the highly informal and noisy texts generated by users makes it difficult to apply techniques proposed by the NLP community for dealing with formal and structured content. In this work we focus on the analysis of an IE sub task, Named Entity Recognition, aiming at comprehend how it is possible to improve its precision when handling user-generated content.

We choose to use the microblogging service *Twitter* as source of user-generated content. This choice is based on several reasons. Firstly, it is one of the most popular platforms for sharing opinions, comments and general news. Secondly, even if the messages posted are highly informal and noisy, they represent an optimal source for understanding user behaviours. Lastly, the shortness of the contents makes difficult to define the context in which they are generated.

In this work, we analyze a set of *tweets* obtained from crawling a specific classical music radio channel, *BBC Radio 3*. We focus on a radio channel because we want to verify if using its schedule can help on defining the context while users generate *tweets.* Indeed, we aim to detect relationships between the tracks broadcasted and the messages posted, thanks to which we want to improve the detection of the entities.

The method proposed is divided in two parts: we use a matching algorithm for creating links between *tweets* and tracks played on the radio, aimed to recognize when users refer to an entity included in the schedule. On the other side, we create a statistical model characterizing the informal language of the *tweets,* used for predicting when in a text named entities are included.

The thesis is structured as follows: chapter 2 presents a review of the previous works done in the Information Extraction field, focusing on its application on user-generated content and Music Information Retrieval. Afterwards, in chapter 3 it is presented the methodology proposed, describing the dataset analyzed and the system built. In chapter 4, the results obtained from the evaluation of the system are shown. Finally, in chapter 5 conclusions are discussed together with possible future works.

1.1. Motivation

In this work, we have centered our attention on understanding the main issues while performing NER from user-generated content in the Classical Music domain. Our study has been driven by two main goals: on one side, we aim to comprehend how it is possible to use Machine Learning techniques for modelling the informal language of the users' raw texts while performing NER. On the other side, we investigate how integrating external contextual information, namely the schedule of the radio in our case, can improve the accuracy of NER.

Within the IE framework, NER plays a key role in the construction of several high-level NLP tasks, for instance relation extraction, opinion mining and summarization. Moreover, there is a wide range of application contexts which have been shown to benefit from NER, including knowledge management, competitor intelligence, customer relation management, eBusiness, eScience, eHealth, and eGovernment [Derczynski et al., 2015].

Several studies done by the NLP community have already proved how IE techniques need to be calibrated when dealing with user-generated content, as shown in section 2.2. Basing

on those results, we focus on the field of Classical Music, where according to the literature reviewed, nothing much has been done in the context of IE.

Our interest in the music field is partially motivated by recent works done by the MIR community, where NLP techniques have been demonstrated to be useful when dealing with tasks such as Artist Similarity, Sound and Music Recommendation and Genre Classification. In sections 2.3 and 2.4, a review of those MIR works is presented.

However, the literature related to NER systems, specifically built for the music field, is not wide. In our work, we want to propose a music-specific method for recognizing classical music entities from user-generated content, an enabler for other MIR and NLP tasks, which potentially can take advantage of the results obtained by our NER system.

2. State of the Art

In the next sections it is presented a review of the works done in the field of Information Extraction. Firstly, an overview of IE general framework is presented. Afterwards, studies related to IE applications to user-generated content, and in particular to microblogging services, are introduced. Finally, MIR tasks which have benefited from IE models are presented.

2.1. Information Extraction

Information Extraction is part of the Natural Language Understanding (NLU) field, and it can be described as the task of transforming an input unstructured data, for instance raw texts, in an output structured data easily readable by a machine, suitable for populating databases [Cardie, 1997]. The opportunity to have a huge amount of text data in electronic form has been the initial point thanks to which IE started to be defined. One of the goals of this process is to detect and link useful information, while discarding irrelevant item. [Cowie & Lehnertand, 1996].

One of the first IE system was JASPER, presented in [Andersen et al., 1992]. It was created for extracting relevant information from press releases, and in particular its aim was to help financial traders in having real-time financial news. With a high accuracy, it was able to replace basic mechanical operations typically done by employees, as extracting earnings and dividend from raw texts.

While IE starts to take hold within the scientific community, an important role in the evaluation of these new systems has been played by the Message Understanding Conferences (MUCs) [Lehnert & Sundheim, 1991]. During these conferences, participants were asked to analyse a corpus of texts, and to fill pre-build templates with information extracted automatically from the input, much like that in figure 1.

In recent decades, the increased interest in IE techniques is linked to the invention and the rapid spread of the World Wide Web. Indeed, it begins to be available a large amount of

unstructured data, in part user-generated, which becomes source of useful information, thanks to new achievements of various fields such as Data Mining, Machine Learning and Natural Language Processing [Small & Medsker, 2014].

In particular, developments in neural architectures have become an important resource for building a new generation of IE systems. The main advantage of these systems is that they do not need language-specific knowledge resources [Lample et al., 2016]. Moreover, these systems has been demonstrated to be robust to the noisy and short nature of social media messages [Lin et al., 2017]

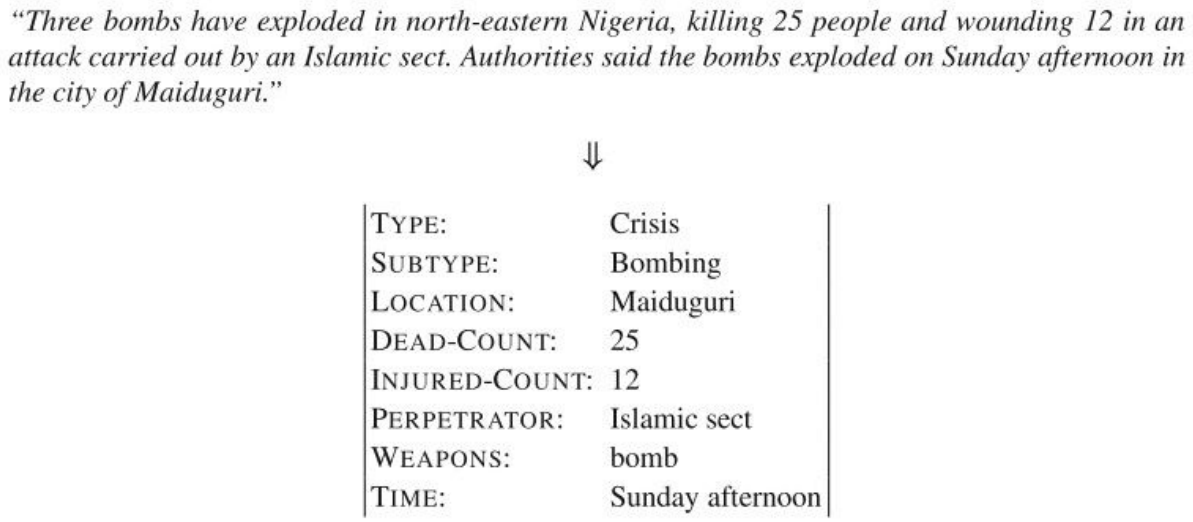

Figure 1: Example of automatically extracted information from a news article on a terrorist attack. The process of extracting such structured information involves identification of certain small-scale structures like noun phrases denoting a person or a person group, geographical references and numeral expressions, and finding semantic relations between them [Piskorski & Yangarber, 2013].

The main difficulties while building IE systems arise from the complexity and ambiguity of natural language. Indeed, different sentences expressed in dissimilar scenarios can carry the same information, or in other cases the meaning of a sentence can be implicit, so more difficult to extract without considering a more wide context [Piskorski & Yangarber, 2013].

