POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Adaptive Management of Multimedia and Georeferenced Contents: the MAGIS Approach for Citizen Journalism

TESI DI LAUREA MAGISTRALE IN
MANAGEMENT ENGINEERING – INGEGNERIA GESTIONALE

Author: **Elisa Rossi**

Student ID:	944750
Advisor:	Mariagrazia Fugini
Co-advisor:	Jacopo Finocchi
Academic Year:	2020-2021

Abstract

The present dissertation illustrates a web-based framework where geographic maps are associated to knowledge, linking geographic elements with geo and temporal-referenced multimedia contents. Among the key features, multimedia content integration, time management and self-adaptivity to different application contexts are the most relevant ones. A hybrid approach is proposed to classify stored contents, that combines a Machine Learning (ML) technique for text classification with the intervention of human experts. A dynamically configured user navigation is hence proposed based on the adaptivity of the tool to dynamic classification of contents. Adaptive navigation is supported by a domain-specific ontology defined in this Thesis. The overall approach is presented in a framework and prototype named *Multimedia Adaptive Geographic Information System* (MAGIS), representing an extension to existing Geographic Information Systems (GIS), and conceptually devised as a common foundation to be adapted to different domain-specific implementations. A practical demonstration in the form of a web-based prototype is provided in one context, namely *Citizen Journalism* (CJ), where users (both common and specialised) can tailor their navigation according to selected topics. The purpose of the CJ use case described in the Thesis is to detail how the framework gets instantiated in a semi-automatic way using both ML and human intervention.

Keywords: Geographic Information System, Framework, Multimedia, Adaptivity, Ontology, Machine Learning and Classification, Tags.

Abstract in Lingua Italiana

La presente Tesi illustra le caratteristiche di base di un framework in cui la conoscenza è associata a mappe geografiche, con l'obiettivo di collegare elementi geografici con contenuti multimediali referenziati nello spazio e nel tempo. Tra le funzionalità chiave, l'integrazione di contenuti multimediali, la gestione del tempo e l'auto-adattamento ai diversi contesti applicativi sono le più rilevanti. Viene proposto un approccio ibrido per classificare i contenuti archiviati, che combina una tecnica di Machine Learning (ML) per la classificazione dei testi con l'intervento di un esperto. Viene quindi proposta una navigazione utente configurata dinamicamente in base all'adattabilità dello strumento alla classificazione dinamica dei contenuti. La navigazione adattiva è supportata da un'ontologia specifica del dominio definita in questa Tesi. L'approccio generale è presentato in un framework e prototipo denominato *Multimedia Adaptive Geographic Information System* (MAGIS) che rappresenta un'estensione dei Sistemi Informativi Geografici (GIS) esistenti ed è concepito concettualmente come base comune per essere adattato a diverse implementazioni specifiche del dominio di interesse. Una dimostrazione pratica sotto forma di prototipo basato sul web è fornita in un contesto di esempio, chiamato *Citizen Journalism* (CJ), o giornalismo partecipativo, in cui gli utenti (sia comuni che specializzati) possono personalizzare la propria navigazione in base ad argomenti selezionati. Lo scopo del caso d'uso CJ descritto nella Tesi è quello di dettagliare come il framework viene istanziato in modo semiautomatico utilizzando sia il ML che l'intervento umano.

Keywords: Sistema Informativo Geografico, Framework, Multimedia, Adattabilità, Ontologia, Classificazione e Machine Learning, Tag.

Contents

Introduction

Among the most recent aims of geographic-based applications, the aim of enriching maps with georeferenced contents is prominent, allowing one to explore selected areas and run thematic, multilevel analyses [1]. This is achieved by supporting exploration of various levels of interest and many zoom levels: from the largest one, enriching navigation by accessing more linked content, to the narrowest one, focusing on details of interest and limiting the so-called *information overload*. Some features are still missing from existing applications, such as the automatic classification of added contents and the dynamic grouping of contents based on their topic. Another issue is to support collaborative work in applications such as urban planning or collaborative research, or in highly dynamic contexts, like citizen journalism, where the contents vary in a seamless way.

This dissertation proposes a framework called MAGIS (*Multimedia Adaptive Geographic Information System*) to associate structured and unstructured data, specifically multimedia data, to geographic maps, linking cartographic elements with geo and temporally referenced contents via geotags and temporal tags. The result is a web environment that can be navigated both geographically and temporally. The framework is a logical structure designed to be adaptable to different application areas or contexts (i.e., semantic domains), such as history, material and immaterial cultural heritage, architecture and urban planning, business intelligence, digital twinning of devices and artefacts, and so on.

The key innovative features can be summarised as follows:

1. *Handling multimedia content*. The integration of structured data into geographic maps is already achieved by effective and

widely used solutions, such as map thematization and choropleth maps [2]. However, in existing systems, data navigation is typically carried out by applying filters on measurable values, and data analysis can be performed using statistical and analytical tools. This proposal aims at moving a step ahead by *integrating cartography with unstructured data*, which appears as a less established area, and at enhancing Business Intelligence analysis in these systems. To this aim, the framework *combines structured data* – such as demographic, economic, or historical data coming from structured data sources – *with multimedia content* – such as pictures, audios, images, music – together with a set of metadata describing the content item. Unstructured data can be collected and extracted from external sources or provided by "volunteer" users. For example, one content can be a geo-area containing structured data related to on-site population, statistical data about demography, forecast data about the evolution of GDP, plus unstructured data, such as satellite photos, YouTube videos about the history of the area, images uploaded by inhabitants, or blog with text and images about a topic related to the area. Georeferenced multimedia data are particularly important in fields like World News, Intangible Cultural Heritage (ICH), Neo-Geo, Demographic, Ethnographic or Anthropological Analysis, not to mention areas like tourism, culture, museal and musical sciences. Each area has the currently the necessity of bringing together heterogeneous types of content, like textual posts, pictures, videos and audio recordings, narrations, virtual reality, and 3D data, and so on.

2. *Temporal dimension*. Besides the geographic location of contents, the framework aims at displaying the chronological placement of contents along time. For instance, it will be possible to retrieve the content associated with a geographical area in a certain historic period or to discover how an element on the map – e.g., a building – has being evolving along time analysing pictures and documents about style variations through human interventions. Temporal information, which constitutes *added knowledge*, can be present optionally, or only

in some application domains, e.g., in contexts such as history events or journal news, for which the chronological contents are inherently embedded.

3. *Dynamically layered representation.* Contents are organised in thematic layers which can represent categories of related information, or any other cluster of information relevant as a group. Layers are hierarchically organised and dynamically generated according to domain-specific contents. The organisation into layers, made possible through some dynamic aggregation criteria, reduces the multitude of objects appearing on the map, facilitating content navigation, and reducing information overload that would occur if all the content items were shown at the same time.
4. *Automatic Machine Learning (ML)-based content classification*. After a first training provided by a human operator, the content is automatically classified by a supervised ML algorithm into domain-specific classes of objects by means of semantic tags. This allows to automatically adapt the navigation interface to the available contents, to achieve the system adaptivity to different contexts and facilitating the retrieval of information. The final objective is to mitigate the information overload that would occur when showing all the content items at the same time.

These key characteristics contribute to make the MAGIS framework *self-*a*daptable* to different application areas. By *instantiation* of MAGIS , designers of a multimedia-based geographic information system can develop an application addressing a specific domain context. The framework comprises a meta-structure (a predefined model of concepts valid for all contexts, such as “content item”, “geo object”, “point of interest”), and a set of context-related parameters (such as “historic period” for *Cultural Heritage* context, “news source”, “reportage”, “magazine” for the *News* context) which allow the framework to be instantiated via self-adaption to different contexts, based on automatic content classification. For example, the model can be adapted to the *Precision Agriculture* context using pictures of the different solutions applied in cultivated areas, videos for e-learning, and so on.

The objective of the present dissertation is to lay the foundations of a general framework that is modular and adaptable to different uses and applications. The framework will be presented in all its modules, but it will be then implemented just in some parts as a practical demonstration. As a general overview, the framework is expected to include a map layer that will be the foundation of the geographic-based application, and a set of modules that will manage additional contents related to the map itself.

This structure has been inspired by the quite recent development of the so-called Multimedia Geographic Information Systems (MM-GIS), which in turn derive from the adaptation of traditional GIS to unstructured content. The framework presented here is called *MAGIS*, which stands for "*Multimedia Adaptive Geographic Information System*".

Rather than relational data management, what is outlined is an ontology for cartographic knowledge (geographical objects, georeferenced data, and relationships with each other) that allows semantic analysis, navigation, and dynamic maintenance of the content associated with the map. The use of an ontology is required having to deal with multimedia and, more in general, unstructured contents (including the temporal dimension). In this sense, the adoption of an ontology facilitates the exploration of associated knowledge, by considering correlations and logical connections between content items. Adopting a general ontological structure (*meta-structure*) as starting point for the implementation of the framework facilitates the instantiation of a domain-specific relational model for the development of the application at hand.

The MAGIS approach combines expert-driven and supervised ML-driven classification. A domain expert defines an ontology for context-specific semantic categories, and manually classifies a training set of contents. The contents collected afterwards are automatically classified using the ML algorithm, associating each content to an ontology node.

Classifying contents into domain-specific categories by means of meaningful semantic tags is aimed to build a well-organised repository of geographic-related information, facilitating content retrieval, and reducing information overload. Besides supporting context-adaptive presentation, the ontological classification also helps to achieve a language-independent content navigation, that often represents an issue in geographic systems [3].

The separation among distinct topics is meant to facilitate the navigation of the map by giving the possibility to *filter* the elements of interest, or to *cluster* them according to semantic user-defined criteria, so reducing information overload. As said, the navigation of multimedia content is organised in several layers representing different topics. The basic layer is the map itself where cartographic objects are visualised. Layers are dynamic and not predefined, taking into account available contents and user preferences. The deployment of different layers requires the contents to be properly *classified*. To be effective, this classification must be specific to the application context, namely it must be based on context-specific tags.

The solution proposed in this dissertation is a ML software prototype that considers various domain contexts, such as urban planning, history, citizen journalism and participatory design, which can be used by experts and be enriched by users/citizens via specific upload tools, which also validate data before they are definitively stored (we reference ILAUD [4] for the grounding themes about generation and use of maps for participatory urban design).

The dissertation is structured as follows. Chapter 1 reviews existing literature and current map-based applications. Chapter 2 presents the objectives of the MAGIS project and its main features. Chapter 3 expands the concept of ontology and shows its implementation. Chapter 4 illustrates an application of MAGIS, namely a web-based application prototype which makes the theoretical framework concrete. Chapter 5 describes in detail the approach to define the proper classification settings for the ML algorithm and its application in the prototype. Chapter 6 concludes the dissertation showing present issues and future developments.

1 Related Work

Nowadays, geographic-based applications are a widespread object of study. Developers are implementing many different solutions that combine geo-referenced information related to a map, ending up with maps that can be used either as general interest exploration tools or as professional tools for advanced analyses purposes. Many geographic-based tools have been proposed in the literature in different contexts, each based on thematic contents to address specific needs.

The development of a geographic-based application involves four relevant sets of activities [5]. First, *data acquisition* is aimed to acquire the data of the map and the additional features from different data sources. Second, *data representation* structures the acquired data into a unified formal codification that is easier to be managed. Third, *data analysis* is used to manipulate and query the data to extract useful information and advanced knowledge. Fourth, *data visualisation* is aimed to make the output of the analysis available to end-users through a proper visual representation that can be navigated in an interactive way.

In the next paragraphs, current approaches and existing applications related to the four phases are presented.

1.1 Data Acquisition

Data Acquisition aims to acquire map data and additional features, which means extracting data from their original source and encoding them into the destination representation format.

Usually, sources are *digitized maps*, either cartographic (ancient and modern maps), cadastral or photographic, and collections of *georeferenced data* of different types (economic, demographic, environmental, etc.). It is possible to acquire data either *automatically* or *interactively*. In the first case, automatic tools recognise the features of data coming from external sources, either structured or unstructured (such as archives, databases, social platforms, websites etc.) and autonomously import them into the destination storage system. In the second case, a human operator recognises and manually records the significant elements of the map and possibly enters additional information as annotations.

An automatic approach to data acquisition is building the map exploiting data directly gathered from the field. It is the case of *Internet of Things* (IoT) technologies applied to the so-called *Smart Cities*, where sensors are distributed across a defined territory to collect data from the environment. Less advanced applications can refer to simple GPS trackers or smartphones localisation systems that allow localising the device in the territory to deduce both static (e.g., paths or itineraries) and dynamic elements (e.g., traffic monitoring, path interruptions, flows of people through a certain point in town).

Another option is acquiring data from existing *cartographic datasets*. For instance, downloading maps and thematic datasets from a geographic information system [5] can be useful to gather a ready-to-use set of information to focus the attention on successive analyses. In this case, the acquisition software needs to convert the data from the original format of the different sources to a common destination format, considering the differences among the original sources; here a stage of data integration and data quality assessment is needed.

A further possibility is to rely on *open data sources* that include georeferenced data. One of the major examples is *OpenStreetMap*, a collaborative project aimed to create a free editable map of the world through the aggregation of geodata [6]. OpenStreetMap is the basis of many open-source websites that provide a wide variety of free map extracts in different formats, like *GeoFabrik* [7] or *BBBike* [8]. Other types of open data can be found on governmental data sources, such as *Eurostat.* [9], *European Union Open Data Portal* [10], or *US Data.gov* [11], which collect datasets on a national and international basis.

On the interactive side, it is possible to collect data provided by volunteers who offer their effort to map some aspects of a certain territory. Sometimes these data are called *Volunteered Geographic Information* (VGI), i.e., geospatial content generated by non-professionals using mapping systems available on the Internet, that offers possibilities for government agencies at all levels to enhance their geospatial databases [12]. End users can also contribute to improving the map features by signalling errors, variations, or integrations. For instance, the mobile application *Waze* strongly relies on the contribution of its users that signal traffic conditions along a certain itinerary. These types of applications are based on the so-called *participatory mapping* and *crowdsourcing* approach, which strongly requires a methodology to assess the accuracy of the crowdsourced information.

Based on the reliability of the data source and the data acquisition process, some *data quality assessment* may be necessary, especially in crowdsourced systems. This is to prevent the so-called *Garbage in – Garbage out phenomenon (GIGO)*: bad input data imply for sure a bad output. When collecting data from different sources, their integration should be done on good-quality data only, so that a data preparation phase must be run to clean data and solve potential conflicts and inconsistencies.

Finally, data acquisition is also linked to the definition of a *legal framework* to create a cartographic knowledge base. This must include elements like the privacy level of information, the security level of users, or the certification of the sources.

1.2 Data Representation

The aim of *Data Representation* (or *Data Modelling*) is to give a formal and codified representation of the significant information included in a map, namely places, elements, characteristics of the territory, but also additional information linked to the map which enrich cartography with georeferenced data (e.g., demographic, economic, cultural data etc.).

This type of information must be *geo-referenced*, meaning that it must be associated with geographic coordinates, either in two dimensions (e.g., latitude and longitude) or in three dimensions (e.g., including the elevation of the points of the territory). This is the basis of the so-called *Digital Cartography*, i.e., the digital version of a traditional geographic map where the position and the description of elements are stored in a set of digital files [13].

The significant elements of the map are usually graphical or spatial objects of *vectorial type*. *Points* typically represent punctual elements like a monument, a water source, or the pick of a mountain; *lines* represent streets, rivers, or railways, while *areas* can stand for buildings, cities, or forests. Each point is identified by its coordinates (X, Y, Z) and position [14]. More complex object types are *polygon collections* (e.g., boundaries of a country) [15].

The software platforms that are specialised in memorising and visualising geo-referenced data are called *Geographic Information Systems* (GIS). A GIS is a framework for gathering, managing, and analysing data. Rooted in the science of geography, GIS integrates many types of data. It analyses spatial location and organizes layers of information into visualizations using maps and 3D scenes. With this unique capability, GIS reveals deeper insights into data, such as patterns, relationships, and situations—helping users make smarter decisions [16].

GIS technology applies geographic science with tools for understanding and collaboration. It helps people reach a common goal: to gain actionable intelligence from all types of data. Hundreds of thousands of organisations in any virtual field are using GIS to make maps that communicate, perform analysis, share information, and solve complex problems around the world. In a GIS, maps are the geographic container for the data layers and analytics one wants to work with. GIS maps are easily shared and embedded in apps, and virtually accessible by everyone, everywhere. GIS integrates many kinds of data layers using spatial location. Most data have a geographic component. GIS data include imagery, features, and base maps linked to spreadsheets and tables.

The field of GIS started in the 1960s as early concepts of quantitative and computational geography emerged [17]. The first computerised GIS in the world is attributed to Roger Tomlinson, the pioneer who first worked to initiate, plan, and develop the *Canada Geographic Information System* in 1963. The Canadian government had commissioned Tomlinson to create a manageable inventory of its natural resources. He envisioned using computers to merge natural resource data from all provinces. Tomlinson created the design for automated computing to store and process large amounts of data, which enabled Canada to begin its national land-use management program. He also gave GIS its name.

Today, GIS gives people the ability to create their own digital map layers to help solve real-world problems. GIS has also evolved into a means for data sharing and collaboration, inspiring a vision that is now rapidly becoming a reality – a continuous, overlapping, and interoperable GIS database of the world, about virtually all subjects. Today, hundreds of thousands of organisations are sharing their work and creating billions of maps every day to tell stories and reveal patterns, trends, and relationships about everything.

Different GIS platforms are available today. One of the most common is *ArcGis* developed by the Californian ESRI (Environmental Systems Research Institute), that is declined into *ArcGIS Enterprise*, running on the client's infrastructure, either on-premises or in cloud, and *ArcGIS Online*, that is a Software-as-a-Service (SaaS) cloud solution [18]. The tools allow to build interactive maps to guide data exploration and visualisation and to make useful data analyses: reveal relationships, identify prime locations, use optimal routes, and analyse patterns to add valuable context to data and make predictions [19].

Another example is *GISMaker* by ProgeCAD [20], a software solution for elaboration and manipulation of geo-referenced geometric data, which is compatible with all GIS applications currently on the market. It is one of the few GIS in the global landscape including a native CAD. The program includes the typical benefits of CAD applications with the ones of the most known GIS, since it can import and export different file formats (e.g., shapefile) and to easily transform CAD models into GIS layers [21].

A third GIS tool that is common nowadays is *PostGIS*, a database extender for the open-source object-relational database *PostgreSQL* [22]. PostGIS adds extra types (geometry, geography, raster, and others) to the PostgreSQL database. It also adds functions, operators, and index enhancements that apply to these spatial types. These additional functions, operators, index bindings and types augment the power of the core PostgreSQL DBMS, making it a fast, feature-plenty, and robust spatial database management system [23].

A very widespread concept used in Data Representation is the *Point of Interest* (POI), a geo-referenced spatial point that is associated with its significant information. A POI is a specific point location that someone may find useful or interesting [24]. Most consumers use the term when referring to hotels, campsites, fuel stations or any other categories used in modern GPS navigation systems. A GPS point of interest specifies, at minimum, the latitude and longitude of the POI, assuming a certain map datum. A name or description for the POI is usually included, and other information such as altitude or a telephone number may also be attached. GPS applications typically use icons to represent different categories of POI on a map graphically.

One of the most interesting research fields connected to POIs is the *Point of Interest Recommendation*, aiming to provide personalised recommendations of places of interests, such as restaurants, for mobile users [25]. It is a significant task in location-based social networks (LBSNs) as it can help provide better user experience as well as enable third-party services, e.g., launching advertisements [26]. Due to the complexity of the task and the influence of many factors, such as user preferences, geographical influences, and user mobility behaviours, some studies propose a geographical probabilistic factor analysis framework that allows to capture the geographical influences on a user's check-in behaviour [25]. Since most part of literature considers all check-ins in a whole and their temporal relation is usually overlooked, further research deals with successive personalised POI recommendation in LBSNs, in order to provide good recommendation promptly based on users’ current status [26]. This task is much harder than standard personalised POI recommendation or prediction because it only recommends those locations that a user does not visit frequently or has not visited yet, but he/she may like to visit it at the successive time stamp; at the same time, it is much more valuable because of the personalised service.

POI data support a range of applications, including digital mapping, enhanced routing products and validation of private databases. POI mapping is very useful and efficient in digital map creation because it gives detailed route information with pictures of a target site. In short, it acts as a tool in navigation systems. Digital maps for current GPS devices usually include a simple range of POIs for the map area. On the other hand, some websites concentrate on the collection, authentication, management, and dissemination of POIs which end-users can load on their devices to replace or add on the existing POIs. End-users can also create their own custom POI collections. Saleable POI collections, especially those that craft with digital maps, or that are traded on a subscription basis are generally secured by copyrights [27].

Information related to POI is usually grouped into *layers*, levels of homogeneous data that are dedicated to specific subject areas. Types of information that can be associated with a map can be related to many different scopes and disciplines: archaeology and history, art and architecture, crafts and commercial activities, transports, tourism, agriculture and environment, cultural, recreational and sporting activities, politics (e.g., election results), economics (e.g., natural resources, incomes), nature (e.g., geology, botany, zoology), medicine (e.g., epidemiology, distribution of diseases), meteorology (e.g., temperatures, winds, rains), demography (e.g., inhabitants, ages, ethnic groups, languages, religions), etcetera.

The map can also be associated with *unstructured data*: typically, pictures, videos, or textual data (e.g., reporting that a path is obstructed). They may include satellite pictures of an area (e.g., Earth It [28]), 360° pictures (e.g., Street View [29]), videos recorded by drones, historical maps.

Conventional GIS data models provide a static representation of reality. Over years, developers have also implemented *four-dimensional maps* by including the changes of the map elements over time, to get the so-called *Temporal GIS* [30]. The incorporation of temporal components has been implemented with the relational model and then with the object-oriented data models in Computer Science (CS). Temporal information has been incorporated into GIS spatial data models by time-stamping layers (the snapshot models [31]), attributes (space-time composites [32]), and spatial objects (spatiotemporal objects [33]). In [34] the main achievements of spatiotemporal modelling in the field of Geographic Information Science are presented. The paper overviews Temporal GIS, spatiotemporal data models, spatiotemporal modelling trends and future trends in Temporal GIS.