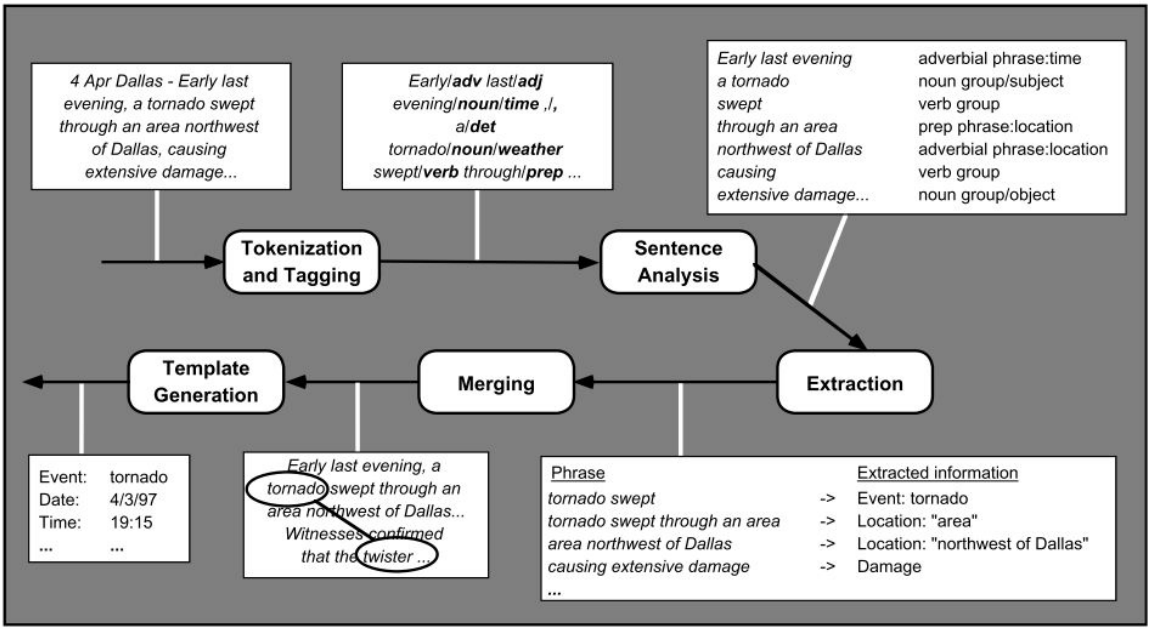

Figure 2: Example of architecture for an information extraction system [Cardie, 1997].

Generally, an IE system is composed by a sequence of sub-tasks, as in the example of figure 2, each one with a specific scope. Within that, we can find:

- *Tokenizing*: it consists on splitting the input sentence in basic units, called *tokens*, such as numbers, punctuation, symbols, and words of different types (e.g. with an initial capital, all upper case, etc.)

- *Part-of-Speech tagging:* it is the process of marking up each token with its grammatical category, such as *noun*, *verb*, *adjective* etc.

- *Text chunking*: it involves dividing sentences into nonoverlapping segments on the basis of fairly superficial analysis [Ramshaw & Marcus, 1995].

- *Dependency parsing*: it describes the syntactic structure of a sentence in terms of the words (or lemmas) and an associated set of directed binary grammatical relations that hold among the words [Jurafsky & Martin, 2009].

- *Named entity recognition, classification, normalization and linking*: it is a group of actions performed to entities in unstructured data. Entities, or named entities, that could be detected using IE are organizations (e.g., 'World Wildlife Fund), persons (e.g., 'Barack Obama'), place names (e.g., 'the Andean Mountains'), temporal expressions (e.g., '17 September 1990), numerical and currency expressions (e.g., 5 Million Dollars) and many others, accordingly to the specific purpose of the system.

- *Coreference resolution*: it examines each entity encountered in the text and determines whether it refers to an existing entity or whether it is new [Cardie, 1997].

- *Relation Extraction*: it is defined as the task of identifying relevant semantic relation between entities in a text [Oramas et al., 2015b].

In this work, we mostly deal with Named Entity Recognition, and following is presented a brief description of the several tasks related to named entities.

2.1.1. Named Entity Recognition and Classification

Named Entity Recognition (NER), or alternatively Named Named Entity Recognition and Classification (NERC), is the task of detecting entities in an input text and to assign them to a specific class. For instance, in the sentence "*Ford is the automotive company created by Henry Ford in 1903*", *Ford* and *Henry Ford* should be recognized as entities, but the first should be classified as organization while the latter as person. Researchers start to define this topic in the early '90, and over the years several approaches have been proposed [Nadeau, 2007]. Early systems were based on handcrafted rule-based algorithms, while recently several contributions by Machine Learning scientists have helped in integrating probabilistic models into NER systems.

2.1.2. Named Entity Normalization

Named Entity Normalization (NEN) aims to derive from the surface forms of the entities used in the texts, an unambiguous reference to real World entities. It can be applied to different kind of entities, as example transforming *'J S Bach'* to *'Johann Sebastian Bach*', or '*1 Million*' to '*1.000.000*'. Moreover, as presented in [Ritter et al., 2011], an example of normalization could be to link the variations of the word *'tomorrow',* found in a group of *tweets* :

'2m', '2ma', '2mar', '2mara', '2maro','2marrow', '2mor', '2mora', '2moro', '2morow', '2morr', '2morro', '2morrow', '2moz', '2mr', '2mro', '2mrrw', '2mrw', '2mw', 'tmmrw', 'tmo', 'tmoro', 'tmorrow', 'tmoz', 'tmr', 'tmro', 'tmrow', 'tmrrow', 'tmrrw', 'tmrw', 'tmrww', 'tmw'

Historically, NEN techniques has been used in the context of structured databases for performing tasks as record linkage and data deduplication [Jijkoun et al., 2008]. However, it has been proven to be extremely valuable when dealing with user-generated content, where often entities are not presented in its full forms. Misspell words, nicknames and creative constructions are hardly detected in IE system, and NEN aim is to improve the performance when dealing with these particular forms .

2.1.3. Named Entity Disambiguation or Linking

Named Entity Disambiguation (NED) or Entity Linking (EL) tackles the problem of linking the surface form of an entity recognized to the semantically correct entity, when different candidates are found. Using the contextual information of the input text, the aim is to maximise the agreement between the entity found and the possible related ones [Cucerzan, 2007]. For instance, considering the sentence: '*Thomas and Mario are strikers playing in Munich*', the goal of an EL system is to recognize the entities *Thomas* and *Mario*, and to link them to *Thomas Muller* and *Mario Gomez*, two well-known soccer players. The main difference between NED and EL system is the kind of inventory used: dictionary in case of NED, and encyclopedia with EL [Moro et al., 2014]. In this work, we will refer at the task of disambiguating and linking using the abbreviations EL.

2.2. Information Extraction for User-generated Content

The end of the 20th century has brought a radical shift in the use of the Web, passing from the named Web 1.0, where users mostly only interacted as consumers of content, to the Web 2.0 where users began to be their-self creators [Berners-Lee, 1999]. This change gave to the scientific community interested in IE incredible opportunities and at the same time new challenges. Indeed, with the spread of new interactive tools, as social media platform and microblogging services, users have been more and more engaged in creating content publicly available, which have become source of useful data for academic and private investigations.

Difficulties arise when trying to extract knowledge from this data using IE models trained for dealing with content with a formal structure and a clean language. Indeed, user-generated contents are very personalized, which often take on complex styles, combine various information and embed much noise [Zhang et al., 2011] .

While the first IE technologies have mostly used Internet for building knowledge databases [Cardie, 1997], with the Web 2.0 methods for extract high-level information from user-generated content become the center of attention, and in particular *Opinion Mining* and *Sentiment analysis* gained a lot of popularity [Pang & Lee, 2006]. Nowadays, it is possible to express personal ideas about almost everything, using tools such as online forums, customer reviews, news-article comments, blogs and social media. Consequently, the IE community started to investigate how to adapt systems to each particular case, obtaining results applicable to a wide range of fields, from financial decision making [Habib & Keulen, 2014] to public health [Elhadad et al, 2014].

In this thesis, we will focus on application of information extraction to *Twitter*, a popular microblogging service founded in 2006. In the next section, it is presented a review of previous works done in this direction.

2.2.1. Information Extraction from Twitter

Twitter [1] can be considered the largest source of public opinion actually available online, and for sure the most popular. It has almost 700 million of registered users, and every day an average number of 58 million of *tweets* are daily posted [Statistic Brain, 2016]. *Tweets* are the messages shared in the platform, previously limited to 140 characters but recently doubled for most languages. A *tweet* can be described as short, informal, ungrammatical and noise prone text [Liu et al., 2011]. The relevance of the information which can be extracted from *tweets* lies in the uniqueness of the content created. In addition, the easy accessibility to the service, facilitated by the spread use of mobile devices, helps the diffusion of the news through the platform, arriving to be one of the most up-to-date source of information available online [Ritter et al., 2011]. In table 1, examples of noisy tweets are presented.

The interest by NLP community arise as the platform became more and more used, and first experiments try to understand how to improve basic tasks, such as Part-of-Speech Tagging, Lexical Normalisation, Named Entity Recognition etc. while dealing with a *tweets* corpus .

As shown in various works [Gimpel et al, 2011] [Han & Baldwin, 2011] [Liu et al., 2011] [Ritter et al., 2011], off-the-shelf methods previously built for processing different kind of text sources do not perform well with *Twitter* data. In fact, there are two issues needed to be considered while processing *tweets* for information extraction:

- *Text informal nature* : the lack of conventional orthography, derived from the use of ad-hoc abbreviations, phonetic substitutions, ungrammatical structures and emoticons, make conventional features such as part-of-speech (POS) and capitalization not reliable. Furthermore, this generates a long tail of Out Of Vocabulary (OOV) words, which makes more difficult the use of automatic systems based on supervised learning.

---

1 www.twitter.com

- *Limited contextual information :* individual *tweets* lack of a complex discourse structure, due to its shortness and informal nature. Ambiguity becomes then a major issue, especially when dealing with IE methods which use coreference information. The discourse information can be hardly extracted from a single *tweet*, considering that its threaded structure is fragmented across multiple documents.

1	The Hobbit has FINALLY started filming! I cannot wait!
2	Yess! Yess! Its official Nintendo announced today that they Will release the Nintendo 3DS in north America march 27 for $250
3	Government confirms blast n nuclear plants n japan...don't knw wht s gona happen nw.

Table 1: Example of noisy text in tweets [Ritter et al., 2011].

Thanks to recent research in adapting NLP tools to *Twitter* data, new results have been achieved adapting IE systems to work with *tweets.* Several tools have been presented, such as *TwitIE* [Bontcheva et al., 2013], a modification of the GATE ANNIE open-source pipeline for news text [Cunningham et al., 2001], and *twitter_nlp* [Ritter et al., 2011] a POS tagger, chunker and Named Entity Recognizer, available for use by the research community.

Furthemore, an analysis of the performances of various state-of-the-art Entity Recognition and Entity Disambiguation systems has been presented in [Derczynski et al., 2015]. They have been tested against several *Twitter* datasets, investigating what the main sources of error are, and which problems should be further investigated to improve the state of the art. In detail, the authors found that poor capitalisation is one of the main issues when dealing with microblog content. Apart from that, typographic errors and the ubiquitous occurrence of out-of-vocabulary words also cause drops in NER recall and precision, together with shortenings and slang, particularly pronounced in *tweets*.

In the next section, we will focus on works in the field of Music Information Retrieval based on the analysis of user generated content, in particular focusing on microblogging.

2.3. Information Extraction in the Music Domain

Music Information Retrieval (MIR) is an interdisciplinary field which borrows tools of several disciplines, such as signal processing, musicology, machine learning, data mining, psychology and many others, for extracting knowledge from musical objects. NLP techniques have also drawn the attention of the MIR community, especially for dealing with the huge amount of information contained in unstructured texts publicly available, in particular from the web. In the last decade, several MIR tasks have benefited from NLP, such as sound and music recommendation [Oramas et al., 2015a], automatic summary of song review [Tata & Di Eugenio, 2010], artist similarity [Schedl & Hauger, 2012] and genre classification [Oramas et al., 2016a].

In [Oramas et al., 2015a], NER and EL techniques are used to semantically enrich knowledge graphs representing sound and musical items. Exploiting contextual information, such as tags and text descriptions, it has been shown that features extracted from the enriched knowledge graphs can be used for creating an hybrid recommender system.

While trying to automatically generate song reviews starting from album reviews , in [Tata & Di Eugenio, 2010] it is presented how the basic step in identifying the song title can be performed thanks to NER techniques. Once the titles are identified, they are used to locate information related to each song in the original text, which afterwards are merged for creating a summary of the songs.

Moreover, including in a multimodel dataset information extracted using Opinion Mining, Sentiment Analysis and Entity Linking techniques, it has been proven to be useful for performing Music Genre classification. In [Oramas et al., 2016a] textual, semantic and sentimental features extracted from song and album reviews are combined with acoustic features for predicting the music genre of the input items.

According to the reviewed literature, Entity Recognition and Linking is one of the basic sub-task on which high level IE models are based for extracting semantic information from raw texts. A first approach for detecting a Musical Named Entity (MNE) from raw text has been proposed in [Zhang et al., 2009], centered in Chinese language. It has been shown how mixing a machine learning method, based on Hidden Markov Models, and a rule-based model incorporated in the pre-processing and post-processing, can effectively bring significant improvements in the MNE recognition.

A more recent approach tries to tackle the problem of detecting musical entities from *Last.fm* [1] raw texts, combining state-of-the-art EL systems [Oramas et al., 2016b]. The method proposed relies on the *argumentum ad populum* intuition, so if two or more different EL systems perform the same prediction in linking a named entity mention, the more likely this prediction is to be correct. In detail, the off-the-shelf systems used are: *DBpedia Spotlight* [2] [Mendes et al., 2011], *TagMe* [3] [Ferragina and Scaiella, 2012], *Babelfy* [4] [Moro et al., 2014]. In figure 3, the workflow of the method proposed is presented.

Moreover, a first Musical Entity Linking, MEL [5] has been presented in [Oramas et al., 2017] which combines different state-of-the-art NLP libraries and SimpleBrainz, an RDF knowledge base created from MusicBrainz [6] after a simplification process.

Lastly, it is import to notice that the previous techniques have contributed to the wider purpose of building a Knowledge Base (KB) specific to the music domain. A KB is a collection of organized knowledge structured with a specific taxonomy or ontology. In [Oramas et al., 2016c] it is presented a methodology for building a Musical KB using information extraction techniques.

---

1 http://www.last.fm
2 https://github.com/dbpedia-spotlight
3 https://sobigdata.d4science.org/web/tagme/tagme-help
4 http://babelfy.org
5 http://mel.mtg.upf.edu/static/index.html
6 https://musicbrainz.org/

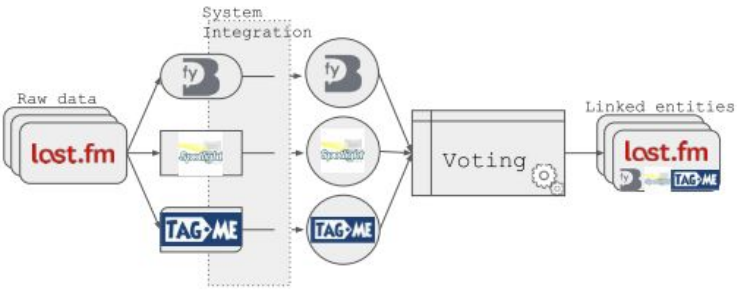

Figure 3: Workflow of the proposed method in [Oramas et al., 2016b].

2.4. Music Information Retrieval and User-generated Content

In section 2.2 we have seen how user-generated content has attracted the NLP community, leading to the development of several techniques adapted to this framework. Also MIR researchers have been able to use the increasing number of resources available in form of natural language text, especially from the web.

The first approaches proposed for extracting musical knowledge has been focused on source such as reviews [Whitman & Lawrence, 2002] and online forums [Sordo et al., 2012]. An interesting point of the analysis of unstructured text from the web is to understand how users interact between each others while generating content. Especially when trying to extract semantic information, network analysis has been proven to be useful, and particularly adaptable to source such as online forums .

Thanks to its structure, also *Twitter* contains meaningful information about user interactions. The use of hashtags and citations within the *tweets* helps in contextualizing the information carried in a single message, creating an interconnected network of texts. Furthermore, the possibility to link each *tweets* with further information, such as place and time where the text has been written, can be used for retrieving similarity between content

generate by users. This kind of information has been exploited by the MIR community for improving results in tasks such as artist similarity and recommender systems [Zangerle et al, 2012], but also for discovering cultural listening patterns [Schedl & Hauger, 2012].

For building a recommender system using a *Twitter* corpora, [Zangerle et al, 2012] analyzes *tweets* generated by several users containing keywords like X*nowplaying* or *listeningto*. From those, they mapped the song and artists detected to entries in MusicBrainz and FreeDB [1] databases. Finally, the recommendation of music titles is built using the co-occurrence of titles within a user stream.