Adding information about historic events, a *Historic Atlas* is derived to illustrate the evolution of phenomena characterising different geographical areas in a certain era. One example is *GeaCron*, an interactive Global Historic Atlas from 3000 A.C. [35]. Its mission is to make historical information universally accessible for everyone, through intuitive and attractive geo-temporal maps, as well as configurable timelines. According to its founder and director Luis Múzquiz, "*GeaCron intends to be a facilitator for those websites that have historical content, such as online encyclopaedias, eBooks, digital journalism, geography and history websites or teaching sites. At the same time, the information offered on these websites may be useful to detail and enhance the contents of GeaCron*".

Another interesting project related to historical and temporal information systems is the European *Time Machine Project* [36], coordinated by Frederic Kaplan, Professor of Digital Humanities at the École Polytechnique Fédérale de Lausanne (EPFL). The initiative is aimed at designing and implementing advanced Artificial Intelligence (AI) technologies to gather and give open access to a vast amount of complex historical datasets about Europe's cultural heritage (CH). The idea is to map in a digitalised form Europe's entire social, cultural, and geographical evolution. This is meant to transform fragmented data coming from different data sources (from medieval manuscripts and historical objects to smartphone and satellite images) into usable knowledge. A smaller version of this wide project is the *Venice Time Machine* [37], which comes from a collaboration between EPFL, the

National Archive of Venice and the Ca' Foscari University. This project deals with a digitalisation effort on the document series stored in Venice Archive to create a Big Data repository from the XVI Century to the contemporary age.

Besides public content of general interest, a map can be associated with *personal content* produced by individual users, such as annotations or pictures. In this case, the map can be used as a basic structure for news reports, photo features, or even a diary of an event, a trip, a project, or a life experience. Different websites and applications have been developed for this purpose. For instance, *My Travel Map* [38] allows to create a personalised map and to share it with friends; it is possible to select a country from a world map and specify if the user is born there, lived there or has simply visited that country. In this way, a coloured map can be visualised based on the selected countries. A more structured approach is the one of *MyTripMap* [39], a travel website where ready-to-use itineraries are shown. It exploits user-made travel paths created on Google's My Maps through markers, and it is aimed to put in contact Italian travel websites to create a complete touristic guide for travellers.

From the technical point of view, all these pieces of information can be codified and managed by *data structures* of different types, depending on the expected usage in the successive analysis and visualisation phases. Many times, a common relational database can be enough, or a database specialised in geographic data, while in some other cases, a geo-referenced database can be more suitable, up to a more complex ontology.

Recently, new types of GIS have been implemented to make the system flexible enough to collect *multimedia content* as well. In contrast to traditional GIS, *Multimedia GIS* (MMGIS) [40] is not only able to collect, analyse and store data in traditional formats, i.e., text, images (pictures) and graphs, but also audio (sound), animations and video (moving pictures). Conceptually there are two ways to view MM databases. One way is to include all media as an integral part of a single database (e.g., Oracle). Another way is to include all media as a sub-database, connected in such a way to achieve communication among them. MM database consists of the following sub-databases:

(1) a *Text Database* may contain text in the form of messages, comments and definitions etc., as required by the user need. Sometimes text in the form of oral presentation with the help of an audio database may be included. (2) An *Image Database* may contain all the images, maps, sketches, graphics, photos, etc. combined with the text, audio etc., where necessary. (3) An *Audio Database* may constitute all the sound signals, oral presentation (of text) and music. (4) A *Video Database* may contain animation and full-motion videos along with sound and music as per requirement. (5) The *Main Database* is the central database of the system that is capable of connecting all other sub-databases. Current examples of Multimedia GIS in commerce are *ArcGIS Insights* [41] and *ArcGIS StoryMaps* [42], [43] by ESRI.

In the following sections, some considerations about what illustrated so far are reported.

1.2.1 Models, structures, and formats of spatial data

When dealing with cartographic data in digital format, it is important to consider three main aspects of the models and structures to represent them [44]:

1. Database models – alphanumeric-tabular component of spatial data.
2. Data structures (vector/raster) – geometric component of spatial data.
3. Codification formats of spatial data.

Databases structures

A *database* is a time-persistent collection of non-redundant data that are correlated and organised to be easily retrieved, managed, and modified. A geographic database (*geodatabase*) of a GIS can be defined as an archive of territorial entities and their relations, structured in files managed by a Database Management System (DBMS).

A *data model* is needed to define how data collection is structured and it depends on the objectives of a system. Different data models can drive the organisation of data. The *relational* data model (RDBMS) is the most diffused in current applications used one, together with the *object-oriented* model (OODBMS) employed for more complex data structures. *Objects* are elements or concepts of the real world that can be uniquely identified and are characterised by their properties (attributes) and behaviours (methods). This type of data model allows one to define new attributes and behaviours of objects and hence is suitable for applications that require complex data structures (pictures, sounds and so on).

Data scientists are moving towards a combined structure called *object-oriented relational DBMS* (OR-DBMS) used for example for geo-referenced contents: object-based skeletons connect objects through relational relationships. An example is ArcGIS by ESRI.

Cartographic data structures in digital formats

To be stored and managed as a database, geographic elements must be defined through a *spatial data model*. A geographic database can be codified into two digital structures. The first is the previously mentioned *vector structure*, which is made of basic geometric elements such as points, lines, and polygons, and that is used to represent discrete objects (e.g., elements on a territory represented by polygons). The second is the *raster structure*, which is a grid of pixels that include some information, which is used to represent continuous phenomena (e.g., altitude of territory over the sea level).

Codification formats of spatial data

In a GIS environment, different formats can be used to manage large amounts of data. The most common ones are described below.

- *Shapefile*. The shapefile is a storage format of vectorial data developed by ESRI, that can record their localisation, form, and attributes. It is made of a set of connected files that include a homogeneous collection of objects (points, lines or polygons) and the attributes of each characteristic are stored in a table [45]. Three main files constitute a shapefile, with different extensions. **.shp* is the main file including the geometric features (shapes) of the object; **.dbf* includes the attributes related to the shapes of the main file in tabular format; **.shx* is a set of pointers that connect shapes and attributes between the two previous files. Among other auxiliary files, the **.prj* includes the reference system and positions. The shapefile is commonly used for small data sets.

- *Geodatabase*. The geodatabase is an open structure to store and manage GIS data (like geometries, tables, and images) within a DBMS. In a geodatabase, each object (feature) is memorised in a raw of a table together with its attributes and represents a spatial entity that needs to be managed by the GIS (e.g., a building, a river etc.). The primary advantage of spatial databases, over file-based data storage, is that they let a GIS build on the existing capabilities of relational database management systems (RDBMS). This includes support for SQL and the ability to generate complex geospatial queries [46]. Moreover, this model provides an easier compilation of attributes through domains and subtypes and can import data from other formats, including the shapefile.
- *CAD*. A *Computer-Aided Design* (CAD) file is a common way to create digital drawings and it is very widespread in many sectors, especially in the architectural and urban fields. Developers use different data formats for CAD files, either neutral such as STEP or QIF, or proprietary, such as DXF or DWG [47].

1.2.2 Conceptual modelling of geographic databases

Representing spatial information usually involves the acquisition, storage, and manipulation of geographic locations of real-world entities. These locations cannot be manipulated as simple attributes; they subsume a set of aspects that are needed to obtain the expected application's capabilities [5]. For instance, the choice of the reference system influences the way data will be interpreted, as well as the static or dynamic nature of the geographic object. This means that data must be structured and stored in a way that is suitable for the data type and the expected successive analyses. Hence, the importance of the conceptual modelling approaches in GIS environments.

According to [5], conceptual modelling in database design often makes use of the formal approach known as *Entity-Relationship (ER)* *modelling*. In brief, it classifies real-world objects into entities, together with their features (attributes), and connects the different entities by means of relationships (associations between phenomena). The study provides an example applied to a geographic environment showing a city containing land parcels as a principal spatial phenomenon (Figure 1).

An advanced version of the ER conceptual modelling is the so-called *Hierarchical Entity-Relationship Diagram* (HERD) [48], which creates a hierarchy of 'flat' ER models connected by external relations. The method applies packing operations that group entities and relationships into higher-level ER diagrams (called *structures*) according to certain criteria. In this way, different levels of aggregations are created so that analyses from different perspectives can be applied. An example in [48] refers to the representation of the schema about a university, before and after the packing operation (Figure 2).

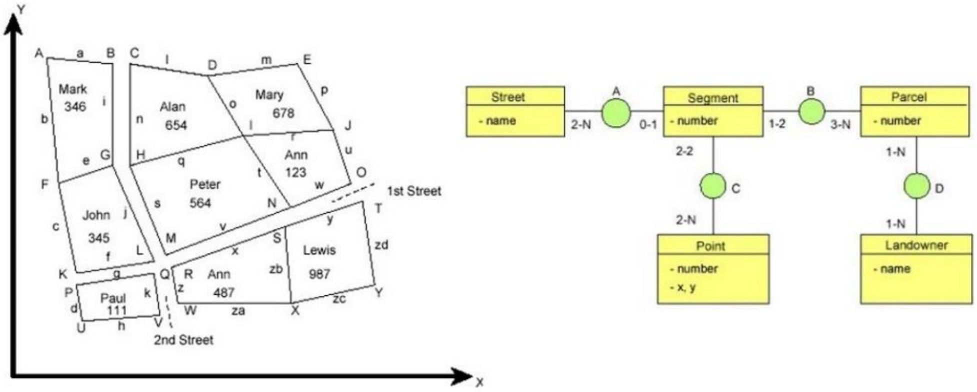


Figure 1 - Example Entity-Relationship Conceptual Modelling [5]

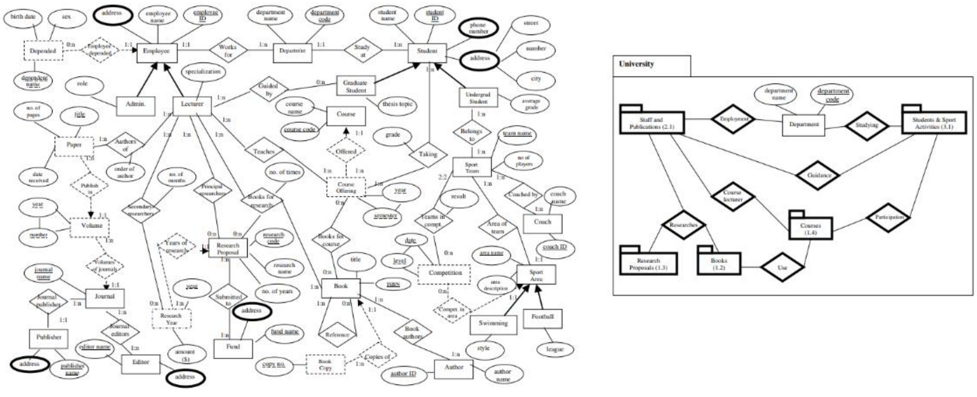


Figure 2 - Example of Hierarchical Entity Relationship Diagram (HERD): on the left, the original 'flat' ER diagram; on the right, the top-level HERD structure after the packing operations. [48]

More complex structures result in real *ontologies*, i.e., controlled vocabularies that dynamically describe objects and relationships between them in a formal way. In the cartographic field, some schools of thought believe that traditional data structures represent the environment under a reductionist approach, as they fail in recognising its dynamic and holistic nature. Some scholars [49] tried to fill this gap by analysing environmental complexities and shifting towards an ontological representation of both static and dynamic properties of a geographic scenario and prescribing spatial, temporal, semantic, interactive, and causal relationships among environmental elements.

1.3 Data Analysis

Data Analysis is aimed to query the database previously built to extract useful information and knowledge. It is about manipulating the collected data to climb the so-called *Data, Information, Knowledge, Wisdom (DIKW) Pyramid* (Figure 3). Each step answers different questions about the initial data and adds value to it. The more data are enriched with meaning and context, the more knowledge and insights are extracted out of it, which gives the chance to take better, conscious, and data-driven decisions [50].

This is the basis of *Business Intelligence* (BI), i.e., a set of mathematical models and analysis methodologies that exploit the available data to generate information and knowledge useful for complex decision-making processes [51].

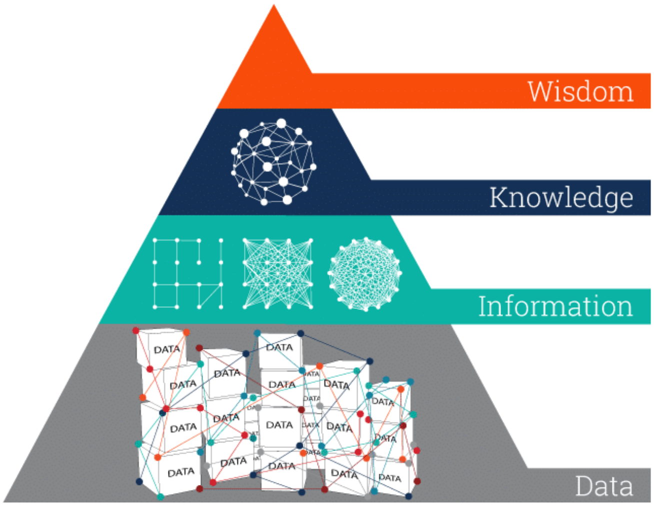


Figure 3 - Data, Information, Knowledge, Wisdom (DIKW) Pyramid. Each step of the pyramid answers questions about and adds value to the initial data. [50]

BI architecture includes three components:

1. The set of *data sources* providing raw data, that can be heterogeneous in the formats and must be integrated to be analysed as a single system.
2. The *data warehouse and data marts*, that use extraction, transformation, and loading (ETL) tools to store data in an aggregated form and run multidimensional cube analysis (not discussed in this dissertation).
3. The *BI methodologies* that use mathematical algorithms to extract knowledge from data and to take decisions.

The different levels of BI components are shown in Figure 4. *Data Exploration* refers to *passive BI analysis* based on statistical methods and reporting systems. *Data Mining* is a set of active BI methodologies such as pattern recognition, machine learning and data mining techniques. The *Optimisation* block refers to optimisation models that allow to evaluate alternative actions and determine the best solutions. The top of the pyramid contains the decision-making process.

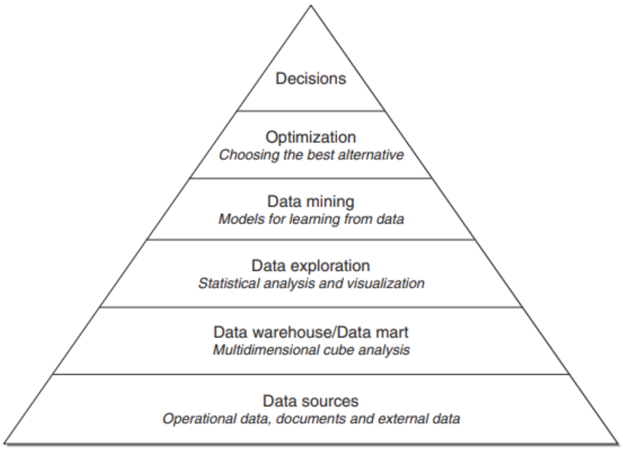

Decisions
Optimization
Choosing the best alternative
Data mining
Models for learning from data
Data exploration
Statistical analysis and visualization
Data warehouse/Data mart
Multidimensional cube analysis
Data sources
Operational data, documents and external data

Figure 4 - The main components of a Business Intelligence system. [51]

1.3.1 Data Mining and Machine Learning

Data mining (DM) is a process used by companies to turn raw data into useful information. By using software to look for patterns in large batches of data, businesses can learn more about their customers to develop more effective marketing strategies, increase sales and decrease costs [51]. DM depends on effective data collection, warehousing, and computer processing [52], and involves exploring and analysing large blocks of information to extract meaningful patterns and trends.

Machine Learning (ML) is the area of AI defined by A. Samuel as "*the field of study that gives computers the ability to learn without being explicitly programmed*" [53]. ML changes the paradigm of traditional programming: while traditionally the machine is provided with some input data and a program written by an operator and, according to a certain algorithm, produces an output, in ML the computer is given large sets of input data and the expected output, and the ML algorithm itself learns and provides the model to achieve that output.

The connection between DM and ML the two areas – data mining and machine learning – lies in that ML processes are used to build DM models that power applications including website recommendation programs.

The key point is the need for very large amounts of data to be analysed and taken as examples, to build an effective algorithm on them. One of the reasons why ML and DM have had such a big hype lately is the increased availability of data coming from very different sources in many different formats. We live in the era of the so-called *Big Data*, one of the deepest and most pervasive evolutions of the digital world [54]: enormous amounts of data, heterogeneous in source and format, that can be analysed real-time or in an aggregated form to extract trends and useful insights, to look for correlations or to classify associated information.

In this field of analysis, *spatial and geo-referenced data* can be considered a type of Big Data. Different modern practices are focused on map-based tools to run specific analyses, exactly exploiting Big Data, and making use of ML and DM algorithms. This makes the role of the map shift from an independent tool for simple geographic representation to a basis on which additional instruments for Big Data Analytics are built.

M. Kanevski *et al*. [55] is an introduction to ML models/algorithms and *their potential applications to geospatial data*, with a particular focus on artificial neural networks and statistical learning. In general, geospatial data are not only data in a geographical low dimensional

(2D, 3D) space but rather data embedded into high dimensional geo-feature spaces, which consist of geographical coordinates and features generated from, for example, digital elevation models, science-based models, remote sensing images, etc. In this sense, a relevant problem where ML and Statistical Learning have contributed is the management of the growing number of dimensions, so that dimensionality reduction methods (e.g., Principal Component Analysis) have gained great popularity. In terms of patterns/structures, the issues closely related to the main problems of learning from data can be identified as pattern recognition/detection (i.e., find/detect structured information in data without making restrictive hypotheses about data distributions, being able to detect and separate "useful" structured information from the noise), pattern modelling (i.e., correctly model structured information taking into account available data, expert knowledge, and science-based models), pattern predictions/completions (i.e., forecast/predict in space and in time the points where there are no measurements). Other traditional topics where ML has contributed are problems of optimization and control, calibration of science-based (meteorological, physical) and empirical (cellular automata, multi-agent systems) models, modelling/imitation of processes and events, etc.

ML applied to georeferenced data has also been used by scholars for social and demographic purposes. In a study by D. Feldmeyer *et al.* [56], some socio-economic indicators (identified with the names of *residents, unemployment, migration,* and *elderly*) were predicted based on OpenStreetMap (OSM) using three types of ML algorithms (random prediction as a baseline with linear regression; one ML algorithm; and one deep learning algorithm). The four metrics explain societal and economic conditions and are a common basis for many socio-economic indicators and of relevance for assessing and evaluating complex phenomena such as resilience, vulnerability, and sustainability. Firstly, the baseline was established by random prediction and a linear regression model. Secondly, random forest and deep neural networks (DNN) were applied as a ML and deep learning approach. Thirdly, these models were compared to the ground truth and evaluated for predictive power. The result of this study was twofold: on one side, it was meant to compare ML algorithms and model performances, getting to a sort of ranking of the models used. From this point of view, the evidence showed that linear regression was better than random prediction, random forest was better than linear regression, and DDN was better than random forest. On the other side, the main special predictors were highlighted for each of the four socio-economic metrics, with the evidence that, especially for the DNN model, the number of residents per municipality was best predicted with the lowest error, followed by migration, elderly, and unemployment. To make an example to better clarify this second objective, the most important features identified to predict the number of residents in a certain area were *Train system, Infrastructure, Shopping and culture*, and *Rurality*.

The use of such complex algorithms and techniques has for sure enabled recent advancements in computer vision and AI for spatial analysis. Nevertheless, the efforts needed to segment data and to code the algorithms to make them effective and efficient are a restriction for those interested in conducting spatial and remote sensing analyses without a software engineering background. To extend the potentiality of these tools to a less-skilled community, the Swiss company Picterra developed the namesake *Picterra Tool*, a geospatial cloud-based platform specially designed for training deep learning based detectors, quickly and securely [57]. The tool was defined by Mark Altaweel as "*A relatively easy to use interface that allows users to upload remote sensing images whereby users can identify and train an automated detector to find and detect objects of interest*" [58]. With this platform, which provides a ready-to-use deep learning model, the user can upload raster data, identify, and train a detector to find relevant objects of interest, identify objects for measuring accuracy to compare his results, and then apply the created model to his dataset.

In summary, the key elements of ML and DM models for our application, namely geographic maps, are their ability to run *predictive analyses*. The basic aim is to pinpoint on the map the locations where events have occurred (crimes, landslides, collapses, or the opening/closure of commercial activity) in order to be able to *forecast* future events via predictive algorithms.

A relevant topic consists of *predictive police* applications, aimed to identify (i.e., predict according to some parameters) the areas of a city where the risk of potential crimes is higher. These applications are meant to support the decision making of by municipalities and police forces so that those risky areas can be more carefully patrolled. One example is the *Decision Support System* (DSS) proposed by M. Camacho-Collados and F. Liberatore [59] in collaboration with the Spanish National Police Corps (SNPC). Since many police agencies have reduced resources, especially personnel, with a consequential increase in workload and deterioration in public safety, the use of the DSS is meant to help to optimise effective use of the scarce human resources available. The objective of the tool is to define partitions of a territory, analyse the crime records provided by the SNPC and determine patrol districts based on forecasted crime risk. The results of the experiments conducted in the Central District of Madrid showed that the proposed DSS was outperforming the patrolling area definitions currently in use by the SNPC.

Predictive analyses and correlations identification on georeferenced information can obey to various purposes, including environmental, urbanistic, econometrics, and even for risk prediction (such as environmental, seismic, hydrogeological, sanitary, social risks).

A specific tool to analyse georeferenced information is the use of *geometric* or *spatial queries*, where distances, areas, intersections of areas or optimal paths are evaluated. If long distances are considered, the distortion of Earth curvature needs to be taken into account as well. To this purpose, *GeoRaster* is a feature of *Oracle Database* allowing to store, index, query, process, analyse, and serve georeferenced raster image and gridded data and its associated metadata [60]. It provides native data types and an object-relational schema to store and manage multidimensional arrays, grid layers and digital images that can be referenced to positions on the Earth's surface or in a local coordinate system. What differentiates GeoRaster is the ability to perform raster analysis on extremely large images and data sets, provide in-place image processing and analysis with no development required, and provide parallelised image processing with a simple invocation of

PL/SQL procedures[1]. GeoRaster is used with data from any technology that captures or generates raster data and images, such as remote sensing, photogrammetry, and geospatial thematic mapping.