In [Schedl & Hauger, 2012], similarly it is used a dataset of *tweets* containing the hashtag *#nowplaying*. The idea in this work is to extract geospatial listening patterns from microblogs, analyzing if and in which way they differ among different parts of the world and eventually interpret these differences.

Publicly available datasets of *tweets* specific for music investigations have been created, among the others Million Musical Tweets Dataset [2] [Hauger et al., 2013] and #nowplaying dataset [3] [Zangerle et al., 2014] .

1 http://www.freedb.org/
2 http://www.cp.jku.at/datasets/MMTD/
3 http://dbis-nowplaying.uibk.ac.at/

3. Methodology

We propose a hybrid method which recognizes musical entities in user-generated content using both contextual and linguistic information. We focus on detecting two types of entities:

- *Contributor:* person who is related somehow to a musical work (composer, performer, conductor, etc)
- *Work*: musical composition or recording

As case study, we have chosen to analyze *tweets* extracted from the channel of a classical music radio, *BBC Radio 3* [1] . The choice to focus on classical music has been mostly motivated by the particular discrepancy between the informal language used in the *tweets* and the formal nomenclature of contributors and musical works. Indeed, users when referring to a musician or to a classical piece in a *tweet*, rarely use the full name of the person or of the work, as shown in table 2. The choice of this particular channel has been based on the number of followers, over 100K, and the high level of users engagement, particularly active in comparison to other classical music radio channels in *Twitter*.

No **Schoenberg** or **Webern**?? **Beethoven** is there but not his **pno sonata op. 101**??

Informal form	*Formal form*
Schoenberg	Arnold Franz Walter Schoenberg
Webern	Anton Friedrich Wilhelm Webern
Beethoven	Ludwig van Beethoven
pno sonata op. 101	Piano Sonata No. 28 in A major, Op. 101

Table 2: Example of user-generated tweet from BBC Radio 3 channel (top). Entities found in the tweet (left), and corresponding formal forms (right).

1 https://twitter.com/BBCRadio3

In this work, we focus on the NER sub task, tackling the two problems, presented in section 2.2.1, while using Information Extraction techniques on a *Twitter* corpora: the *text informal nature* and the *limited contextual information*. For accomplishing it, we create a system composed of two main parts:

- Firstly, we use the radio schedule for recreating the context while users generate *tweets*. Thanks to it, we are able to detect if an user is referring to a specific work or contributor recently played. We manage to associate to every track broadcasted a list of entities, extracted using a probabilistic model based on both linguistic and contextual features: POS and chunk tag, token position in the *tweet*, gazetteers and intra-token relationships.

- Afterwards, we detect the entities on the user-generated content by means of two methods: on the one side, we use the entities extracted from the schedule for generating candidates entities in the user-generated *tweets*, thanks to a matching algorithm based on time proximity and token similarity. On the other side, we create a probabilistic model capable of detecting entities directly from the user-generated *tweets*, aimed to model the informal language used. In this case, the features are the same used for modelling the schedule.

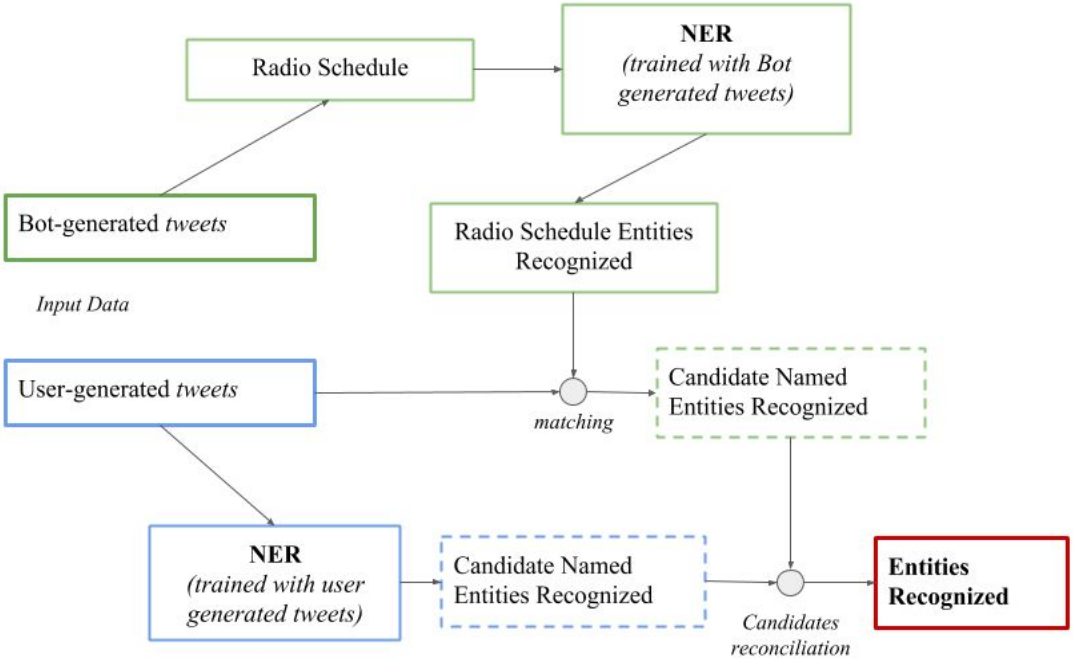

Figure 4: General overview of the proposed system for recognizing named entities from user-generated tweets, using linguistic features and radio schedule.

In the next sections, we describe the setting under which the experiment is carried out. First, we refer to the source corpus, both for the user-generated content and the radio schedule. Afterwards, we introduce the set of features used in the statistical model. Finally, we present the matching algorithm for generating candidate entities using the schedule.

3.1. Source dataset

In May 2018, we crawl *Twitter* using the Python library *Tweepy* [1]. We divided the dataset in two parts:

1) the first part contains a set of user-generated *tweets* related to the *BBC Radio 3* channel. This set represents the source of user-generated content on which we aim to predict the named entities. We create it filtering the messages containing hashtags related to *BBC Radio 3*, such as '*#BBCRadio3*', '*#bbcradio3*' or '*#BBCR3*'. In addition, we include *tweets* generated by the *BBC Radio 3* official channel and the ones generated by the channel of a *BBC Radio 3* program, *BBCInTune* [2]. We obtain a set of 3,205 unique user-generated *tweets*.

2) The second part is built using messages automatically generated by the *BBC Radio 3 Music Bot* [3]. The *BBC Radio 3 Music Bot* is a bot programmed for generating a *tweet* each time that a track is broadcasted on the radio. Examples of this set are presented in table 3. The result of the crawling is a set of 5,098 automatically generated *tweets*, thanks to which we have been able to recreate the schedule of the radio during May 2018.

1 http://www.tweepy.org/
2 https://twitter.com/BBCInTune
3 https://twitter.com/BBCR3MusicBot

Wrap up warm and dive right in... Rimsky-Korsakov's delightfully colourful, folk-inspired opera The Snow Maiden
Heard some of Opera 'Oberon' today... Weber... Only a little....
Listening to the wonderful 5th Symphony by Mahler on @BBCRadio3 and wondering WHY so many 5th symphonies are SO fantastic. Beethoven, Sibelius, Prokofiev, Shostakovich, the list goes on...

Now Playing Joaquín Rodrigo, Goran Listes - 3 Piezas espanolas for guitar #joaquínrodrigo,#goranlistes https://t.co/otS02e94Uu
Now Playing Robert Schumann, Luka Mitev - Phantasiestücke, Op 73 #robertschumann,#lukamitev https://t.co/uwYWQDjqzN"
Now Playing Pyotr Ilyich Tchaikovsky, MusicAeterna - Symphony No.6 in B minor #pyotrilyichtchaikovsky, #musicaeterna https://t.co/tB5OfzGljF"

Table 3: Examples of user-generated tweet by BBC Radio 3 listeners (top) and automatically generated tweets by the BBC Radio 3 Music Bot (bottom).

3.2. Features' Description

In this section, we present the features used for creating a probabilistic model able to predict the named entities from user-generated content. For each token, we mix a set of standard linguistic features together with several gazetteers, built specifically for classical music entities. In addition, we complete the token characterization with a series of features representing the context in what it is presented.