In this context, the concept of *spatial data mining* is introduced, as an analysis of geospatial data where spatial data are analysed to discover interesting and previously unknown, but potentially useful, patterns. However, explosive growth in the spatial and spatiotemporal data and the emergence of social media and location-sensing technologies emphasise the need for developing new and computationally efficient methods tailored for analysing big data. R. Vatsavai *et al.* [62] reviewed the major spatial DM techniques and algorithms, that, across the different computing methodologies, mainly focus on ML models, especially unsupervised ones, such as clustering algorithms. J. Han *et al.* [63] provided instead a spatial DM system prototype called *GeoMiner*, that worked with three main kinds of rules: *characteristic rules*, *comparison rules*, and *association rules*, in geospatial databases, with a planned extension to include mining *classification rules* and *clustering rules*. In GeoMiner, a spatial DM language called GMQL (*Geo-Mining Query Language*) was designed and implemented as an extension to *Spatial SQL*. Moreover, an interactive, user-friendly DM interface was constructed, and tools were implemented for the visualisation of discovered spatial knowledge.

1.3.2 Business Intelligence

Georeferenced input data are suitable to run classical BI analyses, that use mathematical algorithms to extract knowledge from data and take decisions in different contexts and business areas.

Among the latest implementations, A. Papandreou [64] investigated the planning and development of a mapping, georeferenced and analytical tool and its utility in the agricultural sector. Today, the technological advancements have supported the agricultural sector to optimise cultivations thanks to the so-called *Precision Agriculture*,

[1] PL/SQL is a procedural implementation (Procedural Language) of SQL language for the development of applications that use Oracle’s RDBMS. PL/SQL is a language structured into blocks that combines the ability of SQL language in manipulating data in the database and the ability of procedural languages to process application data. [61]

which is based on technologies and instruments that initially record the existing state of the parcel, then manage the data and eventually apply the inputs covering spatially and temporally the needs of each plot item according to its variation. The importance of using BI technologies in rural areas is related to the development through the information that this technology can provide to the user, including what cultivation exists in the parcel, its geographical limits, crop history, etc.

One of the most diffused applications of BI techniques on georeferenced data is the so-called *geomarketing*, a methodology where geographic maps are used to make analyses for marketing purposes. It is often employed to monitor the sales network of commercial activity and to analyse the geographic distribution of customers, of the demand and competitors. For example, choosing where to open a new outlet is a critical decision for retail firms. To this purpose, A. Baviera-Puig *et al.* [65] discussed a geomarketing model that could help managers to design supermarket location strategies according to shop features, competitors, and environment, whilst estimating supermarket sales. Geomarketing offers a way of carefully and methodically analysing the location of target consumers to achieve greater profitability. Geomarketing works because local market potential and purchasing power depend on demographic characteristics within a shop's trade area. The basic assumption is that people tend to congregate with others who are similar in terms of certain factors that may determine consumption, such as social status, household composition and ethnicity. Thus, geomarketing can be defined as the use of GIS to analyse data and make retail decisions, with the aim of meeting consumer needs and wants whilst making a profit.

Another thematic sphere where geographic analysis is employed by enterprises is meant to rationalise and optimise the *Supply Chain*. Research by S. Kang *et al.* [66] suggests a three-stage model framework that uses GIS to design a microalgae-based biofuel supply chain to meet the goal of economic commercialisation. Starting from the evidence that microalgal biofuel is considered as promising renewable energy for transportation, scholars have developed a supply chain model to minimise production costs. First, the design stage defines the spatial layouts and dimensions of each scale of biorefineries and runs an economic evaluation to estimate the investments and operating costs for different design options. Using the spatial dimensions determined in the first stage, the second stage selects the candidate locations for the biorefineries using a GIS-based site evaluation. In the third stage, a mathematical optimisation model is formulated to make multi-period strategic and tactical decisions of the supply chain under the total cost minimisation objective. This research demonstrated that significant cost reduction is possible by avoiding underutilisation of the capacity, which is possible thanks to more cost-efficient technologies and the integration of high-value co-products into the biorefinery portfolio.

As another example, a Portuguese study in the Brazilian Amazon [67] investigated the use of *geo-intelligence* techniques to support the evaluation of forestry areas in the Southern Amazons Mesoregion to identify illegal forestry activities and evidence of timber laundering. The research had a twofold objective: on one hand, evaluate how the use of geo-intelligence methodologies based on medium resolution imagery can be used on understanding the real situation of forest exploitation in areas authorized by the government in the Brazilian Amazon; on the other hand, verify if the forestry management activities are being carried out in compliance with the respective environmental standards, being truly sustainable, or whether laundering the timber illegally extracted from other areas.

1.3.3 Natural Language and Semantic Analysis

A quite innovative aspect is the application of *Natural Language* interrogations to maps, which bring practical benefits, among others, to GPS navigator devices and moving objects. A paper by M. Walter *et al.* [68] proposes an algorithm that enables robots to efficiently learn human-centric models of their environment from natural language descriptions. The novelty of the algorithm lies in fusing high-level knowledge, conveyed by speech, with metric information from the robot's low-level sensor streams. The idea is that a *semantic graph* provides a common framework in which concepts from natural language descriptions (e.g., labels and spatial relations given by an operator during a tour of the environment) and metric observations from low-level sensors are integrated.

Natural language processing requires some semantic representation of geographic objects. The instruments designated to *semantic analysis*, typically an *ontology* associated with an inferential engine, give the possibility to get some logical or probabilistic deductions starting from a geographic (or anyway georeferenced) knowledge base.

Sometimes, these semantic deductions are made by humans who voluntarily give their contribution to producing large crowdsourced geographic datasets. This is the aim of *Volunteered Geographic Information* (VGI), as introduced in section *1.1 Data Acquisition*. A. Ballatore *et al.* [69] agree that OpenStreetMap (OSM) is the leading VGI project, aiming at building an open-content world map through user contributions. The authors describe OSM semantic as a set of properties (called 'tags') defining geographic classes, whose usage is defined by project contributors on a dedicated Wiki website. The emerging problem is that, due to its simple and open semantic structure, the OSM approach often results in noisy and ambiguous data, limiting its usability for analysis in information retrieval, recommender systems, and DM. The authors propose to design an *OSM Semantic Network*, a mechanism for computing the semantic similarity of the OSM geographic classes, which could alleviate this semantic gap.

The exploitation of semantic meaning to facilitate content navigation is often assisted by the presence of an *ontology*, like in [70]. The support of an ontology can provide the benefit of specialisation in a given context, such as the CH domain [71].

At this point, the next step would be performing *semantic analysis* not only on pure geographic data, but also on data of historical or cultural type, to extract useful insights about the population, habits, evolution, and psychology of a certain area, and to facilitate the dissemination of cultural heritage. Research by E. Meyer *et al.* [72] presents the development of a *Virtual Research Environment* (VRE) dedicated to the exploitation of Cultural Heritage information in the archaeological field. The main issue highlighted by the authors is that GIS have proved their potentialities in the scope of preserving information in a digital form, but they are not always adapted to the management of features at the scale of a particular archaeological site. Thus, they propose a *Web Information System*, which has the objectives of completing digital archiving for archaeological datasets, allowing innovative data inquiry notably through clickable maps and 3D models, and providing attractive visualization and communication of the site information thanks to thematic and interactive interfaces.

1.4 Data Visualisation

The fourth set of activities to develop a geographic-based application is *Data Visualisation*. Its objective is to make the different types of data available to final users through the visualisation of the map together with the associated pieces of information and related contents.

In this paragraph, the aspects to consider when implementing a map-based user interface are reported, together with some examples of existing applications.

In general, to graphically visualise a map, developers use some tiles that are composed of a grid; the most used formats are TMS, XYZ and WMTS [73]. Even if the input data is vectorial, the basic tiles are often of raster type and are generated statically, then stored on a map server for different zoom levels and returned when requested by a client. Other dynamically generated elements are overlapped on these, such as optional layer information. For vector data, the most popular format is the shapefile format.

Data visualisation can be guided by interactive navigation of the map, or it can be the response to some specific queries made on the data previously analysed. The reason why data visualisation is so important, especially thanks to the latest technological progress that makes the contents of a digital map more interactive and appealing, is that the possibility to map and navigate geographic-based data allows a wider public to better understand them. [74] provides a set of examples of interactive maps that demonstrate the powerful connection between geographic data and BI.

Some GIS use the technique of displaying different kinds of *charts* superimposed on the base map. For instance, a map called *Manhattan Population Explorer* developed by Justin Fung [75] visualises a model of the dynamic population of Manhattan, block by block and hour by hour for a typical week in late Spring (Figure 5). The map shows vertical bars of different colours representing the estimated population flow at a certain point of the city at a certain moment of the day. The population estimates are the result of a combination of US Census data and a geographic dispersion of calculated net inflows and outflows from subway stations.

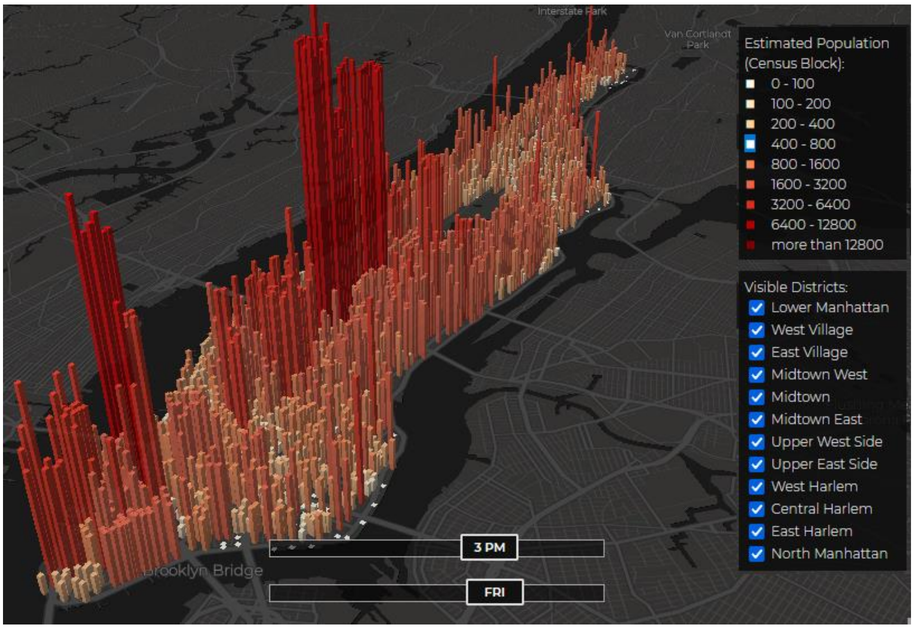


Figure 5 - Visualisation of Manhattan Population Explorer [75]

In general, the information provided by map visualisation systems is typically shown by means of texts, markers (i.e., graphic symbols that represent a specific geographic point), lines of different colours and thickness.

To make some examples, the classic Google Maps [76] shows the elements of the map such as restaurants, shops or other points of interest through markers of different colours and symbols for each typology.

NJ Hotels Near NYC [77] is a website authored by Jeff Howard developed to suggest hotels near public transit to Manhattan, which provides maps from a simple and basic design that use points, lines and polygons (Figure 6).

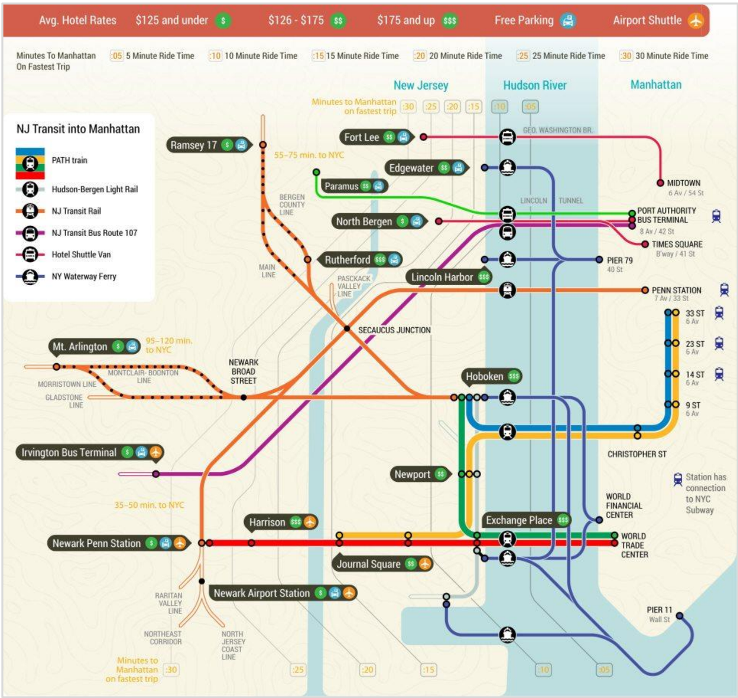


Figure 6 - NJ Hotels Near Transit to NYC [77]

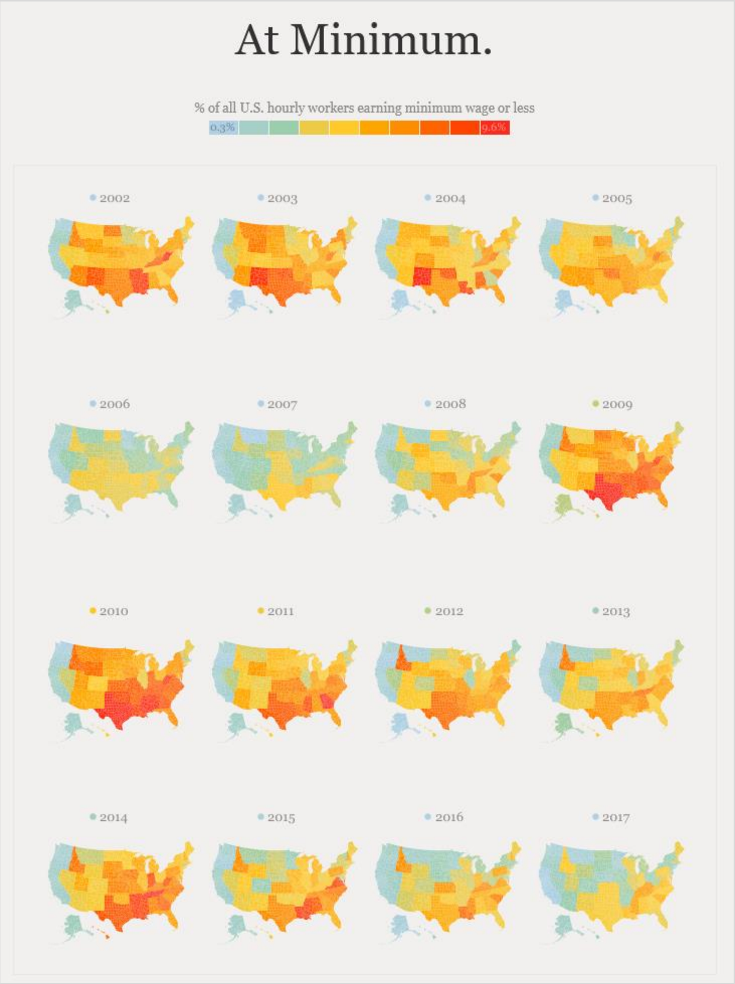


Figure 7 - At Minimum - percentage of all US hourly workers earning minimum wage or less [78]

Another important technique for visualising the data associated with a geographic map are the so-called *choropleth maps* which consists of colouring different geographic areas in different colours and shades [2]. For example, a collection published on *Tableau Public* by Justin Davis [78] show different choropleth maps (one per year, from 2002 to 2016) that visualise, by means of a coloured scale, the percentage of all US hourly workers earning minimum wage or less (Figure 7). Hovering the mouse pointer over a country highlights the percentage of workers in that area; looking year by year and State by State, it is possible to see if the trend is up or down.

In another implementation by S. Afzal *et al.* [79], a *typographic map*[2] is built by merging text and spatial data into a visual representation where text alone forms the graphical features (Figure 8). This type of design comes from the fact that labels generally play a supporting role in visual representations, but a radical new idea is to generate graphics where the textual labels alone form the visual features: in other words, the labels become the image. This is a form of calligraphy and is known as a *calligram*. In this case, the map uses the names of streets, highways, parks, and city blocks to form a geographical map of the city itself.

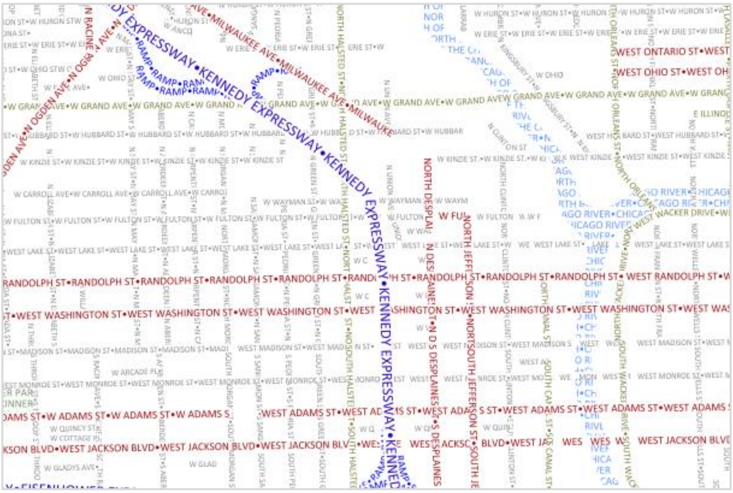


Figure 8 - Typographic map for Chicago, built using an automatic visualization technique and geographical data from OpenStreetMap [79]

[2] *Typographic maps* are spatial visualisations where the graphical features making up the visual representation consist only of text of different sizes, rotation, and graphical properties. Each text object is arranged so that it conveys not only the semantics of the spatial data (i.e., the label) but also its shape. Thus, the visualisation utilises spatial position effectively by placing the labels in the area they belong. [79]

The visualisation can be either in two or three dimensions, while the navigation typically occurs on a two-dimensional screen. In the latest years new systems that exploit Augmented or Virtual Reality (AR/VR) services have been implemented. For example, G. A. Lee *et al.* [80] developed a tool called *CityViewAR*, a mobile outdoor Augmented Reality (AR) application for providing AR information visualisation on a city scale. The CityViewAR application was developed to provide geographic information about the city of Christchurch, which was hit by several major earthquakes in 2010 and 2011. The application provides information about destroyed buildings and historical sites that were affected by the earthquakes (Figure 9). The geo-located content is provided in several formats including 2D map views, AR visualization of 3D models of buildings on-site, immersive panorama photographs, and list views. The aim is to exploit the AR technology to go back in time and see the city as it was, both before and right after the devastating earthquakes.

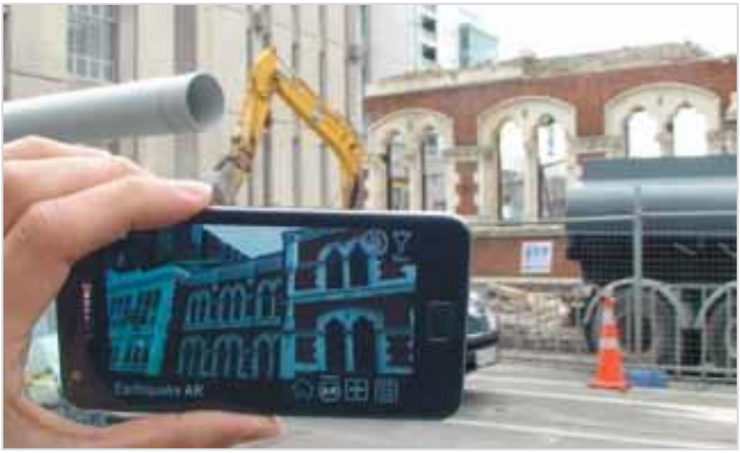


Figure 9 - CityViewAR showing virtual building on-site in AR view [80]

A wider overview is given by the effort of C. Parker and M. Tomitsch [81], who use the *Task by Data Type Taxonomy (TTT)* framework to analyse 9 AR apps (*Acrossair, Augmented Car Finder, Google Sky Map, iOnRoad, Layar, Lookator, Sunseeker, Wikitude,* and *Yelp Monocle*) according to some tasks: Overview, Zoom, Filter, Details-on-demand, Relate, History and Extract.

Consulting cartographic data could require the adoption of some selective data access criteria, typically based on *data privacy* and *data security* issues: here, the classic concepts of row and column security, typical of relational databases, can be translated into criteria to access map areas (row security) and map layers (column security), depending on the user profile.

Alternatively, instead of being constrained to a specific type of visualisation, the returned information can be made available through APIs to all visualisation environments that respect some characteristics. In other cases, it could be sufficient to take an existing cartographic system as the basis of visualisation and create additional *custom layers* including additional information.

The user interface is an aspect of GIS where further improvements are still needed. Some research has highlighted the need for more dynamic and adaptive map layers, as in [82]. Authors claim that visual representation alone is often insufficient to cope with the quantity and diversity of datasets brought together in application areas; *user interaction* can be seen as complementary to visual representation and helps alleviate this problem. Some geographic-based systems provide elaborate interactive data visualisations, but they are typically domain-specific, enabling interactions dedicated to the specific data managed by a particular application. In most applications on the market, once the map has been built, interactive navigation is simple and direct, but limited to basic interaction techniques such as pan & zoom, layer toggling or text search. These techniques consider the layers as flat images that can only be superimposed, juxtaposed, and sometimes drilled through. On the contrary, the creation and editing of more elaborate layer composites is cumbersome, involving many indirect manipulations.

2 The MAGIS Framework

2.1 Goals to achieve

The general aim of this dissertation is to propose an approach to the development of geographic applications that could extend existing GIS approach and that enriches geographic data with additional georeferenced material connected to the map. This model will be able to integrate a map with new cartographic data, both structured and unstructured, including texts, multimedia, links to external sources and the temporal dimension, and will be built as an open system, so to enable users to contribute to the enrichment and maintenance of contents.

In other words, the objective is to build an archive of additional content related to a map, that is adaptable to different contexts and applications, and which people can interact with, in a participatory way. The derived system covers the entire lifecycle of map-related contents, from data acquisition and storage to data analysis and presentation.