3.2.1. Token-specific features

The first features extracted help us in defining a series of linguistic attributes of the words used in the *tweet*. For each token, we retrieve:

- POS tag
- Chunk tag
- Position of the token within the *tweet*, normalized between 0 and 1
- If the token starts with a capital letter
- If the token is a digit

For extracting the POS tag and the chunk we use the Python library *twitter_nlp* [1], presented in [Ritter et al., 2011] .

3.2.2. Gazetteers

The gazetteers proposed are tailored to the case of classical music entities. We create them using information extracted from *Wikipedia* [2] and other external sources, and then extended them with information from the schedule. The gazetteers used in this work are:

- Contributor first names
- Contributor last names
- Classical singing voice types ("*soprano"*, "*tenor"*, *etc*.)
- Classical work types ("*symphony"*, "*overture"*, "*march", etc.*)
- Classical instruments
- *Opus* forms ("*op*", "*opus*")
- Work number forms ("*no*", "*number*")
- Work keys ("*C*", "*D*", "*E*", "*F*" , "*G*" , "*A*", "*B*", "*flat*", "*sharp*", "*major*", "*minor*")

The first three gazetteers listed are oriented to detect if a token is part of *contributor* entities. The gazetteers of first names and last names are created using the radio schedules and several *Wikipedia* pages containing lists of classical music composers [3]. On the contrary, the other gazetteers identify characteristics of tokens which might be part of a classical music work title.

3.2.3. Intra-token features

We introduce a set of features for including into the probabilistic model information about the context on which a word is used. For each token, we include as features the POS tag and the chunk tag of the previous and the following two tokens.

---

1 https://github.com/aritter/twitter_nlp/
2 https://www.wikipedia.org/
3 https://en.wikipedia.org/wiki/List_of_classical_music_composers_by_era

Furthermore, we also add within the intra-token features if the previous or following token is a digit. It has been included for emphasising the cases where it is present a work name. Indeed, frequently the name of the work include the *opus* or the work number, as instance *'Piano Sonata No. 28 in A major, Op. 101'*. In table 4, an example of POS and Chunk tag considered as intra-token features.

POS Tag	WRB	IN	NNP	POS	NNP
Token	how	about	Sciarrino	's	Anamorfosi
Token Number	Token -2	Token -1	Token	Token +1	Token +2
Chunk Tag	B-ADVP	B-PP	B-NP	B-NP	B-NP

*Table 4: Excerpt of user-generated tweet annotated with POS and chunk tag. When analyzing the token "*Sciarrino*" all the POS and chunk tags in the table are used as feature for recreating the context of the token.*

3.3. Machine Learning Model

In order to understand which statistical technique performs better using the set of features presented in section 3.2, we use the *Weka* package [1] presented in [Frank et al., 2016] for testing some of the standard algorithms of the Machine Learning literature. Our objective is to create two models, one for predicting the entities into the bot-generated *tweet*, one for the user-generated *tweet*. *Weka* offers a series of pre-built classifiers, and in this work we choose to test the performances of the following algorithms:

- Support vector machine (SMO) [Platt, 1998]
- Naive Bayes (NaiveBayes) [John & Langley, 1995]
- C4.5 decision tree (J48) [Quinlan, 1993]
- K-nearest neighbours (IdK) [Aha & Kiblerm, 1991]

1 https://www.cs.waikato.ac.nz/ml/weka/

For training the model, we use 600 manually annotated *tweets* in the case of user-generated content, while 2K bot-generated *tweets* for modelling the schedule.

The difference of amount of *tweets* used is due to the fact that generally user-generated *tweets* are longer than the ones generated by the bot. Consequently, because of limited hardware capacity it has not been possible to use more examples in the training part. The evaluation of the model has been done using 10-fold cross validation. In chapter 4, the results obtained are presented.

3.4. Schedule Matching

In this section, it is presented how using the radio schedule we extract candidates entities from the user-generated *tweets*. The hypothesis considered is that when a user posts a *tweet*, it is possible that he or she is referring to a track which has been played a relatively short time before. In this cases, we want to show that knowing the radio schedule can help improving the results when detecting entities.

The *tweets* generated by the bot present a predefined structure, which facilitates the entity recognition. Indeed, as in the example of figure 5 they begin with "*Now playing",* followed by the list of contributors of the work, separated by comma. Then, after the hyphen-minus there is the name of the work, followed by a series of hashtags or other users citations, and finally a link to a page of the *BBC Radio 3* website.

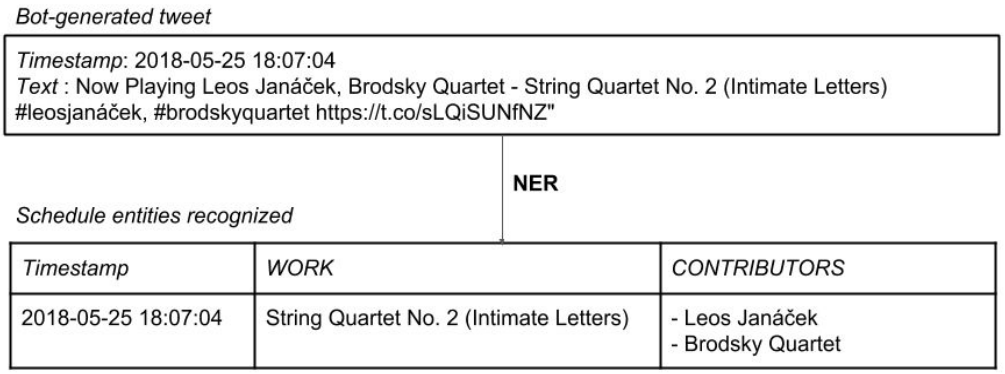

Figure 5: Example of entities recognition from a track of the radio schedule.

However, the number of entities within a *tweet* may vary because of different numbers of work contributors. Consequently, we choose to use also in this case the list of features presented in section 3.2 for building a probabilistic model able to predict the entities in the bot-generated *tweets*. It will be shown in chapter 4 that the precision of this model is almost perfect, and it will lead to generate a list of time-related entities that can be used in improving the detection in the user-generated *tweet*.

3.4.1. Matching Algorithm

Once associated to each track a list of entities, we use the information extracted from the schedule to perform two types of matching. Firstly, within the tracks we identify the ones which have been played in a fixed range of time before the generation of the user *tweet*. It leads to have a list of candidates from one to maximum five tracks of the schedule. Afterwards, we search the entities associated to the tracks selected within the *tweet.* In figure 6, it is presented an example of entity found using this method.

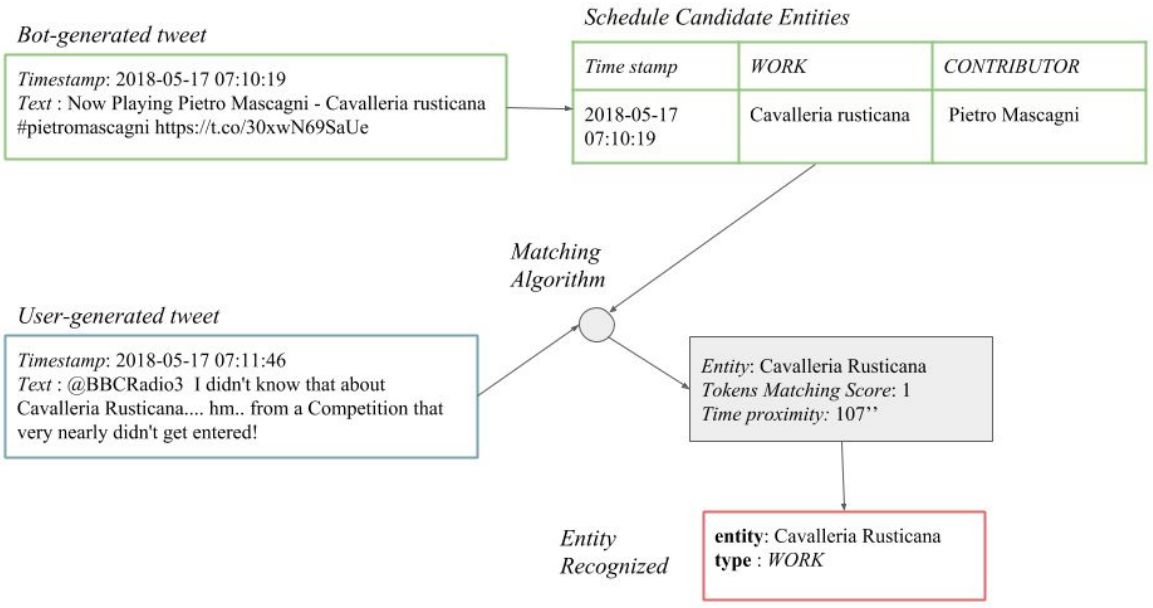

Figure 6: Example of the workflow for recognizing entities in user-generated content using the information from the radio schedule.

The search of the entities from the schedule within the user-generated *tweets* is based on a token-level string matching. The score of this matching is computed as the ratio of the number of tokens in common between an entity and the input *tweet* and the total number of token of the entity:

$$score = \frac{\#(tokens\ NE \cap input\ tweet)}{\#\ tokens\ NE}$$

For instance, if a schedule entity is composed by two tokens, if both are in the user-generated *tweet*, the score of the matching is 1. If only one of the two token is found into the target *tweet* the score is 0.5.

In order to exclude trivial matches, tokens within a list of stop words are not considered while performing this matching. The list is composed by a mix of english, italian, french, german and spanish common words (pronouns, prepositions, definite articles), which has been found to be commonly used in the musical work titles. The final score is a weighted combination of the token-level matching score and the time proximity of the track, aimed to enhance matches from tracks played closer to the time when the user is posting the *tweet*.

3.5. Entities Recognized

The output of the system is the list of entities recognized in the user-generated tweets, both using the ML model and the matching algorithm. The evaluation of the two parts of the system is presented in section 4.2. When evaluating the performances of the two methods together, an entity is considered correctly recognized if at least one method recognizes the entity.

The entities are recognized at a token-level, hence if an entity in the text is formed by two tokens, if both are recognized they are considered as two separate entities. More sophisticated approaches able to identify multi-token entities, labelling each token using Inside-Outside-Beginning (IOB) tags, have not been considered in this work.

4. Evaluation and Results

In this chapter, the performances of the proposed NER system are presented. Firstly, using k-fold cross validation we compare the results of the Machine Learning techniques listed in section 3.3. Afterwards, the complete system is evaluated on a test set of 100 user-generated *tweets* randomly chosen.

4.1. K-fold Cross Validation

K-fold Cross validation is a technique for evaluating the general performances and the accuracy of statistical models [Kohavi, 1995]. In k-fold cross validation, one breaks a dataset into k parts. For the first fold, take the 1st subset for testing, and the other k-1 for training the classifier. For the second fold, take the 2nd subset for testing, and the others for training, and so on. These k folds are then independently trained and tested, and the results are averaged. We choose to use 10 folds for performing cross validation. In table 5, *precision, recall* and *F-measure* values obtained are presented.

		User-generated			**Bot-generated**	
	Algorithm/Entity	*Contributor*	*Work*	*Other*	*Contributor*	*Work*
Precision	*SMO*	**0.782**	**0.784**	0.983	**0.984**	**0.979**
	iBk	0.647	0.503	0.969	0.943	0.976
	NaiveBayes	0.535	0.258	**0.983**	0.959	0.895
	J48	0.713	0.561	0.951	0.975	0.959
Recall	*SMO*	**0.804**	**0.687**	0.985	**0.984**	**0.979**
	iBk	0.637	0.531	0.968	0.983	0.922
	NaiveBayes	0.786	0.505	0.922	0.915	0.949
	J48	0.581	0.116	**0.986**	0.968	0.968
F-measure	*SMO*	**0.793**	**0.733**	**0.984**	**0.984**	**0.979**
	iBk	0.642	0.517	0.968	0.963	0.948
	NaiveBayes	0.637	0.342	0.952	0.937	0.921
	J48	0.64	0.193	0.968	0.972	0.963

Table 5: Summary of the results obtained for the considered techniques. In bold, the best performances for the Precision, Recall and F-measure measures.

We can see how in both user-generated and bot-generated corpora, the *Contributor* and *Work* entities are better recognized using the Support Vector Machine technique. It has been proven that this algorithm gives better scores than conventional systems while performing NER [Isozaki & Kazawa, 2002], and our results reflect this behaviour.

However, from table 5 we can deduce more interesting facts. Firstly, the performances of the model built with bot-generated content are much better than the ones of the case of user-generated model. Indeed, while recognizing the entities associated to the schedule, we reach a 0,98 precision and recall for *Contributor* entities, and almost 0,98 in the case of *Work*.

SMO			
Contributor	*Work*	*Other*	*<-- classified as*
491	19	101	*Contributor*
38	189	48	*Work*
99	33	8745	*Other*
NaiveBayes			
Contributor	*Work*	*Other*	*<-- classified as*
480	60	71	*Contributor*
68	139	68	*Work*
349	340	8188	*Other*
iBk			
Contributor	*Work*	*Other*	*<-- classified as*
389	35	187	*Contributor*
37	146	92	*Work*
175	109	8593	*Other*
J48			
Contributor	*Work*	*Other*	*<-- classified as*
355	7	249	*Contributor*
40	32	203	*Work*
103	18	8756	*Other*

Table 6: Confusion matrices of the user-generated model, obtained using 10-fold cross validation with the selected techniques.

It can be considered as a consequence of the predefined structure of the bot-generated *tweets*, where the *Contributor* entities are located always before the *Work* entities. Moreover, in the bot-generated *tweets* formal language is used, which means that entities are always presented in their full form, so it is easier to recognize them in comparison to the case of user-generated content.

The decrease of the performances of the user-generate model can be explained by several factors. Surely, the informal language used when posting a *tweet* influences negatively the entity recognition. In addition, in the user-generated corpora it has been necessary to annotate the entities, which are not *Contributor* or *Work*, as *Other*. The effect of adding this other kind of entity is that it gets more difficult to make predictions, and the uncertainty can be observed in the confusion matrices, presented in table 6. However, even for the user-generated *tweets* the precision and recall are around 0,8, which implies that entities are recognized more than randomly.

Moreover, evaluating the attributes used in the model drives us to other conclusions too. In table 7, the top five attributes selected are shown. In both cases, the POS tag is the most relevant feature, carrying a linguistic information proved to be important while recognizing the entities. However, from the differences between the two lists we obtain more information. In fact, in the case of the bot-generated corpora, the position of the token is the second most useful attribute. As discussed before, the fixed structure of the *tweets* created by the bot is an helpful characteristic, and the attribute evaluation confirms this.

In the user-generated model, we see how the gazetteers of last names and work types helped the algorithm in detecting named entities. The importance of the last names gazetteer is considered major than the first names one, probably due to the fact that users when citing a *Composer* often use only the last name. Furthermore, detecting if a token is a type of work or an instrument also is evaluated as useful while searching for *Work* entities.

User-generated	**Bot-generated**
POStag	*POStag*
iscapital	*position*
islastname	*iscapital*
isworktype	*isfirstname*
isinstrument	*islastname*

Table 7: List of top results from attribute evaluation.

Finally, we can state that generally *Contributor* entities are easier to recognize than *Work* entities. The reason behind this difference relies on the length and mixed structure of the *Work* entities. Generally, when referring to a contributor it is used the first name in combination with the last name, or alternatively just the last name. The nomenclature of the *Work* entities is frequently more complex, and it includes common words that can be easily misclassified as other entities.

4.2. System Evaluation

In the previous section, we have seen how SVM model outperforms the other algorithms considered, basing on the 10-fold cross evaluation. Starting from this result, we use it for recognizing named entities in a test set of 100 *tweets* manually annotated. In addition, we apply the matching algorithm presented in section 3.3.2, obtaining a complete evaluation of the method proposed. In table 8, results obtained are shown.

The performances of the SVM model decrease consistently when applied to the test set. Especially in the case of *Work* entities, it seems to not be able to detect most of them, according to the extremely low recall obtained. On the contrary, precision is higher than in the case of *Contributor* entities, where the high rate of false positives leads to a not accurate recognition. However, we can see how the matching against the schedule helps to achieve better results for both kinds of entity.