The result of this project can be therefore described according to the following keywords:

- *Modularity*: a logical structure divided into several software components or *modules*. Some modules are necessary for the application to work, while others offer additional features that can be implemented independently.
- *Adaptability*: easily reusable in different application contexts.
- *Multimedia content*: a media element, such as image, video, audio, text, document, or any other file format.
- *Geo-reference*: content items are associated with a specific geographic location on the map, that is the Point of Interest.

- *Time reference*: content items are assigned some temporal tags to create a timeline through which items can be filtered.
- *Thematic tagging*: content items are also assigned some semantic tags representing the *topic* of the item, to classify contents on the semantic level. Topic-related tags are linked one to the other through semantic relationships expressed by an ontology.

The management of georeferenced data is the basis of traditional GIS, while the use of multimedia instead of just structured data makes the proposed framework like a multimedia GIS (MMGIS). The difference with existing MMGIS is that the framework of this project is meant to be adaptable to different contexts and is built to automatically classify multimedia contents by means of a ML classification technique. For this reason, the name of the present work has been chosen to be "*MAGIS*", which stands for "*Multimedia Adaptive Geographic Information System*", to distinguish this system from traditional GIS and multimedia GIS.

Overall, the development of the present project is oriented to the achievement of the following goals:

1. *Enrich the map* with a set of information related to a specific thematic area (that is called *Context*), expressed by a set of multimedia contents. What is envisioned is a framework built into layers, where each layer corresponds to a different context. This is meant to facilitate the retrieval of data related to a certain topic or to make aggregate analyses based on the type of content. For example, a user that is looking for points of interest with a strong historic significance (e.g., monuments or historic buildings) is probably not interested in naturalistic or leisure locations. The separation of different topics into layers is therefore expected to be a value-added feature of the framework.
2. Allow users to *geographically navigate knowledge and visualise spatial and temporal relationships* between elements. The graphical interface of the application is based on a map, on which different contents will be shown by means of markers connected to points of interest. Based on the topic and time interval selected by the user and based on the zoom level, it will

be possible to see where the different elements are placed in space and in time, for example exploring what a certain area used to look like in a certain year in the past.

3. A third potential goal could be to respect the so-called FAIR principles: Findability, Accessibility, Interoperability, Reusability. They were published in 2016 in the "*FAIR Guiding Principles for scientific data management and stewardship*", a guideline that puts specific emphasis on enhancing the ability of machines to automatically find and use the data, in addition to supporting its reuse by individuals [83]. In brief, data and their metadata must be easy to find by both humans and computers, so that machine-readable metadata are essential for automatic discovery [84]; the access to data must be clearly defined, possibly including authentication and authorisation protocols; usually, the data need to be integrated with other data and need to be interoperable with different applications; finally, metadata and data should be well-described so that they can be replicated and/or combined in different settings to be reused when necessary.

Based on these main goals, MAGIS is meant to:

- *Mitigate the information overload*, by proposing some dynamic aggregation criteria to reduce the multitude of objects that can appear on the map. For example, by zooming out on the map, the points of interest that are very close one to the other will be aggregated in one single marker to facilitate the readability of the map information.
- *Facilitate the exploration of associated knowledge*, by proposing spatial and temporal correlations or thematic connections between objects. For instance, if a user frequently navigates the contents related to a certain topic, the framework is expected to suggest new points of interest related to that topic that the user may want to discover.
- *Import contents from existing archives* without requiring the manual reclassification of each object, by proposing the automatic classification of data. Once an ontology of tags is built to describe the content of the different contexts, a ML

classification algorithm is required to automatically analyse new contents uploaded (e.g., a text, a web article, a picture) and classify them according to those tags. This is meant to facilitate the import of collections of data coming from existing archival sources rather than manually classifying them one by one.

2.2 The MAGIS framework

The general framework proposed in this Thesis collects data from different data sources and organises them into a proper structure to be stored by means of an *ontology*. This framework allows the user to perform some analyses about the gathered data, e.g., DM explorations. The data model is suitable for different applications.

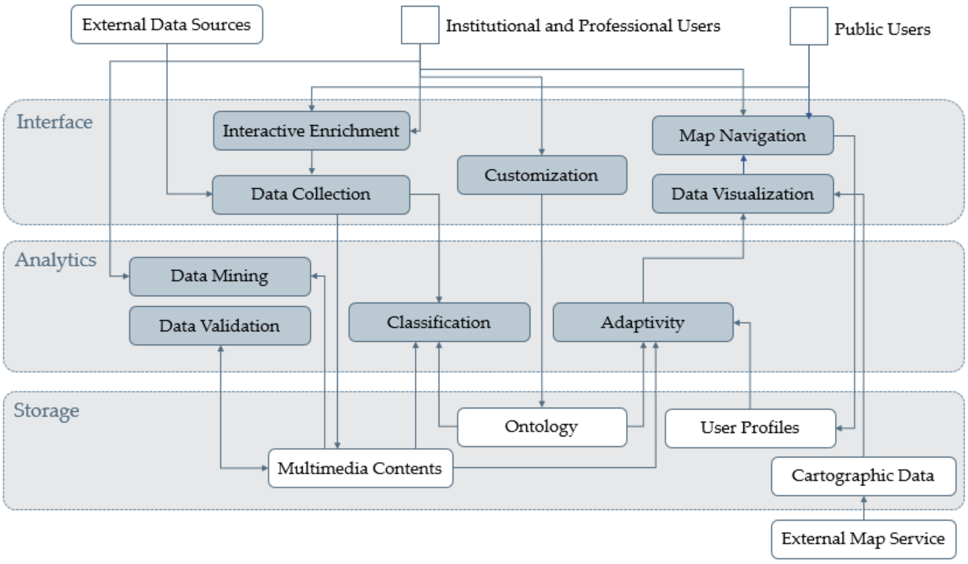


Figure 10 - Framework components and functional architecture

The components of the framework are described hereafter and can be visualised in Figure 10. Overall, the framework is built in one layer dedicated to the storage of geographic and context data; one layer dedicated to analytical modules that are in charge of running the related operations; an interface layer that manages the interaction with users and the data input/data output operations. The three layers are strongly interconnected thanks to meaningful interdependencies among the different modules. To make the picture clearer, white modules represent the different data structures, while light blue ones represent the algorithms involved. In the next paragraphs, the three layers are presented in detail.

2.2.1 Users

Two typologies of users are addressed d by this model.

On the one hand, *Institutional and Professional Users*, such as architects and urbanistic designers, university researchers or media professionals, may need a support tool to analyse the features of a geographic area, so to provide proactive services for urban planning. In this sense, the framework could also work as a learning tool. Among professional users, *Public Administrations* may want to promote improvements in their territory and may need a way to understand the characteristics and some risk issues related to that territory.

On the other hand, *Public Users* such as citizens and individuals can participate in enhancing the knowledge base of the system by enriching the map with interesting information related to an area they know competently, or they are visiting. Each category of users can authenticate to the platform and will be granted access to different contents and functionalities according to their needs and roles.

2.2.2 Storage

The bottom layer of the framework includes modules in charge of *Data Storage*. Data about the map are managed by the *Cartographic Data* module, which is based on a geographic database. This set of data will constitute the map layer, over which other layers will be built to represent information of different contexts or domains. Cartographic data may be also provided by an external map service; for instance, in the implementation example described in Chapter 4, the map layer is built from data collected from OpenStreetMap. In this case, the cartographic module is meant to fetch ready-to-use map tiles downloaded from the provider that need to be regularly updated.

All types of information related to the different contexts are stored in a dedicated database together with multimedia and metadata and are managed by the *Multimedia Contents* module. Multimedia items include a media element, possibly in the form of a hyperlink to external resources, together with a set of metadata describing the media. The presence of updated metadata is essential to classify the media and to propose them to the user in an ordered and effective way. Metadata should include information related to the provenance of the multimedia file, the typology of media, some temporal reference (e.g., when the media was generated, when it was published, whether it refers to a specific date or time interval etc.) and some tags or keywords summarising the content item to facilitate its classification. As the reader may understand, this is one of the key modules of the framework, as it stores not only the map-related multimedia but also is responsible for their temporal reference and semantic dimension.

Besides storing cartographic data and multimedia, a module that manages the *User Profiles* is placed in this layer: based on the profile assigned during the registration phase to the platform, users will be able to perform different types of operations and analyses on the uploaded data. Users' profiling is essential to make the framework adaptable to the specific preferences, as it is the basis for recommendation systems (as introduced in Chapter *1. Related Work*).

The last module represented in this area is called *Ontology* and it is meant to organise multimedia contents into a proper data structure to facilitate content classification and retrieval. This module includes a generic structure called *meta-ontology* that is very abstract and represents a sort of guideline to organise content items in every domain of interest; this generic architecture is made specific for each context thanks to a proper domain-specific second-level *ontology*. The difference between traditional content storage and this ontology is that here the different ontological elements are connected by a complex net of relationships of different types and nature so that a *semantic network* is created, and content retrieval and recommendation become more effective. In the vision of this project, the ontology is the heart of the framework as it joins the adaptive navigation features with the semantic organisation of contents.

As a final remark, when sensitive information is stored, a data protection mechanism is needed to protect all data from external and internal threats. Figure 10 does not represent a proper module for this purpose, as data protection is not included in the scope of this dissertation. However, it is important to consider that protecting data is not an option, but a responsibility and any implementation should follow the *security-by-design* principle. A data protection mechanism should take care of privacy and security issues of the different actors and resources involved, which is essential since the framework deals with open and shared data. Different entities sharing data and project "knowledge" might have diverse security and privacy policies. Security should be given by defining access control rules which restrict access to resources to selected teams of project members. Moreover, as sharing is decentralised (i.e., project participants can leverage existing documents from other project teams but can also create their associations and collaborations on the fly), authorisation should be dynamically provided allowing users to set the sharing policy for the content they provide without affecting the policies of other users [85].

2.2.3 Analytics

The *Analytics* area is populated with the key algorithms of the framework, that are meant to run some specific data analyses.

The *Classification* module implements the grouping of contents based on an architecture of thematic tags divided by topic (e.g., *Citizen Journalism, History, Nature, Climate, Demography, Economics, Architecture/Urbanistic, Cultural Heritage, etc.*.). These thematic domains allow specific analyses that involve precise elements of the map. This module has a twofold objective: on one hand, it is meant to analyse the collected content items and associate a proper set of tags to represent their subject; on the other hand, assign them to a proper class of the ontology according to the associated tags.

Adaptivity is another characteristic module, which uses the context-specific ontology to organise the presentation and navigation of contents on the map. The adaptivity of the framework depends on the context but also on the type of user profile. As will be explained in one of the next paragraphs, adaptivity is envisioned in a double nature: on one side, manual adaptivity is related to the deployment of a domain-specific ontology to represent the concepts of a specific context; on the other side, automatic adaptivity refers to the content classification at instance level, meaning that thanks to metadata, the algorithm will be able to infer which topic the content item refers to and which ontological element it must be assigned to.

Another module placed in this area of the framework is the *Data Validation* module, which is aimed at assessing the quality of new data acquired from external sources and contributors. This module could automatically validate the source of the data, for instance, if they come from an authoritative source like PAs, Census Authorities, or other certified sources. In case of data uploaded by citizens, this module could automatically validate new data by comparing their geo-location with the one where the citizen lives, or with the location of the citizen at the time the uploading takes place.

The *Data Mining* module allows qualified users to conduct complex analyses by combining structured data with multimedia content metadata. Examples of analyses will be geo/temporal data discovery (i.e., automatic discovery of areas that respect some pre-defined geographic or temporal characteristics) or geo/temporal data clustering (i.e., automatic clustering of map areas that present similarities in geographic or temporal characteristics).

2.2.4 Interface

The *Interface* modules are several. On the input side, the *Data Collection* module filters data from external sources and drives them into the storage area, adapting them into the destination format. On the output side, *Data Visualization* and *Map Navigation* reduce information overload, by presenting users an adaptive content organisation, driven by the context-specific ontology, and implemented by dynamic clustering and filtering.

Two different classes of users' profiles and data types are used by the interface to discriminate accesses to information. A *public* (open access) set of data is accessible to common users, who can navigate the map and enrich it with their multimedia contents. To this purpose, the *Interactive Enrichment* module allows individuals to add pieces of information related to areas they are acquainted with. In an advanced implementation of the framework, this module could also allow citizens to contribute via participatory tools such as b*logs and forums* to interact with communities and discuss problems and solutions (e.g., two conflicting elements inserted by different people and discuss which one is correct). Blogs and forums could also be used as a validation starting point in case an individual remotely uploads data about an area different from where he/she lives (for instance, an individual could upload some information about a POI in a certain area and an inhabitant of that area could confirm that information via forum).

A *private* set of data is dedicated to registered professional users who can customise their view over the map, including defining their own classification ontology when needed (*Customisation* module).

2.3 Key aspects of MAGIS

In this subsection, the different features of MAGIS are presented. The key aspects that differentiate the present work from existing geographic-based applications are summarised in Figure 11, where the different components of the framework are classified into storage, analytics, and interface-related features. The different blocks are explained in detail in the paragraphs below (for continuity with data tagging and classification, the ontology is presented in the section related to Analytics rather than in the Storage one).

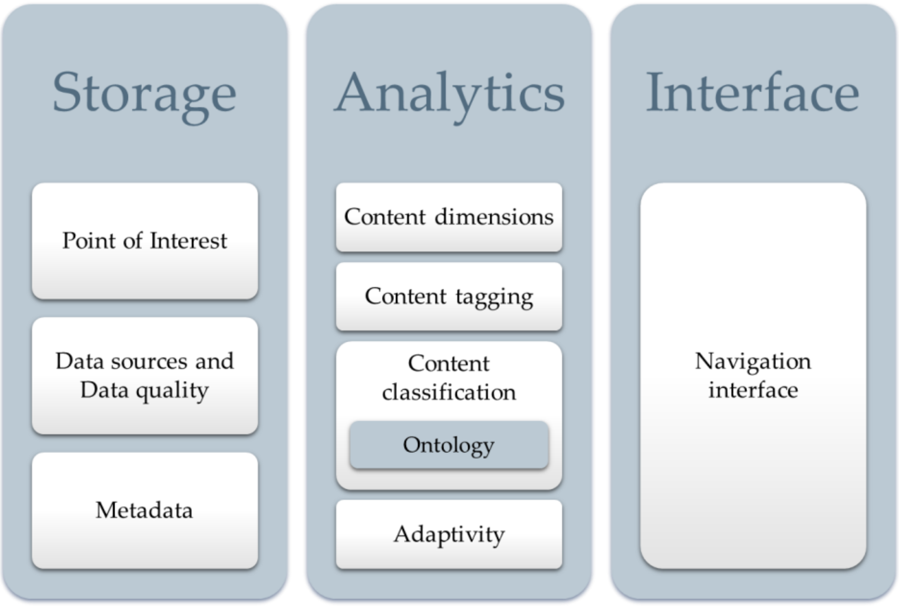


Figure 11 - Main features of the framework

2.3.1 Point of Interest (POI)

To connect geographic elements and associated information, the framework relies on the concept of *Point of Interest* (POI). A POI is a specific point location that someone may find useful or interesting [24]. In modern GPS navigation systems, the term usually refers to hotels, campsites, fuel stations or any other categories of elements associated with the map. A GPS point of interest specifies, at minimum, the latitude and longitude of the POI, assuming a certain map datum. A name or description for the POI is usually included, and other information such as altitude or a telephone number may also be attached.

In this case, the points of interest are those locations that are associated with some type of multimedia content, like a web article, a picture, a textual description. A point of interest can be a single element, like a building or a park, or an extended area, like a city; in the latter case, the content can be associated by a generic point within the area, such as the central point or the surface, or another representative location.

Analogously to existing geographic-based applications, a POI can be graphically represented by a "marker" visible on the map and which constitutes the access point to the associated content elements.

One point of interest can be associated with one or more pieces of content. For instance, the same building can be described by more than one picture, maybe to represent the evolution of that place over time. On the contrary, the same piece of information cannot represent more than one POI; if needed, the same content can be repeated on multiple locations, but the different copies will be treated as different contents by assigning different identification codes. For example, if a web article describes an event scheduled in two places, the user who uploads the article can decide whether to locate it in one or both destinations.

2.3.2 Data sources and data quality

A qualifying aspect of the MAGIS project is the integrated management of aspects related to the source or provenance of the contents.

This can be managed natively (i.e., *by design*) by incorporating it into the data model. In this case, the main issues are related to *privacy, licensing, reliability*. The goal is to recognise whether data come from a source that is authoritative, certified, voluntary, institutional etc. A module of the framework could be dedicated to this aspect and could work by comparing new data with a repository of certified sources according to different levels of reliability.

An alternative (or an additional solution) could be relying on a voluntary validation of data sources. For instance, the presence of a monument in a certain location could be verified by people who know about that area because they live there or because they have recently visited that site.

This second solution, in particular, opens up to another aspect, that is the concept of *data quality*. Data quality is essential when processing any type of data, especially in data analytics. Analysing data is aimed to extract meaningful and valuable information with the objective to support decision making and take effective and timely decisions. The quality of the results of any analytical process strictly depends on the quality of the input data, which is summarised by the so-called *Garbage in – Garbage out phenomenon (GIGO)*: by assuming that the processing model does not produce errors, the use of bad input data implies for sure a bad output. Especially when collecting data from different sources, their integration should be done on good-quality data only, so that a data preparation and cleaning phase must be run to purify data and solve potential conflicts and inconsistencies. In addition, small errors in data can become significant and cause big problems if not prevented (i.e., *snowball effect*).

Besides these considerations, if the framework is opened to voluntary contributors (even indirectly, by importing data from a *user-generated content* archive), a scan of the quality of entered data must be envisioned, mainly with two purposes:

1. Identify contents that refer to the same geographic object.
2. Identify errors and inconsistencies (either in the geolocation or in the metadata).

Again, a module of the framework could be dedicated to these aspects.

2.3.3 Metadata

To make aggregations and find correlations between contents, some metadata attached to the multimedia elements are required. The purpose is to collect a comprehensive set of metadata for each piece of media content to effectively classify the content and run analyses according to their characteristics. Short textual descriptions of the media are most likely to be used to perform the automatic extraction of tags that will be attached to the content.

In the project scenario, direct analysis of the multimedia content itself is excluded, as a specific analysis technique would be required for each type of media (audio, video, image, texts etc.). Nevertheless, in a subsequent extension of the system, these types of content analyses could also be combined with the analysis of the metadata.

The types of metadata taken into consideration are mainly the following ones:

1. *Geographic information*: the coordinates needed to geo-refer the object. They are needed to link the content to the correspondent point of interest so that it can be visualised on the map.
2. *Chronological information*: the location of the object in time, including a timestamp reporting when the media has been uploaded or created, and, if possible, the historic era it is related to, if the media is referred to a specific cultural/historic period.
3. *Descriptive information*: some textual information describing the object, possibly facilitating the retrieval. It also includes a set of tags that summarise the topics of the media content and that will be used for its classification.
4. *Provenance information*: information about the source of the media content.
5. *Technical information*: being multimedia objects, there may be some technical information on the format of the content, like the file size, image resolution, frame rate, etc.

The first three metadata types define the three dimensions in which contents are placed; the fourth one provides information on the origin of the media, while the last one is only of technical nature for any implementation needs.

2.3.4 Content dimensions

As the reader may have sensed, content items are placed in a space defined by three dimensions: geographic, temporal, and semantic.

- The *geographic dimension* is defined by the geographic element (whether point, area, or segment) to which the content is linked. It is defined through the geographic coordinates of the element (possibly including the elevation on the sea level and the extension in case of an area), and it can be navigated like any other map-based application.
- The *time dimension* is defined by the date of the event or by the period to which the content is linked. The date when the content has been produced can also be included, like the publication date of an article or the shooting date of a picture.
- The *semantic dimension* is defined by a series of properties (semantic tags) that indicate the belonging of the content to one or more categories of meaning (connected by a network of relationships, that is, an ontology). This is meant to classify the content and facilitate its navigation, analysis, or retrieval.

2.3.5 Content tagging and classification

In the vision of this project, once the content is uploaded on the platform, its metadata are employed to classify the media according to an ontology of thematic tags representing the correlations between contents.

Tags, i.e., simplified forms of semantic annotation, are metadata, which can be key-value pair or single value. Tags can guide the generation of layers, as they label contents based on their topic.

For more effective navigation of the contents, it is useful to also map the relationships among tags.

- The simplest implementation is a tag *taxonomy*, namely a hierarchy ("is-a", i.e., categories or classes and subclasses and their instances). Of course, several parallel and separate hierarchies can coexist. By making the thematic classification based on a hierarchical dimension, as the chronological and geographical dimensions are already natively, a multidimensional system similar to a data warehouse is obtained.

- The following step is a real *ontology*, that is to represent not only the hierarchical relationships but also other types of relationships between the tags (for example "is related to", "is part of", etc.). The ontology could also be enriched by relying on an existing generic semantic network such as WordNet.

For *manually entered contents*, a generic form could be envisaged in which the uploader directly assigns the tags he/she considers as valid for that object or adds others if necessary.

For *content imported from existing archives*, the tags assigned to each object should be inferred from its metadata. First, metadata must be extracted with a specific converter configured for the format in which the data is collected. This converter extracts, cleans and normalises the metadata. Then metadata must be mapped according to the specific tagging system. This is a part of the system where ML can play a role, through a proper classification algorithm.

One possible approach envisages extrapolating tags directly from metadata without human intervention, with an unsupervised ML algorithm (clustering) and automatically deducing the ontology from these.

In this project, an *expert-driven approach* is combined with an *ML-driven approach*. An application expert first designs the tag ontology and then manually classifies a certain set of content (*training set*), assigning one or more tags to each. From this training set, with a supervised learning algorithm, a model is generated to be applied to the contents imported from outside: based on the metadata, each object is classified and associated with an ontology node.

This hybrid approach is detailed in Chapter 3, where the content classification process is described more in depth and the development of the ontology is presented in its phases.

2.3.6 Ontology

The first phase of content classification deals with the development of an ontology of tags representing the contents shown on the map and their semantic correlations.

An ontology is defined as a "*formal specification of a conceptualisation of a shared knowledge domain*" [86]. It is a controlled vocabulary that describes objects and the relationships between them in a formal way. It has a grammar for using the terms to express something meaningful within a specified domain of interest and it is used to express queries and assertions.