	Contributor		
	Precision	*Recall*	*F-Measure*
SVM	0.495	0.616	0.549
SVM + Schedule Matching	0.520	0.700	0.600
	Work		
	Precision	*Recall*	*F-Measure*
SVM	0.500	0.022	0.043
SVM + Schedule Matching	0.800	0.080	0.145

Table 8: Summary of the results obtained for the complete system.

The performances of the matching algorithm are consequence of the choice of two parameters: the maximum time-distance allowed between the creation of the *tweet* and the time when a track is broadcasted, and the minimum string similarity between the entities associated to the schedule and the text posted by the users.

Our system is based on the tradeoff of these two parameters, allowing the time-matching of the tracks from the schedule within a range of 15 minutes before the *tweets* is posted, and requiring a minimum of 0,5 string similarity between the entity from the schedule and the text.

From what we have observed, increasing the time range leads to have more entities recognized, but at the same time increases the rate of false positive. Decreasing the threshold score while performing string-matching leads to similar consequence. On the contrary, choosing a time window too small or requiring a very high string similarity, leads to an increase of the false negatives predictions.

Another cause of false positive while performing string-matching is the multivalence of some words. Thanks to the list of stop-words created specifically for our case, which are excluded when performing the matching, we have been able to limit partially this problem.

In conclusion, results shown that the NER proposed have benefited from the use of the schedule. However, our approach present also some limitations, which are discussed in the next chapter.

5. Conclusion and Future Works

5.1. Conclusion

We have presented in this work a novel method for performing NER in user-generated content, modelling linguistic features with SVM technique, and using external contextual information for enhancing the results. We analyzed *tweets* related to a classical music radio station, integrating its schedule to connect user messages to tracks broadcasted. We focus on the recognition of two kinds of entity related to the classical music field, *Contributor* and *Work.*

We centered our interest around the music radio channel, *BBC Radio 3*, as representative source of user-generated content related to the classical music field. However, our hypothesis is that the method proposed can be applied to other radio channels for which is possible to retrieve the schedule. In our case, we use *tweets* automatically generated by *BBC Radio 3 Bot* for retrieving the tracks played on the radio.

We approach the problem of NER in user-generated content in two ways: on one side, using SVM technique, we created a model able to predict the presence of named entities, based on linguistics features shaped for characterizing the informal language used by the users. On the other side, thanks to a matching algorithm based on time proximity and string similarity, we located *tweets* where users refer specifically to tracks recently played. Combining the information extracted, we have been able to create a system capable of detecting named entities in user-generated content.

Some of the features selected in our work are tailored to the case of classical music, hence they might not be representative if applied to other fields. We do not exclude that our method can be adapted for detecting other kinds of entity, but it might be needed to redefine the features according to the specific case considered.

According to the results, we have seen a tangible difference between the system performances when dealing with the *Contributor* instead of the *Work* entities. The former type of entity has been shown to be more easily detected in comparison to the latter. We identify several reasons behind this fact. Firstly, *Composer* entities are less prone to be shorten or modified, while due to their longness, *Work* entities often represent only a part of the complete title of a musical piece. Furthermore, work titles are typically composed by more tokens, including common words which can be easily misclassified as *Other* entities. The low *recall* score obtained in the case of *Work* entities might be explained by these observations. On the contrary, when referring to a *Composer* users often use enough information to the system for detecting the entity, as for instance only the surname.

The schedule information used also present several limitations. In fact, the hypothesis that a *tweet* is referring to a specific track broadcasted is not always verified. Even if it is common that radios listeners do comments about tracks played, or give suggestion to the radio host about what they would like to listen, it is also true that they might refer to a specific *Contributor* or *Work* unrelated to the radio schedule. In addition, the possibility to listen the radio program not when it is live broadcasted using a web player, restrict the possibility to create temporal links between *tweets* and tracks.

5.2. Future Works

Starting from the limitations of our approach, we can define some guidelines for improving the method proposed. First of all, the recent developments of neural network architectures lead us to suppose that modelling the informal language characterizing the user-generated content with Deep Learning techniques could improve the recognition of named entities. As shown in [Lample et al., 2016] and [Lin et al., 2017], Long Short-Term Memory (LSTM) units used for building layers of a recurrent neural network (RNN) are being proven to be effective when applied to NLP tasks, such as NER.

Furthermore, there are several string metrics proposed in the computer science literature which hypothetically can perform a more accurate matching between the schedule and the

input *tweets*. Testing some others measures can be useful for understanding which one is the more efficient for this task.

In addition, it could be interesting to include a clustering method for tackling the problem of nicknames and variations of contributor and work names written by the users. In fact, there are several forms which are popularly used in the classical music field, as instance "*Father of the Pianoforte*" while referring to "*Muzio Clementi",* or "*The Blue Danube*" when talking about "*An der schönen blauen Donau*", *Op. 314* by Johann Strauss II. The use of clusters containing these kinds of variations could help the NER system in being more effective, especially when dealing with user-generated content.

From a MIR perspective, other types of musical information can be integrated for improving the performance of the system. For example, in the matching algorithm, instead of considering a fixed maximum threshold for the time-proximity, it can used the specific duration of each track as time range in which search connection with the *tweets*. Indeed, considering the variety of musical pieces' durations, fixing a global threshold might incorrectly limit the search space and consequently lead to an increase of false negatives.

Moreover, using other kinds of metadata related to the tracks, easily retrievable from services as MusicBrainz, also can help the accuracy of the NER. As instance, knowing the instrumentation of a piece, could help identifying the *Contributor* entities, i.e. if we know that a piece is a piano solo, we can expect that the performer is likely to be a single person and not a string quartet.

Lastly, the approach can be generalised for accepting as input different kinds of user-generated content, always when there is present a structured time schedule, which can enrich the context while texts are generated.

6. List of figures

7. List of tables

8. List of abbreviations

EL	Entity Linking
IE	Information Extraction
IOB	Inside-Outside-Beginning
KB	Knowledge Base
LSTM	Long Short-Term Memory
MIR	Music Information Retrieval
ML	Machine Learning
MNE	Musical Named Entity
MUCs	Message Understanding Conferences
NED	Named Entity Disambiguation
NEN	Named Entity Normalization
NER	Named Entity Recognition
NLP	Natural Language Processing
NLU	Natural Language Understanding
OOV	Out-Of-Vocabulary
POS	Part-Of-Speech
RNN	Recurrent Neural Network
SVM	Support Vector Machine
UGC	User-generated Content

9. Bibliography

[Aha & Kiblerm, 1991] Aha D. & Kibler D. (1991). Instance-based learning algorithms. Machine Learning. 6:37-66.

[Andersen et al., 1992] Andersen, P. M., Hayes, P. J., Huettner, A. K., Schmandt, L. M., Nirenburg, I. B., & Weinstein, S. P. (1992). Automatic Extraction of Facts from Press Releases to Generate News Stories. Proceedings of the Third Conference on Applied Natural Language Processing -, 170–177.

[Berners-Lee, 1999] Berners-Lee, T. .Weaving the Web. Orion Business Books, 1999.

[Bontcheva et al., 2013] Bontcheva, K., Derczynski, L., Funk, A., Greenwood, M. a, Maynard, D., & Aswani, N. (2013). TwitIE : An Open-Source Information Extraction Pipeline for Microblog Text. Proceedings of Recent Advances in Natural Language Processing, (September), 83–90.

[Cardie, 1997] Cardie, C. (1997). Empirical Methods in Information Extraction. AI Magazine, 18(4), 65–79.

[Cowie & Lehnertand, 1996] Cowie, J., & Lehnertand, W. (1996). Information extraction. Communications of the ACM, 39(1).

[Cucerzan, 2007] Cucerzan, S. (2007). Large-Scale Named Entity Disambiguation Based on Wikipedia Data. EMNLP-CoNLL 2007, 708–716.

[Cunningham et al., 2001] Cunningham, H., Maynard, D., Bontcheva, K., & Tablan, V. (2001). GATE: an Architecture for Development of Robust HLT Applications. Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL ’02, (July), 168.

[Elhadad et al, 2014] Elhadad, N., Gravano, L., Hsu, D., Balter, S., Reddy, V., & Waechter, H. (2014). Information Extraction from Social Media for Public Health. KDD at Bloomberg: The Data Frameworks Track, (July), 1–4.

[Derczynski et al., 2015] Derczynski, L., Maynard, D., Rizzo, G., Van Erp, M., Gorrell, G., Troncy, R., Bontcheva, K. (2015). Analysis of named entity recognition and linking for tweets. Information Processing and Management, 51(2), 32–49.

[Ferragina and Scaiella, 2012] Ferragina, P., & Scaiella, U. (2012). Fast and Accurate Annotation of Short Texts with Wikipedia Pages. IEEE Software, 29(1), 70–75.

[Frank et al., 2016] Frank, E., Hall, M. A., Witten, I. H. (2016). The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition.

[Gimpel et al., 2011] Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Smith, N. A. (2011). Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Short papers, (2), 42–47.

[Habib & Keulen, 2014] Habib, M. B., & Keulen, M. Van. (2014). Information Extraction for Social Media. Workshop on Semantic Web and Information Extraction, (July), 9–16.

[Han & Baldwin, 2011] Han, B., & Baldwin, T. (2011). Lexical Normalisation of Short Text Messages : Makn Sens a # twitter. Computational Linguistics, V(212), 368–378.

[Hauger et al., 2013] Hauger, D., Schedl, M., Košir, A., & Tkalcic, M. (2013). The Million Musical Tweets Dataset: What Can We Learn From Microblogs. Proceedings of ISMIR 2013.

[Isozaki & Kazawa, 2002] Isozaki, H., & Kazawa, H. (2002). Efficient Support Vector Classifiers for Named Entity Recognition. Proceedings of the 19th international conference on Computational linguistics,, 1, 1–7.

[Jijkoun et al., 2008] Jijkoun, V., Khalid, M. A., Marx, M., & de Rijke, M. (2008). *Named Entity Normalization in User Generated Content Categories and Subject Descriptors.* Proceedings of the Second Workshop on Analytics for Noisy Unstructured Text Data, 23–30.

[John & Langley, 1995] John, G. H. & Langley, P., (1995). Estimating Continuous Distributions in Bayesian Classifiers. Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, 338-345.

[Jurafsky & Martin, 2009] Jurafsky, D., & Martin, J. H. (2009). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Upper Saddle River, N.J: Pearson Prentice Hall.

[Kohavi, 1995] Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In IJCAI, volume 14, pages 1137–1145, 1995.

[Lample et al., 2016] Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural Architectures for Named Entity Recognition. Proceedings of the

2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (July).

[Lehnert & Sundheim, 1991] Lehnert, W., & Sundheim, B. (1991). A Performance Evaluation of Text Analysis Technologies. AI Magazine, 12(3), 81–94.

[Lin et al., 2017] Lin, B. Y., Xu, F. F., Luo, Z., & Zhu, K. Q. (2017). Multi-channel BiLSTM-CRF Model for Emerging Named Entity Recognition in Social Media. Proceedings of the 3rd Workshop on Noisy User-Generated Text, 160–165.

[Liu et al., 2011] Liu, X., Zhang, S., Wei, F., & Zhou, M. (2011). Recognizing Named Entities in Tweets. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL), 1(2008), 359–367.

[Mendes et al., 2011] Mendes, P. N., Jakob, M., García-silva, A., & Bizer, C. (2011). DBpedia Spotlight : Shedding Light on the Web of Documents. Proceedings of the 7th International Conference on Semantic Systems (I-Semantics)., 95, 1–8.

[Moro et al., 2014] Moro, A., Raganato, A., & Navigli, R. (2014). Entity Linking meets Word Sense Disambiguation: a Unified Approach. Transactions of the Association for Computational Linguistics (TACL), 2(0), 231–244.

[Nadeau, 2007] Nadeau, D. (2007). A survey of named entity recognition and classification. Linguisticae Investigationes, (30), 3–26.

[Oramas et al., 2015a] Oramas, S., Ostuni, V. C., Di Noia, T., Serra, X., & Di Sciascio, E. (2015). Sound and Music Recommendation with Knowledge Graphs. ACM Transactions on Intelligent Systems and Technology, 9(4).

[Oramas et al., 2015b] Oramas, S., Sordo, M., Espinosa-Anke, L., & Serra, X. (2015). A Semantic-based Approach for Artist Similarity. Ismir 2015, (October), 100–106.

[Oramas et al., 2016a] Oramas, S., Espinosa-anke, L., Lawlor, A., Serra, X., Saggion, H., (2016). Exploring Customer Reviews for Music Genre Classification and Evolutionary Studies. Proc. 17th International Society for Music Information Retrieval Conference, 150–156.

[Oramas et al., 2016b] Oramas, S., Espinosa-Anke, L., Sordo, M., Saggion, H., & Serra, X. (2016). ELMD: An Automatically Generated Entity Linking Gold Standard Dataset in the Music Domain. Language Resources and Evaluation Conference - LREC, 3312–3317.

[Oramas et al., 2016c] Oramas, S., Espinosa-Anke, L., Sordo, M., Saggion, H., & Serra, X. (2016). Information extraction for knowledge base construction in the music domain. Data and Knowledge Engineering, 106, 70–83.

[Oramas et al., 2017] Oramas, S., Ferraro, A., Correya, A., & Serra, X. (2017). Mel: a Music Entity Linking System. 18th International Society for Music Information Retrieval Conference, Suzhou, China, 2017.

[Pang & Lee, 2006] Pang, B., & Lee, L. (2006). Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 1(2), 91–231.

[Platt, 1998] Platt, J. (1998). Fast Training of Support Vector Machines using Sequential Minimal Optimization. B. Schoelkopf and C. Burges and A. Smola, editors, Advances in Kernel Methods - Support Vector Learning

[Piskorski & Yangarber, 2013] Piskorski, J., & Yangarber, R. (2013). Information Extraction: Past, Present and Future. Multi-Source, Multilingual Information Extraction and Summarization, 23–50.

[Quinlan, 1993] Quinlan, R. (1993). C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA.

[Ramshaw & Marcus, 1995] Ramshaw, L. A., & Marcus, M. P. (1995). Text Chunking using Transformation-Based Learning. ACL Third Workshop on Very Large Corpora, 82–94.

[Ritter et al., 2011] Ritter, A., Clark, S., & Etzioni, O. (2011). Named entity recognition in tweets: an experimental study, Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, 1524–1534.

[Schedl & Hauger, 2012] Schedl, M., & Hauger, D. (2012). Mining microblogs to infer music artist similarity and cultural listening patterns. International Conference Companion on World Wide Web (WWW), 877.

[Small & Medsker, 2014] Small, S. G., & Medsker, L. (2014). Review of information extraction technologies and applications. Neural Computing and Applications, 25(3–4)

[Sordo et al., 2012] Sordo, M., Serrà, J., Koduri, G. K., & Serra, X. (2012). A Method For Extracting Semantic Information From On-line Art Music Discussion Forums. 2nd CompMusic Workshop, 55–60.

[Statistic Brain, 2016] Statistic Brain (2016), Twitter Statistic, Statistic Brain Research Institute, publishing as Statistic Brain. (September), https://www.statisticbrain.com/twitter-statistics/

[Tata & Di Eugenio, 2010] 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, (July), 1376–1385.

[Whitman & Lawrence, 2002] Whitman, B., & Lawrence, S. (2002). Inferring descriptions and similarity for music from community metadata. Proceedings of the 2002 International Computer Music Conference, 591–598.

[Zangerle et al, 2012] Zangerle, E., Gassler, W., & Specht, G. (2012). Exploiting Twitter ' s Collective Knowledge for Music Recommendation. 2nd Workshop on Making Sense of Microposts #MSM2012, (February), 14–17.

[Zangerle et al., 2014] Zangerle, E., Pichl, M., Gassler, W., & Specht, G. (2014). #nowplaying Music Dataset. Proceedings of the First International Workshop on Internet-Scale Multimedia Management - WISMM '14, 21–26.

[Zhang et al., 2009] Zhang, X., Liu, Z., Qiu, H., & Fu, Y. (2009). A hybrid approach for chinese named entity recognition in music domain. 8th IEEE International Symposium on Dependable, Autonomic and Secure Computing, DASC 2009, 677–681.

[Zhang et al., 2011] Zhang, J., Lin, Y., Gong, X., Qian, W., & Zhou, A. (2011). Unsupervised user-generated content extraction by dependency relationships. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6997 LNCS, 116–128.