Ontologies provide a formal specification allowing the use of a common vocabulary for automatic knowledge sharing. Formally specifying a conceptualisation means giving a unique meaning to the terms that define the knowledge about a given domain. To make it shared, an ontology captures knowledge that is common, thus over which there is a consensus; indeed, *ontological commitments* are agreements to use the vocabulary in a consistent way for knowledge sharing.

When classes and relationships are not that complex, the concept of ontology can be reduced to the ones of taxonomy and thesaurus [87]. A *taxonomy* is a hierarchical classification of concepts related to a knowledge domain; it defines the concept of inheritance between objects, that is the father-child relationship. *Thesauri* constitute a structure that allows the control of synonyms and homonyms; compared to taxonomies, they enrich the relationships between concepts including an equivalence relation between terms and associative relations (i.e., "*Is Related To*"). When these relations are not enough, *ontologies* are employed to describe "a portion of the world", which is a knowledge domain. The most relevant characteristic of ontologies in relation to thesauri is that they express all semantic relations as needed whereas thesauri provide a limited number of pre-defined semantic relations between concepts [88]. An ontology, therefore, describes how different concepts are combined in a data structure containing *all* the relevant entities and their relationships within a domain.

In MAGIS, an ontology structuring contents according to their topic is meant to help users selectively navigate materials. Since content items are related to different contexts or semantic domains, each domain is described by a proper hierarchical structure of tags, preliminarily built by an expert.

To provide the most generic description of the framework, this dissertation presents a generic ontology containing all the elements needed, including concepts that do not strictly concern the *content classification* problem. In practice, the ontology will also include classes about the geographic representation of the territory to make the framework as general as possible. The ontology will then be restricted to topics related to content management (information elements) to which the multimedia files, metadata, geo and temporal tags etc., are associated.

The ontology provides a *hierarchical classification* of multimedia based on their tags as metadata. A first-level classification is made according to the *context*: for instance, a media can be related to news context, cultural heritage (CH) context, historical context, architectural context etc. A second-level classification is instead meant to detail, within a certain context, the *topic* of the media: for instance, in the news context, a media can represent sports news, a cultural initiative, a political conference; within the historical context the media can be related to war (or, at a deeper level, a battle), a political event, etc.; in the CH context, a media can be referred to a political speech, a traditional recipe etc. The topics in which the contents are grouped are different for each context. The granularity of tags can be improved by furtherly defining sub-topics if needed.

The ontology also must be able to model the time dimension to visualise the evolution of territory over time.

In Chapter 3, the section 3.2 is dedicated to showing how the ontology has been built into practice in this project.

2.3.7 Adaptivity

One of the key features of this project compared to existing software solutions is its adaptivity characteristic. This system is meant to be able to automatically adapt to the context, that is built from the application domain and the users' behaviour. The adaptability to multiple domains facilitates the reuse of the system in different sectors. For instance, the definition of a general ontology of elements that are then specified according to the domain allows the reuse of the same data structure for many different contexts, enhancing the ability of the framework to adapt to the specific use case. As the specific content is loaded, the system associates the content to an ontology element thanks to its classification tags; moreover, the user can define new tags and new corresponding relations between labels, enriching the semantic interconnections between items and contributing to the dynamism of the system.

In parallel, the framework *adapts to the user's preferences* and navigation habits, thanks to recommendation systems that are already present in many application environments. The implication is that the user is driven towards the most browsed content, consistently to facilitate content retrieval and decreasing information overload.

In other words, the framework makes use of context-specific ontologies to organise the presentation and navigation of contents on the map. In this perspective, adaptivity has a twofold nature:

1. On the one hand, *manual adaptivity* is realised when a domain expert deploys a context-specific ontology to represent the concepts of a specific domain. Innumerable ontologies can be potentially designed according to specific needs and application requirements. Crowdsourcing can also contribute to the creation of new domains or the enhancement of existing ontologies; for instance, a professional may decide to create its own ontology and make it available to other users, either professionals or commoners.
2. On the other hand, *automatic adaptivity* refers to the content classification at instance level: thanks to metadata, the selected classification algorithm will be able to infer which topic the content item refers to and which ontological element it must be assigned to. In this way, it will be possible to massively import large quantities of data at a time, opening the use of the framework to plenty of applications.

In practical terms, the adaptability to the context can be conveyed in different ways, for example, by selecting different query filters or by dynamically generating a layer containing context-dependent elements. Alternatively, it could be possible to dynamically generate the point-of-interest and the clusters of multimedia content associated with them (not only the content of the clusters, but their structure should also adapt to the context).

2.3.8 Navigation interface

The interface for knowledge consultation mainly consists of a geographic map, supplemented by navigation elements that allow the user to move along the other two dimensions (time and semantics). Overall, the framework manages those three dimensions for visualisation through different instruments.

Layers tools

1. *Geographic dimension*: the geographic position of the element on the map is signalled though markers.
2. *Time dimension*: the user is expected to be able to change the time interval using a selector for periods/dates.
3. *Semantic dimension*: the different contexts are expected to be shown though layers containing semantic tag groups (content categories). The user can select the layers of the different contexts that are most suitable for his/her analysis.

Zoom tools

1. *Geographic zoom*: classic zoom along the spatial dimension, natively provided by all cartographic systems. The more the user zooms the map down, the more POIs are shown; the wider the area selected, the more the content is aggregated in a limited number of markers to ease the visualisation.
2. *Temporal zoom*: it is used when events can be grouped into periods of variable length. The user can analyse the POIs and the multimedia on the map based on a certain time interval that can be more aggregated or less aggregated.

3. *Semantic zoom*: grouping larger or more detailed categories based both on how the user navigates and also on the subdivision of contents (if one looks at smaller areas, more details are shown; a larger area is selected, an aggregation of contents is displayed instead). In other words, based on the map zoom, the contents themselves can be differently aggregated.

Search Tool

The navigation can also be in the form of a *search*, i.e., direct query to search for set of content that is not mapped by ML classification. An algorithm for keyword extraction can be applied to optimise the search by previously extracting the most relevant terms related to the content items.

Visualisation adaptivity

Besides these tools, visualisation adaptivity remains an option. The user can customise the navigation either by modifying the layers proposed by the system, or by adding his/her tags or comments to the objects.

These customisations could be stored in the system to re-propose custom layers to the same user later. This is also facilitated if the user is profiled.

If many users modify a layer proposed by the system, it could be updated to suit the use made by the users. This aspect is similar to what some recommendation systems do: considering other users who have a similar profile or who have made this same query, what other queries did they make? What other content did they open?

In this scenario, one aspect that could be difficult to quantify is the relevance of one object compared to another (also because it varies according to the type of user). On this, a ranking could be made based on the number of visits by users to that content.

Content aggregation

While browsing the map, it is important to avoid showing *all* the points-of-interest at the same time to prevent the risk for information overload. It is important instead to aggregate the contents to reduce the amount of information presented to the user. Different types of aggregations are possible; some of them are related to the three main dimensions (geographic, time, semantic) previously described.

One type of aggregation that can be performed is *thematic aggregation*, allowing the user to consult a subset of points of interest at a time. This can be achieved by exploiting the "layers" containing a certain category of objects (i.e., contexts). In practice, specific filters are displayed based on underlying queries: for example, a layer can be built grouping only restaurants and pizzerias, another one can include only monuments and historic buildings etc. With this purpose, contents must be classified by associating each object with tags, on which the layers are dynamically built.

Another possible type of aggregation is *spatial aggregation*, which means to cluster POIs that are very close to each other and generate a single marker that groups them (a usual solution in some cartographic systems). Spatial aggregation is dynamic as it adjusts itself to the geographic zoom level the user is visualising.

Finally, a *temporal aggregation* can be carried out for those contents that have a chronological reference as well. The chronological classification is a little different from the other ones, therefore in the navigation phase, it could be treated as a separate dimension.

Besides working on those dimensions, other types of aggregations are possible. For instance, another solution could be to recognise when more contents are related to the same point of interest, e.g., two photos of the same building. As anticipated in the Data Quality paragraph, aggregating them implies identifying errors and inconsistencies among the different versions and solving conflicts to get to a unified POI description. Moreover, if the classification of objects is made using quantitative properties (such as the population density or the precipitation level), those properties are usually made discrete by converting them into layers (low, medium, high), which can be considered an alternative way to classify and aggregate contents.

In summary, the information overload problem could be faced with aggregations (clusters) and stratifications (layers) of objects, to be used in the navigation interface. The generation of these aggregations could be dynamic to adapt to the contents present (and possibly to the individual user).

3 MAGIS Approach: Content Classification and Ontology

This Chapter presents the *hybrid approach* to content classification (i.e., the combination of human intervention and automatic classification via ML algorithm) and how the meta-ontology and tag ontology have been devised for this project.

3.1 Content classification

In the vision of this project, the content classification is carried out following a hybrid approach: *Expert driven* + *ML driven*.

1. In the first step, a person who is an expert about a domain context builds the context specific ontology that will drive the classification.
2. In a second step, a machine learning algorithm automatically classifies the uploaded content based on the previously built structure. If media are supported by an adequate number of semantic tags (manually entered by the user), the classification is easier. The complexity lies in the fact that, if tags are not directly provided, the framework requires to use a ML algorithm to classify contents based on their metadata (like short textual descriptions or user comments). In this case, a supervised text classifier is required to parse textual descriptions associated with the media and extract keywords that can be assimilated to the tags defined in the ontology.

In addition to this, persists the possibility for professional users to define their own classification tree, i.e., to adapt the structure of the ontology: they can upload contents and define their own context ontology and tags and then re-run the ML process to reclassify the content on this. For instance, the single user can create new topics in a certain context range that better suit his/her contents and those topics enrich the ontology becoming part of a wider structure that can be exploited in subsequent processes. This allows users to make classification adapt to their needs, thus increasing the adaptability of the system and helping to do content classification analysis. This is the basic task of the algorithm presented here.

As a further step of the classification, the contexts can be sub-classified into topics of different hierarchical levels. Each content item is assigned to a certain topic pertaining to a given context. The basic idea is that a domain expert builds the ontology in a way that a ML algorithm can classify uploaded content according to the right contexts and topics. The classification algorithm, based on ML, will assign contents to the contexts, topics and sub-topics, that are part of the same ontology.

To sum up, the classification ontology is preliminarily built by a domain expert and can be customized by users, then the uploaded content is classified with the ML. In the following sections, the choice of the classification tool and the process to build the ontology are presented.

3.1.1 Classification tool

Once the ontology of tags has been built, multimedia contents must be uploaded on the platform and pass through the automatic ML classification.

The objective of this phase is to classify input data (i.e., multimedia content) with a ML algorithm starting from short textual descriptions. Examples of available descriptions can be a title, a caption of a picture, the text of a Facebook post or a headline taken from a blog or an online newspaper. The purpose is to extract some keywords that can be matched with the tags of MAGIS ontology, in such a way as to classify contents according to the correct node. This classification is meant to facilitate the analysis of the content once it is uploaded on the application, allowing the user to retrieve media related to a certain topic, do some filtering or cluster contents based on their semantic characteristics.

The types of algorithms that answer these needs are related to text analysis techniques, which can be divided into text classification and text extraction practices [89].

Text classification is the process of assigning predefined tags or categories to unstructured text. It is considered one of the most useful natural language processing techniques because it is very versatile and can organise, structure, and categorise pretty much any form of text to deliver meaningful data and solve problems. *Natural Language Processing* (NLP) is a ML technique that allows computers to break down and understand text much as a human would. Within text analysis algorithms, *Text Classification* (or *Topic Analysis*) is an NLP technique that knowing a set of existing topics in advance, can extract the most significant ones that are related to the analysed text. In a more unsupervised way, it automatically extracts meaning from text by identifying recurrent themes or topics. In this project, the topic analysis could be used to automatically organise text by subject or theme. Moreover, after training performed under the supervision of a domain expert, the algorithm can be used to automatically classify new contents according to the current ontology of tags.

Text extraction, instead, extracts pieces of data that already exist within any given text. The typical aspects that can be extracted from a generic text can be keywords, prices, company names, and product specifications from news reports, product reviews, and more. The two main applications are *keyword extraction*, which selects the most used and most relevant terms within a text (i.e., words and phrases that summarise the contents of text) and *entity recognition*, where a Named Entity Recognition (NER) extractor based on Neural Networks finds entities such as people, companies, or locations existing within text data and provides the corresponding labels. In this project development, a keyword extractor could be used to extract some tags from the text (e.g., index data and generate tags that can become part of the ontology), while an entity recognition technique could be used to recognise points of interest within the description of a media.

What is required for this project is a tool that is quite simple to apply and that can work well with texts in Italian language. The input data are expected to be the title and a short description of the media that have been gathered and stored with their metadata. The tool is required to return as output the tags that have been defined in the ontology and that represent the content. To do so, the best possibility is that the tool allows performing a training phase with a subset of the collected texts, previously labelled with the tags that are expected as output. Possibly, ready-to-use, pre-trained algorithms available online for free can be used as a starting point for the text classification step.

For automatic classification, the most suitable solution that has been identified is a *Short Text Classification* algorithm based on supervised ML. The choice for algorithms specialised in short texts rather than longer text analysis techniques derives from the nature of texts to be analysed in this project. As said, available texts could be a caption of a picture, the text of a post on social media or the title of an article on an online newspaper, which are far shorter than a paper or a manuscript. Compared with paragraphs or documents, short texts are more ambiguous since they do have not enough contextual information, which poses a great challenge for short text classification [90]. Several ML techniques can perform short text classification [91] [92].

Among the solutions available online, *Short Text Classifier* developed by *MonkeyLearn* [93] has been identified as a suitable candidate for this analysis. The reasons for this choice stand in the following benefits:

a. It is specialised in short texts, which applies well to the input data of this project, namely the title and the brief description of the media. Differently from a long message, the difficulty of a sparse text with a low number of features is that it does not provide enough word co-occurrence, which makes the data pre-processing necessary to understand the meaning.
b. Besides a predefined bag of words, it allows the user to define his own set of tags or categories to use in the classifier. This means that a fully-personalised training set of data can be built to prove the adequacy of the classification model, without relying on precompiled dictionaries only.
c. *MonkeyLearn*'s website offers an Academic plan for free for students and researchers, that can be exploited for the academic experimentation of this dissertation.

3.2 The Ontology

As previously introduced, the intention of this dissertation is to provide the most general and complete description of MAGIS framework. For this reason, a very wide ontological structure has been built, which includes classes related to both content elements and the geographic representation of the territory. In a successive phase, a practical prototype will exploit a ready-to-use map environment, so that it will not be necessary to store the geographic elements in this structure and the ontology will only include additional contents.

To remind the role played by the ontology in this project, the proposal follows a hybrid approach between an expert-driven and a supervised ML-driven classification. An expert defines an ontology that includes context-specific semantic categories, and then manually classifies a training set of contents. The contents collected afterwards are automatically classified using the ML model, associating each content to an ontology node.

The definition of the ontology as the basis of the data structure addresses the phase of *Data Modelling* introduced in the Related Work. To this purpose, the basic elements of an ontology for cartographic knowledge are presented (geographical objects, georeferenced data, and relationships with each other), allowing semantic analysis, navigation, and dynamic maintenance of the content associated with the map.

A proposal of a very general ontology at conceptual level was first defined. Besides, an attempt to develop MAGIS ontology with a professional specialised tool was made: the conceptual schema was implemented using Protégé (version 5.5.0), with the guidance of M. Horridge’s handbook [94].

Before presenting the specific data structure, it might be useful to provide an overview of how an ontology can be used in geographic-based applications. Afterwards, the creation of MAGIS ontology is presented step by step, clarifying notations and conventions.

3.2.1 Ontology-Based Data Access

Having clear what an ontology is, one of its latest applications is the so-called *Ontology-Based Data Access* (OBDA) [95]. OBDA is a new paradigm, based on the use of knowledge representation and reasoning techniques, for the management of resources (data, meta-data, services, processes, etc.) of modern information systems [96]. The key principle is to store data in their repositories according to a three-level architecture, made of data sources, an ontology as a formal description of the domain of interest, and a mapping between the two. Data are disseminated into distributed data sources, and they are retrieved thanks to a proper dictionary that allows users to directly query the data. In fact, the user interrogates the ontology through a familiar language, that is translated into machine-readable language by the mapping layer and transmitted to the underlying databases.

Thus, the use of ontologies is convenient when large sources of domain-specific data must be managed and when what matters is not just the data themselves, but also their relationships and interconnections. In the case of multimedia objects, as in this project, employing ontologies means facilitating the user to access materials and contents in a personalised way according to their specific preferences.

3.2.2 Personalised multimedia access

Personalised multimedia access aims at enhancing the retrieval process by complementing explicit user requests with implicit user preferences [97]. When the content is well-defined according to a proper structure of topics and domains, an ontology-based approach for content filtering can be deployed. A topic ontology is meant to support users in selectively navigating contents, by providing an enhanced representation of the relevant knowledge about the user, the context, and the domain of discourse, as a means to enable improvements in the retrieval process and the performance of adaptive capabilities. ML algorithms can automatically learn explicit semantics indicating which elements appear to be significant for users based on the way contents are explored.

An ontology-based representation is richer, more precise, and less ambiguous than a keyword-based model. An ontology provides further formal computer-processable meaning on the concepts. Moreover, an ontology-rooted vocabulary can be agreed and shared (or mapped) among different systems, or different modules of the same system, and therefore user preferences, represented this way, can be more easily shared by different players [97]. The key for personalised content retrieval is the definition of a *Personal Relevance Measure* (PRM) of a content object for a certain user, which allows to discriminate, prioritise, filter and rank contents in a very peculiar way.

Different approaches have been described in literature. In some cases [98], domain ontologies define semantic concepts related to user preferences and item features to build a hybrid recommendation model. The hybrid nature derives from the fact that concepts, items, and user spaces are clustered in a coordinated way to find similarities among individuals at multiple semantic layers, and such layers correspond to implicit *Communities of Interest* that enable enhanced recommendations.

The coexistence of ontological structures and communities of peer users, which improves the so-called *collaborative recommendation*, is pervasive in literature. Collaborative recommendation is effective at representing a user's overall interests and tastes and finding peer users that can provide good recommendations. However, it remains a challenge to make collaborative recommendations sensitive to a user's specific context and to the changing shape of user interests over time [99]. That is why scholars have been striving to employ and incrementally update *ontological user profiles* which are expected to offer improved coverage, diversity, personalisation, and cold-start performance while at the same time enhancing recommendation accuracy. To make recommendation effective, three aspects must be addressed: short-term user activity, representing immediate user interests; long-term user profiles, representing established preferences; and existing ontologies that provide an explicit representation of the domain of interest. With this knowledge, a system can leverage a variety of sources of evidence to provide the best-personalised experience for the user, combining the semantic evidence associated with the user's interaction, and social knowledge derived collaboratively from peer users.

3.2.3 Tag ontology and meta-ontology

In this project, the elements (i.e., classes) of the ontology represent the domain-specific tags used to classify multimedia contents, which are interconnected by sets of relationships.

On top of this, the additional concept of *meta-ontology* is introduced. Meta-ontology is a recently-coined term that derives from the studies of Willard Van Orman Quine and has a purely philosophical nature. The term refers to the role of ontologies in the investigation of the world and the effort they hide. According to Quine, ontology aims to determine what exists and what does not, while meta-ontology should clarify what ontology investigates and how to interpret ontological axioms [100]. In other words, ontology investigates what is there in the world, while meta-ontology focuses on what people ask themselves when they investigate what is there in the world [101].

Nevertheless, the philosophical definition of meta-ontology falls outside the scope of this project. For the sake of the present dissertation, the concept of meta-ontology is simply used to represent a *general and abstract structure* of elements outlined upfront and subsequently detailed and made concrete in the real ontology defined by the expert and the data objects that represent the ontology instances.

In other words, the proposal is to build a two-level ontological structure:

1. The first ontological level is very synthetic, and it is what is called *meta-ontology*. It corresponds to a very generic data model that is meant to map the key elements of the framework, and it is adaptable to (almost) every application domain.
2. When the framework is instanced in one specific application domain (e.g., in the case of this prototype, Citizen Journalism), a domain expert is entitled to concretise that domain into a second-level *ontology*; this represents an *instance* of the meta-ontology and it is defined by the specific categories of the domain (i.e., the topics and subtopics that are groups of content items and their relationships). This is the level where the single

content items are stored, together with their relationships with their topics. This is the most arduous level because content items can come in large volumes and can refer to multiple classes in a wide semantic network. With this purpose, an automated content classification through ML algorithms can be very helpful in associating each content item to the right ontological class, drastically reducing non-value-added activities performed by the user.

In this project, multimedia elements are classified based on their Topic (e.g., Culture) and Subtopic (e.g., Museum, Concert…); within the latter class, each element is one instance of the Subtopic and is associated with one point of interest. Again, a *two-level ontology* is defined: a meta-ontology represents the generic concepts of Topic and Subtopic and their *semantic relations*, which are then exploded into a domain-specific ontology and its instances by the single pieces of content (Figure 12).

This simple representation is purely hierarchical, which makes the classification a taxonomy, rather than a real ontology.

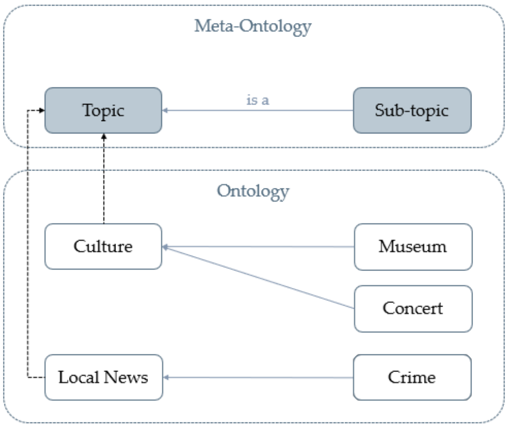


Figure 12 - Example of meta-ontology – Topic, Subtopic, Content, Location

A similar structure is obtained if, besides semantic relations describing connections between concepts, *linguistic relations* are added as well, to define the relationship between concepts and language expressions (Figure 13). This structure can have a dual function: on one hand, it becomes a sort of translator that declines linguistic expressions into different languages, creating a language-independent topic navigation; on the other hand, it helps identify and manage synonyms (i.e., when two linguistic expressions of the same topic belong to the same language). Overall, an "*expressed by*" relationship (*use-for* as a relation between terms [88]) is added to the simple *"is a"* hierarchical relation (i.e., superclass-subclass), converting the taxonomy into a thesaurus.

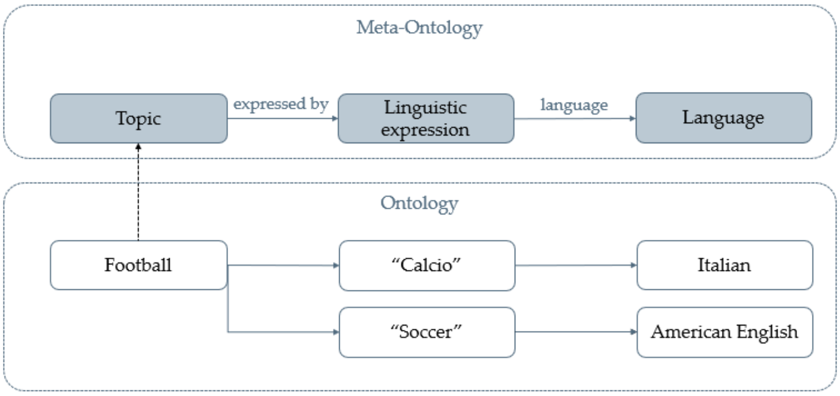


Figure 13 - Example of meta-ontology – Linguistic expressions

A more complex structure can be realised by increasing the number of relationships between objects. For instance, topics can be associated with some properties that can help the selective navigation of multimedia contents, e.g., music genre or the interpreter can specify the characteristics of a concert (Figure 14).

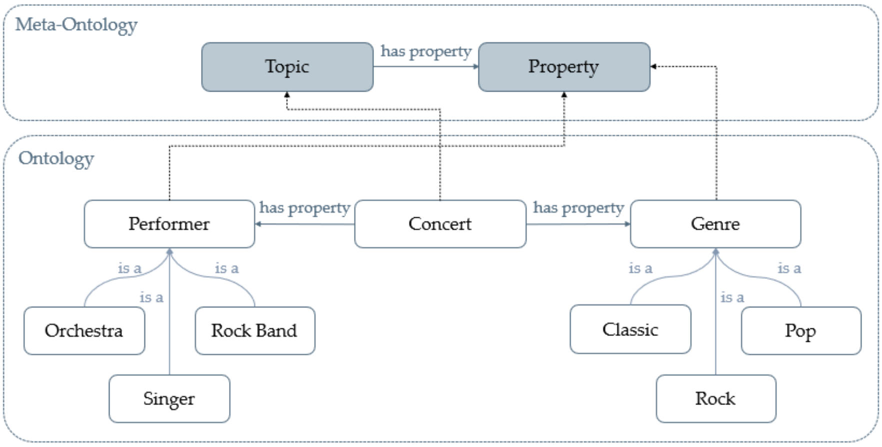


Figure 14 - Example of meta-ontology - Topic, Property

The difficulty here is how to derive those properties from the analysis of contents and their metadata. One possibility is to make use of Entity Resolution ML algorithms. Of course, the reader should recall that the approach proposed in this project is a hybrid method combining human-based and ML-based activity: when content properties are introduced, if an algorithm for automatic assignment is not provided, those properties would only be available for the manually entered content-items, or users themselves would be required to complete the missing associations, compatibly with a crowdsourcing or community-based interactive activity.

To enrich the schema even more, a further relationship "*related to*" can be added, so that the connections among classes make the schema closer to a real ontology. Figure 15 shows an example related to the *News* domain and some macro-topics (e.g., Event) that could be used to aggregate contents combining different hierarchical levels; for instance, a user may be interested in the news related to sports events but not in other sports news not related to events or in events of other topics.

At this point, four typologies of relations are available: "*Is a*", "*Related to*", "*Has property*", "*Expressed by*". Below are some examples related to other domains.

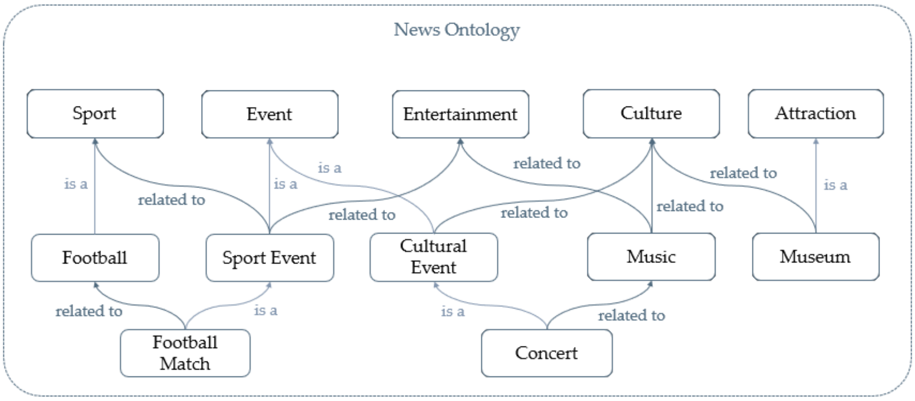


Figure 15 - Example of ontology: News

Figure 16 represents an example of ontology applied to the medical domain. It could be exploited by regular doctors who visit elderly patients at home to register visits and medical documentation. The map tool can support professionals by showing where patients live and how they are distributed on a territory, converting their houses into points of interest, while the doctor can store as multimedia digital documents such as prescriptions or medical reports related to the correspondent patients. Of course, this can be an example of private access to data, as medical information is sensitive and must be kept confidential.

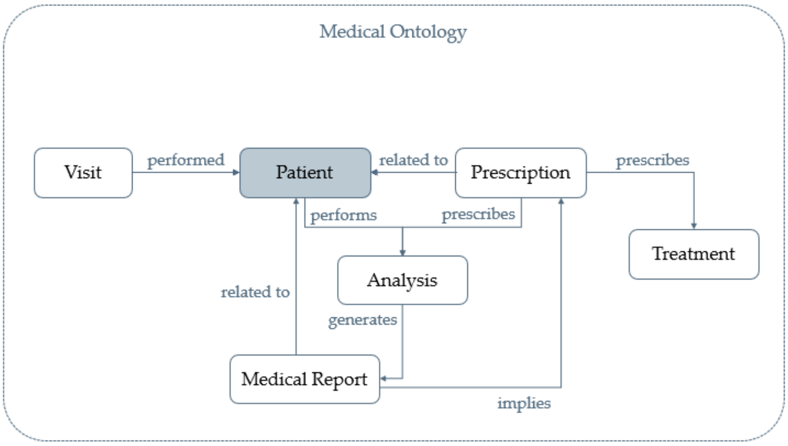


Figure 16 - Example of ontology: Medicine

Figure 17 shows an example applied to the commercial domain. A multinational company may use the map-based application to monitor some important features of its branches distributed across a territory (i.e., the points of interest). For instance, for each branch, it may be interested in tracing some documents about inventories or the catalogues of products published to customers, or storing the commercials used to promote some specific products in a certain timeframe, or even some statistical reports about the economic or financial performance of the branch over time.

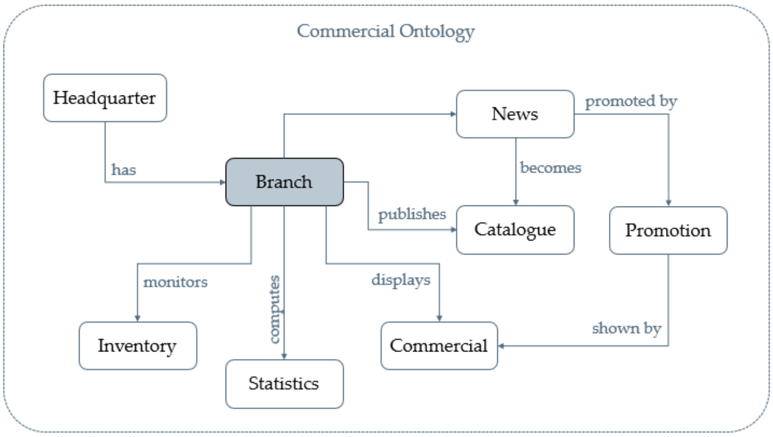


Figure 17 - Example of ontology: Commercial

In these examples, all the three dimensions of the framework – geographic, temporal, and semantic – are involved. The house where the patient lives and the location of the branch are points of interest displayed on the map. The different classes represent the relevant data that the user may be interested to store, which are likely to be multimedia. Finally, stored data may refer to different moments in time so that it could be possible to compare the single performances across time or to have a picture of the different elements on a determined date.

The examples are meant to give a practical representation of how the framework of the map-based application can adapt to different domains and user requirements.

What is still missing is an ontological representation of the time dimension. Time is characterised by different levels of granularity and different aspects can be associated with it. A quite complex but complete representation is the *Ontology of the Italian Application Profile for Time* [102], of which a graphical prospect is provided in Figure 18. The key class is the *Temporal Entity* ("temporal attribute" with respect to the content item), characterised by subclasses and parameters.

This portion of ontology aims to be able to classify the uploaded data according to temporal tags, like the day of the week, the month, or the year, or to assign a time instant or a time interval allowing the user to investigate time-related queries. The representation of a *time hierarchy* facilitates information retrieval rather than a single timestamp. For instance, a user may be interested in investigating the sport-related journalistic news published during a certain month in a certain geographic area.

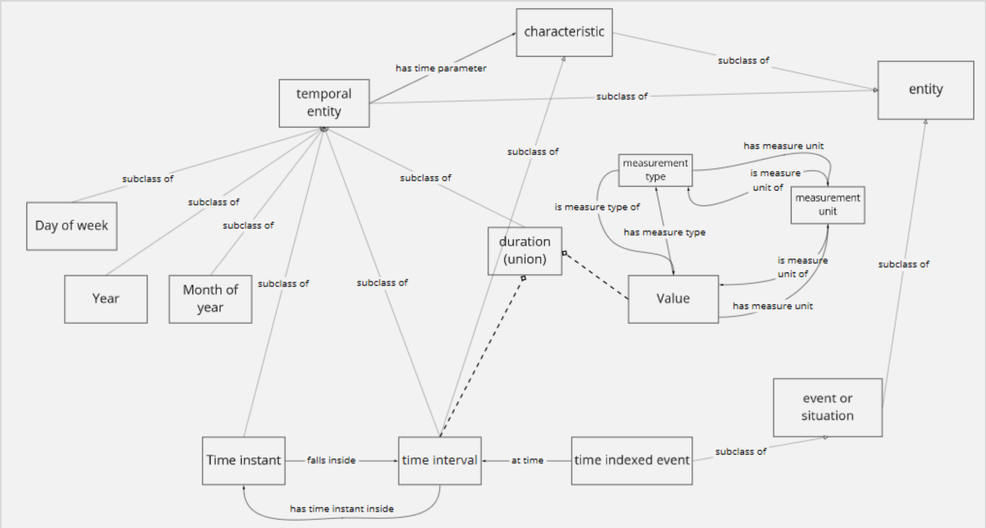


Figure 18 - Time Ontology: Italian application profile

The complexity of this structure can be changed depending on the needs of the single application of the framework. Based on the time-related information that the application requires to monitor, the number of classes and relationships can be reduced or expanded. For instance, for the sake of the demonstrative prototype of this dissertation, the temporal unit of measure is typically the *day*, as the prototype will store web articles published on a certain date, referring to events classified according to the starting and ending day; for this reason, the portion of the ontology related to the *Measurement Type* and *Measurement Unit* can be taken for granted and omitted from the schema.

3.2.4 MAGIS ontology

Figure 19 outlines the general meta-ontology envisioned for MAGIS framework. Being a meta-ontology, it is a very abstract structure that represents the key concepts that should be managed when the framework is implemented in a practical application. This structure is then required to be made concrete into a domain-specific second-level ontology according to the application field.

The schema is divided into two main areas enclosed into dotted frames. On the top part of the figure, the classes related to the management of cartographic data are placed. The central class is the *Map Element*, representing the generic item that constitutes a certain portion of the map layer. It can be related to a punctual element (like a building, or a monument), a linear one (like a street or a river) or even an area (e.g., a park or a lake). Around the element, some examples of sub-classes are represented. The class *Area* represents a generic surface area that can be dynamically used according to the needs; for instance, it could delimitate a city, a region, or a generic area of a defined extension (e.g., a 10 km^2 area around a building). As it could be imagined, these examples are not disjoint, as a specific content item can be part of more than one class. Hence the reflexive relation "*is a*" on the class *Map Element*. For instance, the map element "Park" is also an "Area" with defined boundaries. Some properties related to the geographic and geometric nature of the map element are also added, as some data formats from some geographic data providers also include this type of information.

On the bottom part of the figure, the second dotted frame encloses the classes related to content items managed by the framework. The central element is the *Content*, which is surrounded by a set of other classes and subclasses that represent its attributes and properties. A group of classes deals with the nature of the content item (i.e., the type of multimedia and a link to its access), including the temporal dimension. The media element is managed as a link so that potentially any type of document or unstructured dataset can be managed. The relevant metadata need to be stored to ensure reliable provenance and data validation. A central role is also played by the *Topic* of the multimedia, which is the key for the classification phase. For simplicity, the *Topic* entity only shows the hierarchical subclass relation ("*related to*") but other types of relationships between topics are allowed.

The link between the two main areas is the class *Point of Interest*: as mentioned in previous sections, it represents the connection between the map layer, where it highlights a determined geographic location, and the content layer, as it encloses specific news or noticeable event linked to that location. From a data visualisation point of view, a POI is typically represented by a marker (graphic icon) on a certain latitude-longitude combination. However, since information is often related to a geographic area rather than a single point, a POI can also correspond to an area, sometimes referred to as Region of Interest (ROI). The choice to separate the POI from the content items allows to aggregate more contents related to the same place (such as the artworks housed in a museum or events that happened in the same location).

In this representation, the temporal dimension is described as a property of *Content*. The class *Time* simplifies the time management of content items by representing the moment the item is referred to. For instance, it can map the date when a web article was written or published, when a picture was taken, or when a document was edited. It can also represent the moment when the *event* presented in the content item takes place, like the date of a theatre performance or the time window of a museum exhibition. In this way, the complex time ontology of Figure 18 can be simplified, and the class *Time* can be declined into a list of time-related pieces of information, each regarding an aspect of the lifetime of the specific content item (Figure 20).

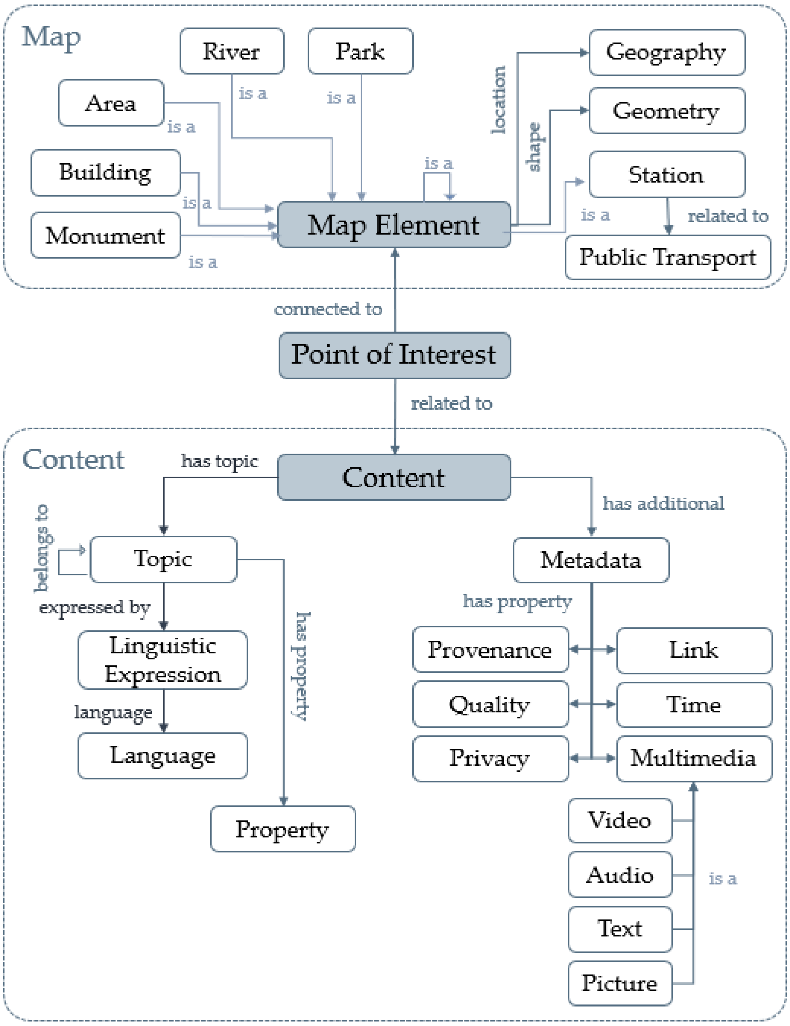


Figure 19 - Ontology of MAGIS framework

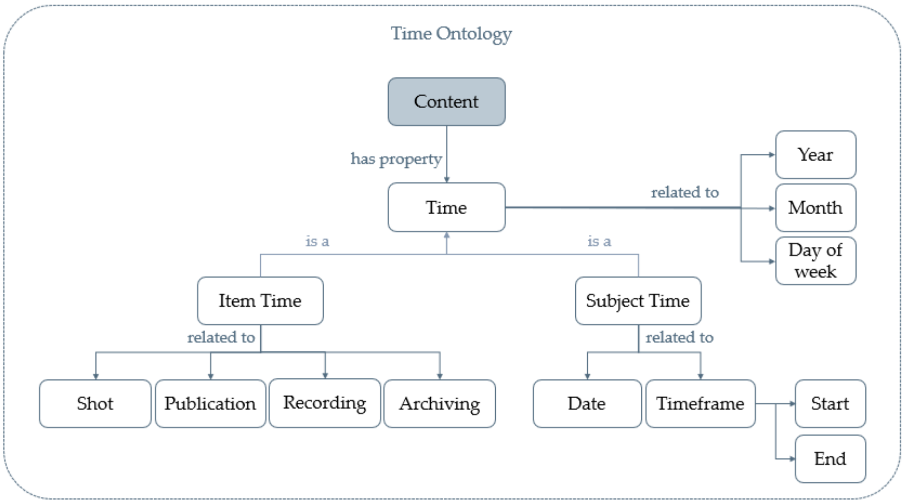


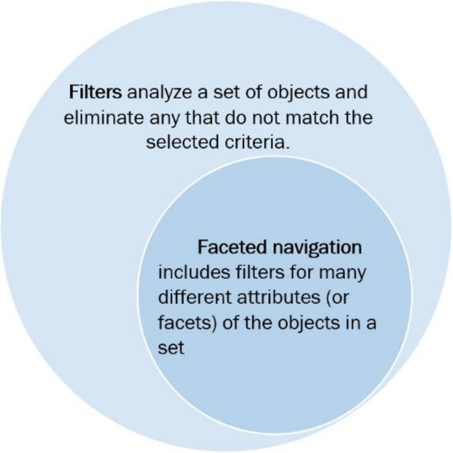
Figure 20 - Details about time dimension in the ontology (Time class)

3.2.5 UI application

To deeply understand the real benefits of a two-layer ontological structure, it may be useful to think about the visualisation phase of the project. Developing the user interface (UI), the domain-specific ontology (second level) and its content items can be effectively used to create dynamic filters or content layers, aimed to reduce the volumes of contents displayed to the user. For example, the application could be able to discriminate by topic or groups of topics, to show just a subset of elements per interrogation. A more sophisticated implementation (not discussed in this project) could imply using the described data structure to support queries in natural language.

If the ontology is sufficiently wide and articulated, it could be employed in the so-called *multi-faceted search*, namely filters that independently combine several classification criteria. *Faceted search* is an “upgraded” version of filtering that can significantly reduce information overload by organising search outputs into groups with different topics. Faceted search provides multiple dimensions and complex filters for users, hence the users can be able to determine the groups they are interested in and find the desired information more quickly [103]. In other words, items are classified along multiple dimensions, called *facets*, rather than a single taxonomic hierarchy, which allows the single content elements to be accessed in multiple ways. Facets are a subset of filters that allow customers who know what they want to narrow by what's important, based on the search terms they use, and without limiting their choice to exactly one item. Facets also help those who aren't sure what they want to outline some of the attributes they might want to consider. They can eventually be useful for teaching customers the kinds of questions to ask [104]. Figure 21 summarises the difference between traditional filtering and faceted search, while Figure 22 shows an example of facets.

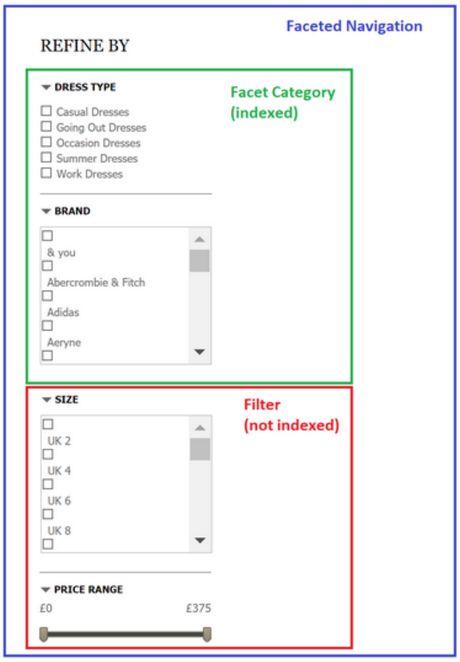
In the case of this project, a multi-faceted search can be created filtering contents based on the classes of the general ontology, such as the temporal dimension, the privacy level, the content nature etc. Figure 23 shows an example of faceted search that could be implemented for the framework.



Filters analyze a set of objects and eliminate any that do not match the selected criteria.

Faceted navigation includes filters for many different attributes (or facets) of the objects in a set

Figure 21 - Filters vs. Facets [105]



▾ DRESS TYPE

- □ Casual Dresses
- □ Going Out Dresses
- □ Occasion Dresses
- □ Summer Dresses
- □ Work Dresses

▾ BRAND

- □ & you
- □ Abercrombie & Fitch
- □ Adidas
- □ Aeryne
- □

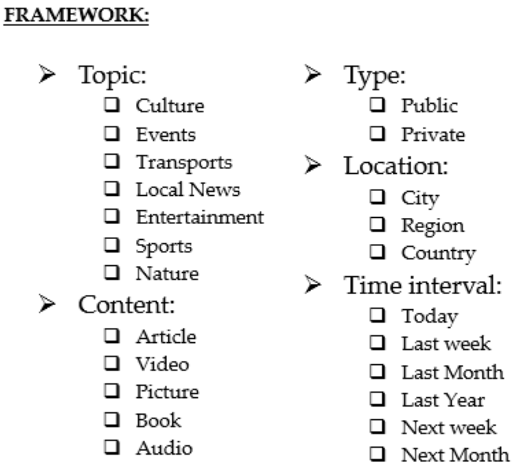
▾ SIZE

- □ UK 2
- □ UK 4
- □ UK 6
- □ UK 8
- □

▾ PRICE RANGE

£0 £375

Figure 22 - Faceted Navigation Example [106]



FRAMEWORK:

- ➢ Topic:
  - □ Culture
  - □ Events
  - □ Transports
  - □ Local News
  - □ Entertainment
  - □ Sports
  - □ Nature
- ➢ Content:
  - □ Article
  - □ Video
  - □ Picture
  - □ Book
  - □ Audio
- ➢ Type:
  - □ Public
  - □ Private
- ➢ Location:
  - □ City
  - □ Region
  - □ Country
- ➢ Time interval:
  - □ Today
  - □ Last week
  - □ Last Month
  - □ Last Year
  - □ Next week
  - □ Next Month

Figure 23 - Faceted Search for the Framework

3.2.6 MAGIS Ontology in OWL

The objective of this section is to translate the ontology built at a high level and shown in Figure 19 into a professional tool for ontology management. Even if an OWL ontology has not been used in the prototype described in Chapter 4, this section is aimed to show an example of ontology-specialised applications and to suggest, step by step, a possible approach for future implementations.

The development of the ontology for MAGIS framework was guided by the handbook by M. Horridge [94] and it was implemented using Protégé (version 5.5.0).

Web Ontology Language (OWL)

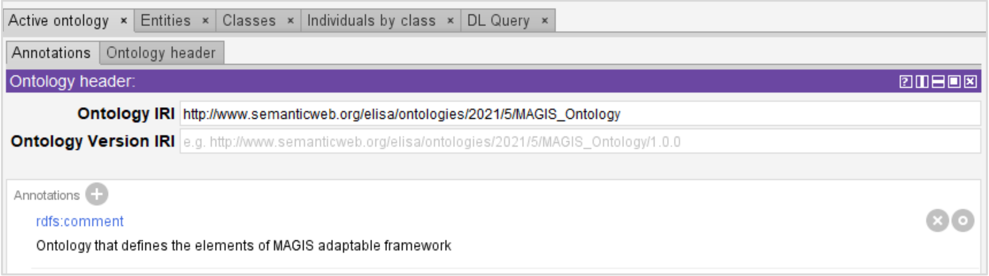
The development of the ontology in Protégé is carried out adopting the *Web Ontology Language* (OWL), a family of knowledge representation languages for authoring ontologies [107].

The key aspects of OWL ontologies consist of Individuals, Properties, and Classes, as introduced in [94]:

- *Individuals* represent objects in the domain of interest. Individuals are also known as instances and can be referred to as being "instances of classes".
- *Properties* are binary relations on individuals - i.e., properties link two individuals together.
- *Classes* are interpreted as sets that contain individuals. Classes may be organised into a superclass-subclass hierarchy, which is also known as a taxonomy. Subclasses specialise ('are subsumed by') their superclasses. The word *concept* is sometimes used in place of class. Classes are a concrete representation of concepts. In OWL classes are built up of descriptions that specify the conditions that must be satisfied by an individual for it to be a member of the class.

OWL ontology creation

After downloading and opening Protégé, a new ontology is created, and it is renamed "MAGIS_Ontology". In the "Active Ontology Tab", in the "Ontology Annotations" view, an annotation describing the aim of the ontology can be added. The comment states "*Ontology that defines the elements of MAGIS adaptable framework*" (Figure 24).



Ontology IRI http://www.semanticweb.org/elisa/ontologies/2021/5/MAGIS_Ontology
Ontology Version IRI e.g. http://www.semanticweb.org/elisa/ontologies/2021/5/MAGIS_Ontology/1.0.0

rdfs:comment
Ontology that defines the elements of MAGIS adaptable framework

Figure 24 - Protégé interface: Ontology Name and Annotation Comment

To define the structure of the ontology, its classes need to be created, together with the properties linking them.

OWL Ontology Classes

Using the "Classes Tab" under the "Entities Tab", *Classes* are defined, that are the main building blocks of an OWL ontology. Although there are no mandatory naming conventions for OWL classes, the so-called *CamelBack notation* is used so that all class names start with a capital letter and do not contain spaces.

The empty ontology contains one class called Thing. As previously mentioned, OWL classes are interpreted as sets of individuals (or sets of objects). The class Thing represents the set containing *all* individuals; because of this, all classes are subclasses of Thing.

The class PoinOfInterest is created as a subclass of Thing. The class PoinOfInterest aims to represent a point of the map that will be considered precisely of users' interest and to which some multimedia content will be associated. The classes Content and MapElement are also created as subclasses of Thing, namely as equivalent classes as PoinOfInterest (Figure 25). The objective is to model the fact that a point of interest must be identified by an element on the map (MapElement), and it is going to be associated with some types of information providing additional content (Content).

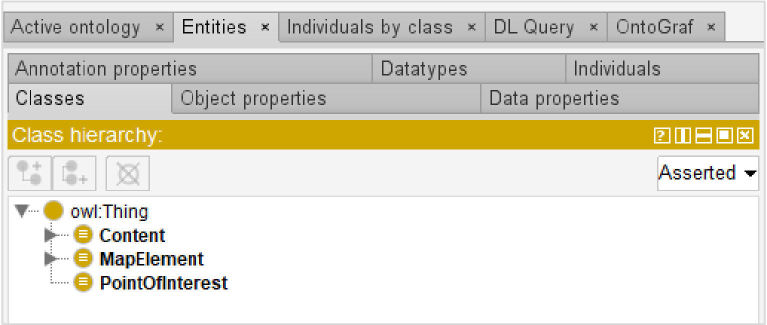


Figure 25 - Protégé interface: Creation of classes

Once added to the ontology, the classes need to be made *disjoint*, so that an individual (or object) cannot be an instance of more than one of the specified classes. This is because OWL Classes are assumed to 'overlap' when they are first created. To 'separate' a group of classes, a proper function is offered by Protégé, that is the 'Disjoints classes' button, at the bottom of the 'Class Description' view. Selecting one class, this option allows identifying the other classes that are disjoint to it. For example, the class MapElement highlighted on the left side of Figure 26 is set as disjoint to the class Content (option "Disjoint with" at the bottom right of the figure), as a content item cannot be a map element as well, being the two classes connected by the intermediate class PointOfInterest.

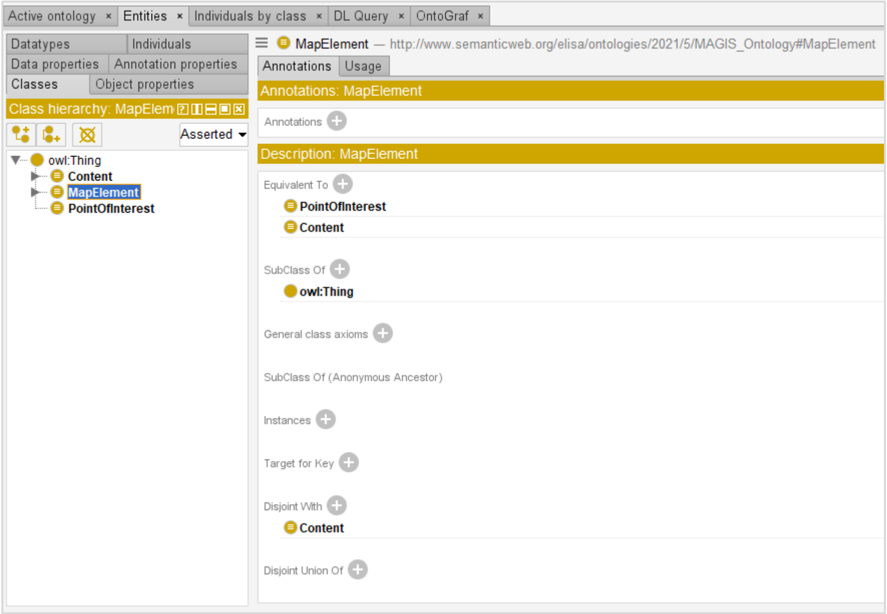


Figure 26 - Protégé interface: Disjunction of classes

Class hierarchies can now be created (Figure 27). The class Content has subclasses Metadata and Topic. The class MapElement has subclasses Area, Building, Monument, Park, River, Geography, Geometry and Station, which in turn has subclass PublicTransportLine. All the classes disjoint, except for the Area class, which can be overlapped with its sibling classes.

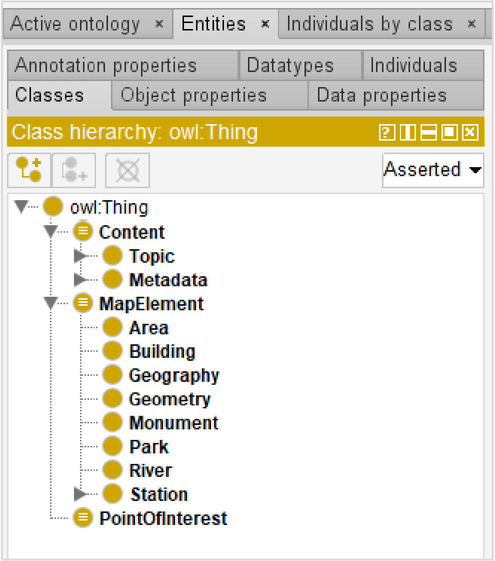


Figure 27 - Protégé interface: Class Hierarchies

The different classes cannot remain isolated but require some connections to be linked one to the others. These connections are the *Properties* (also called *Slots* in Protégé).

OWL Ontology Properties

OWL Properties represent relationships between classes. There are two main types of properties, Object properties and Datatype properties. In MAGIS ontology, only object properties are created.

Object properties are relationships between two individuals. Properties may be created using the 'Object Properties' tab. Although there is no strict naming convention for properties, MAGIS ontology complies with the notation in [94], recommending that property names start with a lower case letter, have no spaces and have the remaining words capitalised. The guide also recommends that, when possible, properties are prefixed with the word 'has', or the word 'is', for example, hasPart, isPartOf.

Properties may have a *domain* and a *range* specified, which represent the classes of individuals connected by the property. Properties link individuals from the domain to individuals from the range.

To deeply understand the structure of the ontology, the main characteristics of a generic property are summarised as follows:

- *Functional*: If a property is functional, for a given individual, there can be *at most* one individual that is related to the individual via the property. An example of a functional property is hasBirthMother because someone can only have one birth mother.
- *Inverse functional*: If a property is inverse functional, it means that the inverse property is functional. For a given individual, there can be at most one individual related to that individual via the property.
- *Transitive*: If a property P is transitive, and the property relates individual a to individual b, and also individual b to individual c, then we can infer that individual a is related to individual c via property P.
- *Symmetric*: If a property P is symmetric, and the property relates individual a to individual b, then individual b is also related to individual a via property P.
- *Asymmetric*: If a property P is asymmetric, and the property relates individual a to individual b, then individual b cannot be related to individual a via property P.
- *Reflexive*: A property P is said to be reflexive when the property must relate individual a to itself.

Going back to MAGIS ontology, the properties connecting the classes introduced beforehand are created as follows (Figure 28).

The property belongsTo is a reflexive property on the class MapElement, making this class both a domain and a range. For instance, this is useful to indicate that a map element like a building belongs to another map element like a geographic area (e.g., a city, a region, or a generic area). The property isRelatedTo is a symmetric property connecting the domain PointOfInterest to the range Content. This means that if a point of interest A is related to content information B, then that piece of information B is also related to the point of interest A. Finally, the property isRepresenting is a functional property connecting the domain PointOfInterest to the range MapElement. This models the fact that a point of interest is representing *at most* one map element so that it cannot represent more elements of the map. This is made to avoid inconsistencies and conflicts.

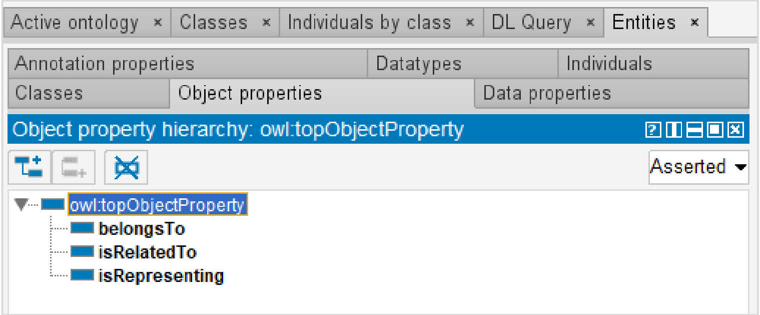


Figure 28 - Protégé interface: Creation of properties

At this point, the concept of *property restrictions* needs to be introduced. A restriction describes a class of individuals based on the relationships that the members of the class participate in. In other words, a restriction is a kind of class, in the same way that a named class is a kind of class. There are two main types of property restrictions:

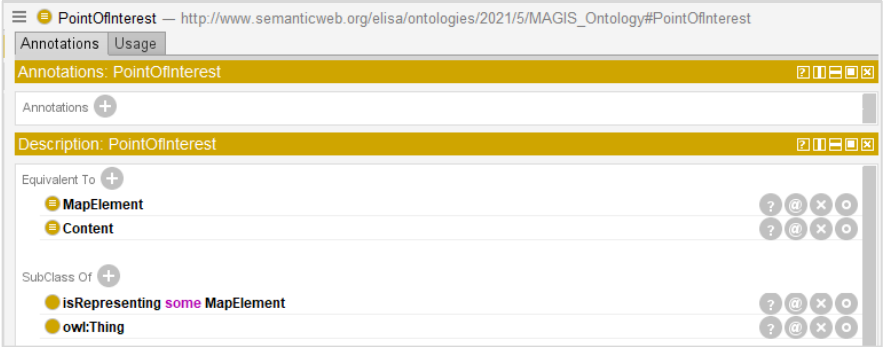
- *Existential restrictions* describe classes of individuals that participate in *at least one* relationship along with a specified property to individuals that are members of a specified class. Existential restrictions are also known as *Some* restrictions.
- *Universal restrictions* describe classes of individuals that for a given property *only* have relationships along with this property to individuals that are members of a specified class.

A restriction describes an anonymous class (an unnamed class). The anonymous class contains all the individuals that satisfy the restriction – i.e., all the individuals that have the relationships required to be a member of the class.

In MAGIS ontology, a restriction to PointOfInterest is needed to specify that PointOfInterest *must represent* a MapElement. For something to be a PointOfInterest, it must have *(at least one)*

MapElement. In other words, a PointOfInterest is a subclass of the things that have at least one MapElement (Figure 29).

The class PointOfInterest has been described to be a subclass of Thing and a subclass of the things that are some kind of MapElement. Notice that these are *necessary conditions* — if something is a PointOfInterest it is necessary for it to be a member of the class Thing (in OWL, everything is a member of the class Thing) and to represent a kind of MapElement. More formally, for something to be a PointOfInterest it is necessary to be in a relationship with an individual that is a member of the class MapElement via the property isRepresenting.



PointOfInterest — http://www.semanticweb.org/elisa/ontologies/2021/5/MAGIS_Ontology#PointOfInterest

Annotations | Usage

Annotations: PointOfInterest

Annotations

Description: PointOfInterest

Equivalent To

MapElement

Content

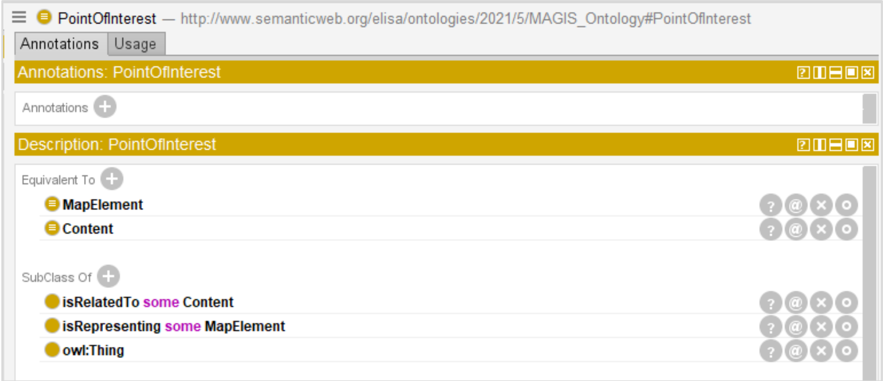
SubClass Of

isRepresenting some MapElement

owl:Thing

Figure 29 - Protégé interface: Existential restriction PointOfInterest - MapElement

In the same way, another restriction to PointOfInterest is needed to specify that PointOfInterest *must be related to (some)* Content (Figure 30).



PointOfInterest — http://www.semanticweb.org/elisa/ontologies/2021/5/MAGIS_Ontology#PointOfInterest

Annotations | Usage

Annotations: PointOfInterest

Annotations

Description: PointOfInterest

Equivalent To

MapElement

Content

SubClass Of

isRelatedTo some Content

isRepresenting some MapElement

owl:Thing

Figure 30 - Protégé interface: Existential restriction PointOfInterest - Content

All properties created so far include *necessary conditions* to describe them. Necessary conditions can be read as: "If something is a member of this class then it is *necessary* to fulfil these conditions". With necessary conditions alone, it is not possible to say that "If something fulfils these conditions then it *must* be a member of this class". A class that only has necessary conditions is known as a *Primitive Class*.

The current description of PointOfInterest means that if something is a PointOfInterest it is necessarily a Thing, and it is *necessarily* related to (*at least one*) Content, which is expressed through a *necessary* condition. However, it is also true that whatever individual is related to a Content is necessarily a PointOfInterest. This is possible to be expressed through a *necessary AND sufficient* condition, which means that not only being related to a Content is the necessary condition for being a PointOfInterest, but it is also sufficient to determine that any (random) individual that satisfies this condition *must be* a member of the class PointOfInterest. A class that has at least one set of necessary and sufficient conditions (in this case, Content) is known as a *Defined Class*.

On the other side, even if a PointOfInterest is something that *must* represent a MapElement (necessary condition), it is not true that a MapElement is necessarily a PointOfInterest. Being a map element is not a sufficient condition to be a point of interest as well.

OWL Ontology completion

At this point, it is possible to proceed with defining classes and properties of the ontology with a higher level of detail. This paragraph presents first the section of the ontology that refers to the geographic components, i.e., the classes and properties that specify the elements that constitute the map. After that, Content subclasses are described. To simplify the reading, the description is presented in bullet points as follows.

- The class Station is a subclass of MapElement. The property isCloseTo is a symmetric property connecting the domain MapElement to the range Station.
- The class PublicTransportLine is added as a subclass of Station. The property isConnectedTo is a symmetric property connecting the domain Station to the range PublicTransportLine. An existential restriction is also added to

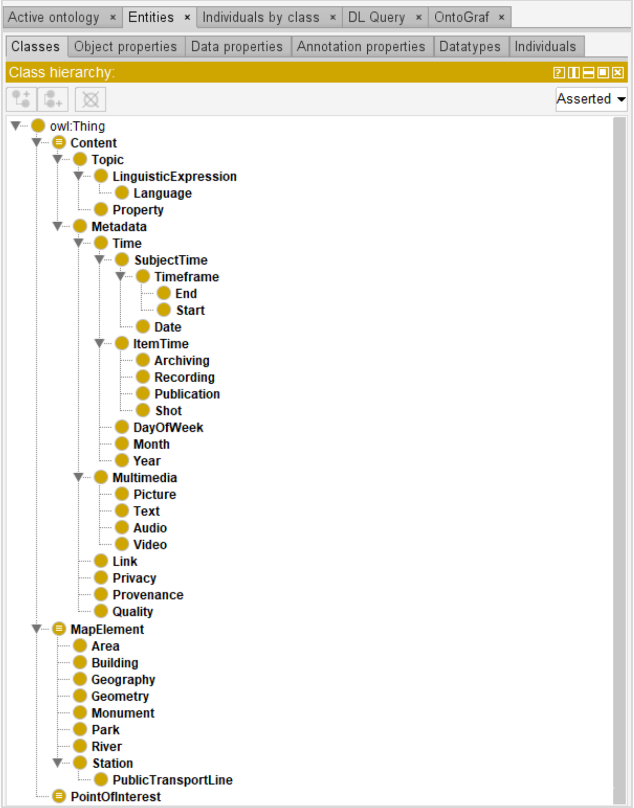
Station pointing that a Station must be connected to (*some*) PublicTransportLine.

- The class Geography is added as a subclass of MapElement. Geography is disjoint with the other subclasses of MapELement. The property isLocated is a functional property (i.e., for each MapElement there can be at most one Geography that is associated via the isLocated property) connecting the domain MapElement and the range Geography. An existential restriction is also added to MapElement pointing that a MapElement must be located into (*some*) Geography element.
- The class Geometry is added as a subclass of MapElement. Geometry is disjoint with the other subclasses of MapELement. The property hasShape is a functional property (i.e., for each MapElement there can be at most one Geometry that is associated via the hasShape property) connecting the domain MapElement and the range Geometry. An existential restriction is also added to MapElement pointing that a MapElement must be shaped into (*some*) Geometry element.
- The class Topic is added as a subclass of Content. The reflexive property belongsTo is linked to the class Topic, meaning that topics can be organised in hierarchies, where lower-level topics belong to wider and more generic topic categories. The property isReferredTo is a transitive property connecting the domain Content to the range Topic. The reason for the transitive property is that if a piece of content is referred to a certain topic T, and that topic belongs to a more generic topic T_1, then that piece of content is also referred to topic T_1. An existential restriction is also added to Content meaning that Content must be referred to (*some*) Topic. The association of a content item to its topic is required to display the element in the correct domain-specific layer in the visualization interface of the web-based application.

- The subclass Property is added to Topic. The property hasProperty is an asymmetric property connecting the domain Topic to the range Property. No existential restrictions are added, as Property represents an optional information.
- The subclass LinguisticExpression is added as a subclass of Topic. The property isExpressedBy is a symmetric property connecting the domain Topic to the range LinguisticExpression. An existential restriction is also added to Topic meaning that Topic must be expressed by (*some*) LinguisticExpression. This latter class is in turn superclass of Language. The property isLanguage is a symmetric sub-property of isExpressedBy, which relates the domain LinguisticExpression to the range Language. An existential restriction is added to LinguisticExpression indicating that a LinguisticExpression must be related to (some) Language. These classes are aimed to make the application language-independent, as it is possible to "translate" the different topics into several languages, based on the specific application.
- The class Metadata is added as a subclass of Content. The property hasAdditional is a functional property connecting the domain Content to the range Metadata. An existential restriction is also added to Content saying that Content must be referred to (some) Metadata.
- The following subclasses are added to Metadata: Provenance, Quality, Privacy, Link, Multimedia, Time. All subclasses are disjoint, except for the classes Link and Multimedia, that can overlap. The property hasMetaProperty is an asymmetric property connecting the domain Metadata to the six subclasses as ranges.
- The following subclasses are added to Multimedia: Text, Picture, Video, Audio. All subclasses are disjoint, as they depend on the format of the multimedia. The property isA is an asymmetric property connecting the for subclasses to the superclass Multimedia.

- Time is the superclass of a further class hierarchy, as already shown in Figure 19. Year, Month and DayOfWeek are subclasses of Time. The property hasTimeProperty is a functional property relating the class Time to the three subclasses. An existential restriction is added to the property indicating that Time must be associated with (some) combination of the three subclasses. Two more subclasses are added to Time: ItemTime and SubjectTime. The property hasItemTime is a symmetric relationship connecting the domain Time to the range ItemTime, while the property hasSubjectTime relates the class Time to SubjectTime. ItemTime is the superclass of four subclasses: Shot, Publication, Recording, Archiving. SubjectTime is instead the superclass of Date and Timeframe, which in turn is the superclass of Start and End.

Figure 31 reports the whole hierarchy of classes and subclasses created in Protégé, while Table 1 summarises the properties created with related key information.



- owl:Thing
  - Content
    - Topic
      - LinguisticExpression
        - Language
      - Property
    - Metadata
      - Time
        - SubjectTime
          - Timeframe
            - End
            - Start
          - Date
        - ItemTime
          - Archiving
          - Recording
          - Publication
          - Shot
        - DayOfWeek
        - Month
        - Year
      - Multimedia
        - Picture
        - Text
        - Audio
        - Video
      - Link
      - Privacy
      - Provenance
      - Quality
  - MapElement
    - Area
    - Building
    - Geography
    - Geometry
    - Monument
    - Park
    - River
    - Station
      - PublicTransportLine
  - PointOfInterest

Figure 31 - Protégé interface: MAGIS class hierarchy

MAGIS Ontology Properties

Property	Type	Domain	Range	Restriction
belongsTo	Reflexive	MapElement	MapElement	
hasAdditional	Functional	Content	Metadata	Some
hasItemTime	Symmetric	Time	ItemTime	
hasMetaProperty	Asymmetric	Metadata	Time, Quality, Privacy, Link, Multimedia, Provenance	
hasproperty	Asymmetric	Topic	Property	
hasShape	Functional	MapElement	Geometry	Some
hasSubjectTime	Symmetric	Time	SubjectTime	
hasTimeProperty	Functional	Time	Year, Month, DayOfWeek,	Some
isA	Asymmetric	Picture, Text, Audio, Video	Multimedia	
isCloseTo	Symmetric	MapElement	Station	
isConnectedTo	Symmetric	Station	PublicTransport Line	Some
isExpressedBy	Symmetric	Topic	Linguistic Expression	Some
isLanguage	Symmetric	Linguistic Expression	Language	Some
isLocated	Functional	MapElement	Geography	Some
isReferredTo	Transitive	Content	Topic	Some
isRelatedTo	Symmetric	PointOf Interest	Content	Some
isRepresenting	Functional	PointOf Interest	MapElement	Some

Table 1 - Protégé: MAGIS Ontology Properties

Visualise the ontology

The whole ontology can be graphically visualised through the "OntoGraf" option in the "Class views" (Figure 32). On the left part of the picture, the hierarchy of classes is reported, while on the right-hand part the ontology is represented graphically. In the visual representation, it is possible to recognise MapElement, PointOfInterest and Content as subclasses of Thing, and their related class hierarchies and interconnections. Blue arrows represent the classical hierarchical property ("subclass of", namely the "is a" relationship), while coloured arrows represent all the other properties described above.

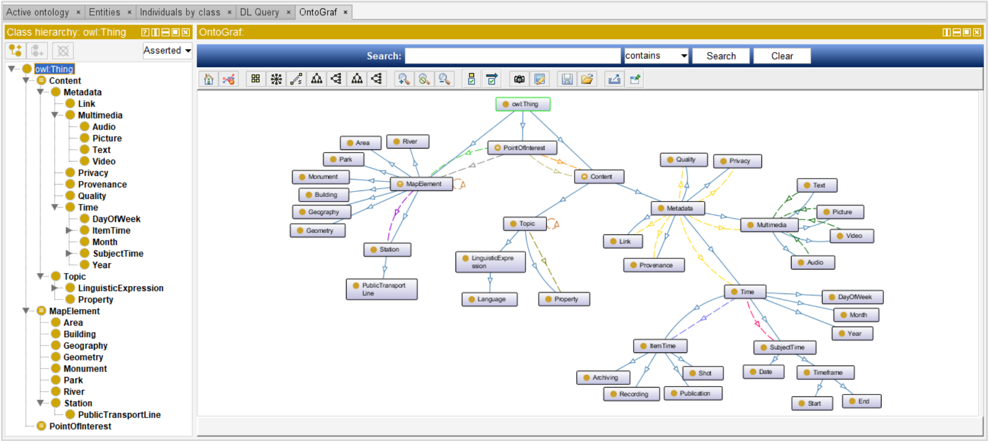


Figure 32 - Protégé interface: Ontology Visualisation

4 The Prototype and its Context: Citizen Journalism

This Chapter presents the demonstrative prototype of the MAGIS framework. It includes some selected modules of the overall framework.

The objective of the prototype is to show in practice the implementation of the map-based application, including content acquisition, automatic content classification, and some visualisation and utilisation examples.

The reference application domain (also referred to as *context*) selected for the practical experimentation is *Citizen Journalism* (CJ). CJ domain is also referred to as *Local News*. It is focused on the collection of web news and journalistic articles that describe a certain geographic area, to get an overview of the evolution/events of that area. Contents can include local news, citizen journalism, environmental reports, fashion, events.

The context is furtherly specified into *topics*, i.e., thematic sub-areas that help describe the content. Each topic is finally associated with a set of tags that classify the specific content. This hierarchical categorisation of tags is clarified in the following chapters.

4.1 MAGIS Toolset

The development of the prototype requires the presence of a geographic layer and the additional data needed to enrich the map, divided into thematic contexts. In the project, the map development and the collection/storage/classification of the multimedia elements have been treated separately.

Three main tools constitute the MAGIS toolset :

1. A web environment with the map layer, meaning the user interface to navigate the geographic dimension.
2. A database where to store the multimedia elements with all their metadata, including some reference to connect the media with the points of interest on the map.
3. A tool for the automatic extraction of the key information related to a media (e.g., title, description, tags, etc.).

The three instruments are described below.

4.1.1 Defining and manging maps

The cartographic data necessary to build the map layer are taken from *OpenStreetMap* (OSM), the free and collaborative world project for the collection of geographic data [108]. The main reason for this choice stands in the open nature of the service: OSM data are covered by a free licence (called *Open Database Licence – OdbL*) that grants an open and free use; OSM maps are instead covered by the *Creative Commons BY SA* licence. Thanks to these open licences, it is possible to access and download all the data present in the database in a completely free way, together with services and tools created by a community of developers.

The possibility to download raw data is an important characteristic of OSM, which allows the creation of services for many diversified uses.

Another important feature for the selection of OSM is the fact that it is a collaborative project, which means that everyone can contribute by enriching or verifying data. The community behind the project is an essential component: not only it improves and uploads data, making the system always updated, but also it is in charge of monitoring its quality. This is a critical aspect because since mostly anyone is free to modify the data, dedicated instruments are required to check how data are modified and correct potential errors and inconsistencies, being them accidental or fraudulent. With this purpose, there are several designated websites and services whose job is to control and signal every variation in the data so that the members of the community can keep them monitored.

For the present prototype, OSM data have been downloaded to create a web environment where the demonstrative project has been developed. Since map development is not the focus of this dissertation, the process to download the data from OSM and create the specific website is not described.

The map environment can be found at the following web address:

http://www.mixmap.it/magis/home/home2.php

A proper section will be dedicated to the map visualisation and user interface. Figure 33 shows an overview.

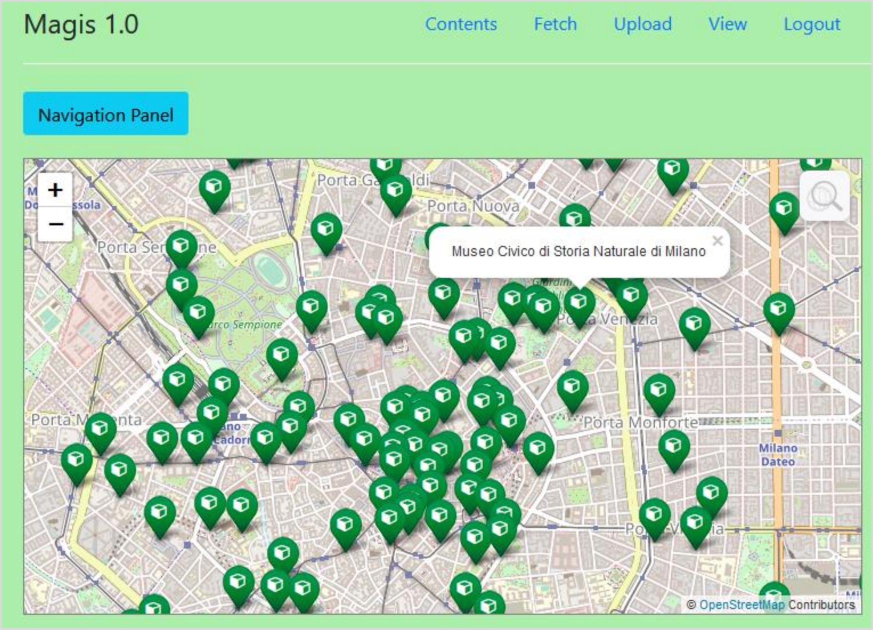


Figure 33 - MAGIS User Interface

4.1.2 The database

A database is needed to store multimedia elements and their metadata, together with some information to link the media to a specific location on the map (i.e., the point of interest).

In general, if a system deals with purely structured data, it is characterised by a regular data structure common to all available information of the same type. Particularly regular structures are the ones typical of relational systems, where data are stored into files with a regular structure called *schema*, in the form of tables. A set of these files composes a relational database, which typically works using the SQL language. On the contrary, when heterogeneous data are involved (e.g., texts, images, time series, or data with an irregular structure), relational databases are no more suitable, and other types of systems are required instead. These are the non-relational (or not-only relational) databases, also called NoSQL (*Not-only SQL*).

In the development of MAGIS framework, if the objective is to download the media and store them directly in a local server, a relational database is not suitable. An option can be relying on a NoSQL database, for instance, a document-based [109] or a column-based NoSQL database [110], where multimedia files are stored into documents that are partitioned into nodes and recollected together when requested. Another option is to use a multimedia database (MMDB), a system that is specialised in the collection and management of multimedia files.

On the opposite, if the multimedia content is not stored locally but retrieved using a web address (e.g., link to a web article or a YouTube video), a multimedia database is not necessary, while a relational system is perfectly functional.

In MAGIS, this second option was selected. The multimedia used in the project are mainly web articles, pictures from online blogs or forums, media from social networks etc. All these data are accessible through a link to a web resource, such as a typical URL; therefore, a relational database was chosen to store the access link to the media, together with its metadata (title, description, publication date etc.).

The selected service provider is *Aruba.it* [111], Italian leader who has activated within its offer the support to *MySQL*: it is an RDBMS, a system for the management of relational databases. In practice, it provides a rapid and well-performant database that allows the creation of web environments offering professional performances. Its services include dedicated servers for data storage. The predefined interaction server-side language is *PHP*.

The structure of MAGIS database for the prototype is shown in the conceptual diagram in Figure 34 (in the schema, only entities and relationships are shown; attributes are specified in the logical representation).

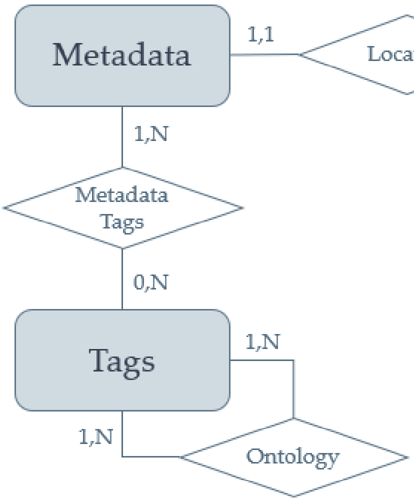


Figure 34 - MAGIS Database Conceptual Schema

The entity Metadata represents the table where multimedia are stored together with the necessary information about the media itself (exactly, its metadata). The relationship element called Located represents the connection between the media object and the map layer. Each media must be associated with a specific location to be displayed on the map, that is the point of interest (POI). A point of interest can be described by more than one media; for instance, the same building can be provided with more than one picture, maybe to represent its evolution over time. Differently, one media must be linked to one single POI; if needed, the same content can be repeated on multiple locations, but the different copies will be treated as separated contents by assigning different identification codes.

Multimedia elements are assigned one or more classification Tags, namely some keywords describing the content of the media. They are needed to classify the content and to facilitate their retrieval and analysis. Since the same tag can be repeated to represent several multimedia objects, and the same media item can be related to multiple topics, the relationship MetadataTags is a many-to-many relationship between the two entities. Ontology is another many-to-many relationship that reflexively links a tag to another tag, as tags can also be related one to the other. This connection is necessary to represent how the different ontological classes are interrelated, allowing to recognise contents of a similar or correlated topic and to propose the right content items to the user.

Moving to the logical representation of the database, five tables need to be created: three of them correspond to the three entities of the conceptual schema (Metadata, POI, Tags), while two additional tables are the bridge tables MetadataTags and Ontology that must be created to represent the many-to-many relationships between the entities Metadata and Tags and the reflexive relationship on Tags.

Metadata

Details about the creation of the table Metadata can be visualised in Figure 35. Each uploaded media is characterised by a MediaCode, an auto-incremental, integer number that uniquely identifies the element. The information that is stored about the media is made of a Title, a Description, the link itself (URL), the Type of media (link, video, photo, audio etc.), information about images included in the link and the provider. All those attributes are stored in *varchar* data types, except for the attribute Description, which is a *text* variable to allow a longer text without a limit in length.

The time dimension is recorded in three attributes: the first is PublicationDate, which is important to know the "age" of that media, especially if it is a web article. Then, a StartDate and an EndDate are recorded in case the media refers to an event like an exposition or a festival that lasts for a prolonged timeframe; in case the event happens on one single date, the StartDate alone can be used (e.g., a commemoration day or a single-date concert). The three attributes are stored according to the *date-time* data type.

Moreover, information about the geographic place is recorded: the attribute CodePOI includes a reference to the entity POI. Since the relationship Located in the conceptual schema is a one-to-many relationship, the key of the entity POI must be included as an external attribute in the entity Metadata. The additional attribute Location includes, in textual form, a reference about the address or a specific location the media is referred[3].

[3] The attribute *Location* is redundant, as each metadata is already linked, though a foreign key, to the point of interest stored in the table *POI*, which also reports the name of the location. This field has been created to facilitate the development of the database, as the geo-linkage of

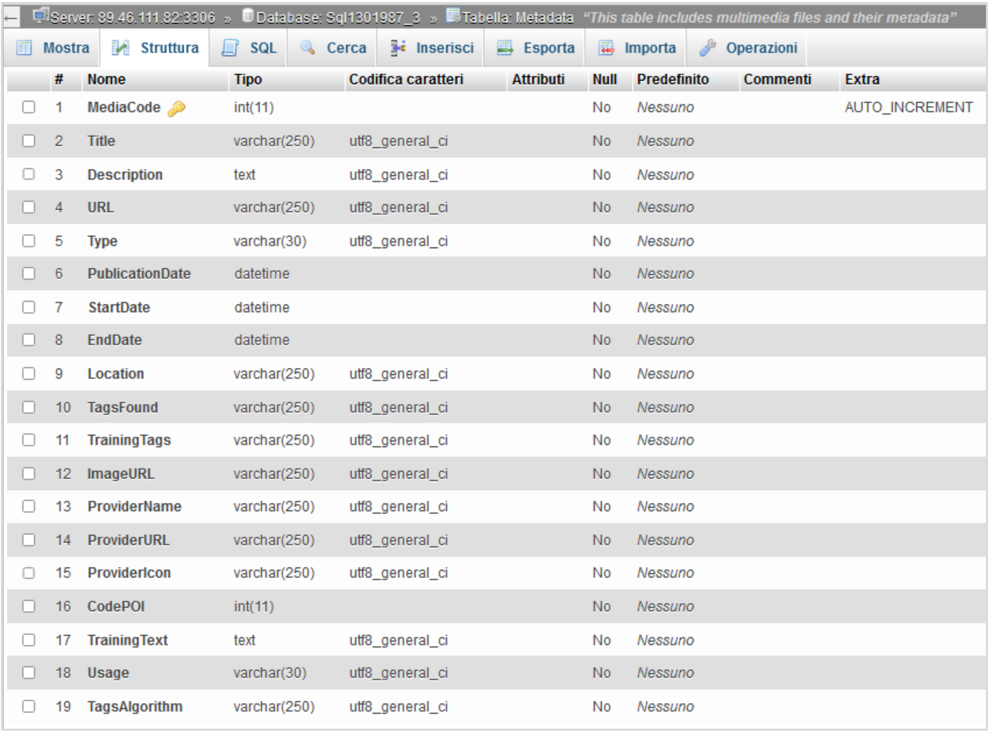
Three attributes of the table are dedicated to the tags describing the media. TagsFound reports the tags automatically extracted by the parsing tool when the text is analysed the first time (see next paragraph about Otero's library). TrainingTags includes the labels of the domain-specific ontology that are manually entered to facilitate the recognition of data. This attribute is used to represent a hierarchy of tags, which the leaf attribute is the one provided by the classification algorithm, but it is also used to associate the content item to several tags. For instance, if the topic of the article is "Cinema", which is related to a public event of cultural type, the hierarchy will be "News; Culture; Public Event; Cinema". A case of multiple hierarchical tags can be constituted by an initiative where a local transport agency offers discounted tickets for a theatre play: in this case, a twofold tag hierarchy is reported, and the attribute will be populated with the string "News; Culture; Public Event; Theatre; Infrastructures & Transports; Transports" (see next sections for the tag ontology). The purpose of this attribute should be to help the framework associate the item to multiple topic-related layers so that it could be displayed in different visualisation configurations.

Finally, TagsAlgorithm collects the tags that have been used specifically in the classification algorithm: in case a content item is used to train the algorithm, this attribute includes the tags that the operator has selected when training or testing the algorithm with that specific tuple. When the prototype will be employed by users, for each new item that will be collected and classified, the attribute will include the tag automatically assigned by the algorithm to the tuple. Please note that this attribute somehow constitutes a sort of foreign key of the table Tags (reporting not the code but the name of the tag), so that the table MetadataTag seems redundant, as the relationship between Metadata and Tags becomes a one-to-many relationship. This is because, in the specific case of this prototype, the classification algorithm returns one single tag per analysed text; based on how future implementations of the framework will be developed, a

multimedia has been carried out in a successive phase compared to the data import. When the geo-reference function was made available, the table *POI* became the official reference about the name of the location, though this attribute was not eliminated, remaining as a support attribute.

different training of the ML algorithm or even a different classification algorithm could return more than one tag for the same content item, which makes the bridge table necessary. In other words, in a complete version of the framework, the relationship between Metadata and Tags is a many-to-many relationship, that needs to be mapped by a proper table so that the attribute TagsAlgorithm is not used. In the specific case of this prototype, this attribute has been used to facilitate content tagging activity, together with ascribing values to the other tables of the database.

Concerning the classification algorithm, the attribute TrainingText represents the text that should be analysed by the ML algorithm, and it is derived from a concatenation of Title, Description and possibly TagsFound. Its purpose is to create an exhaustive source of text that can be used for automatic classification. The attribute Usage indicates for which purpose – training or testing – the specific row of the table has been used; content items that have not been used to build the classification algorithm will admit null values, together with new elements that will be added by end-users.


←
Server: 89.46.111.82:3306 » Database: Sql1301987_3 » Tabella: Metadata "This table includes multimedia files and their metadata"
Mostra
Struttura
SQL
Cerca
Inserisci
Esporta
Importa
Operazioni

	#	Nome	Tipo	Codifica caratteri	Attributi	Null	Predefinito	Commenti	Extra
☐	1	MediaCode	int(11)			No	*Nessuno*		AUTO_INCREMENT
☐	2	Title	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	3	Description	text	utf8_general_ci		No	*Nessuno*		
☐	4	URL	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	5	Type	varchar(30)	utf8_general_ci		No	*Nessuno*		
☐	6	PublicationDate	datetime			No	*Nessuno*		
☐	7	StartDate	datetime			No	*Nessuno*		
☐	8	EndDate	datetime			No	*Nessuno*		
☐	9	Location	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	10	TagsFound	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	11	TrainingTags	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	12	ImageURL	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	13	ProviderName	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	14	ProviderURL	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	15	ProviderIcon	varchar(250)	utf8_general_ci		No	*Nessuno*		
☐	16	CodePOI	int(11)			No	*Nessuno*		
☐	17	TrainingText	text	utf8_general_ci		No	*Nessuno*		
☐	18	Usage	varchar(30)	utf8_general_ci		No	*Nessuno*		
☐	19	TagsAlgorithm	varchar(250)	utf8_general_ci		No	*Nessuno*		

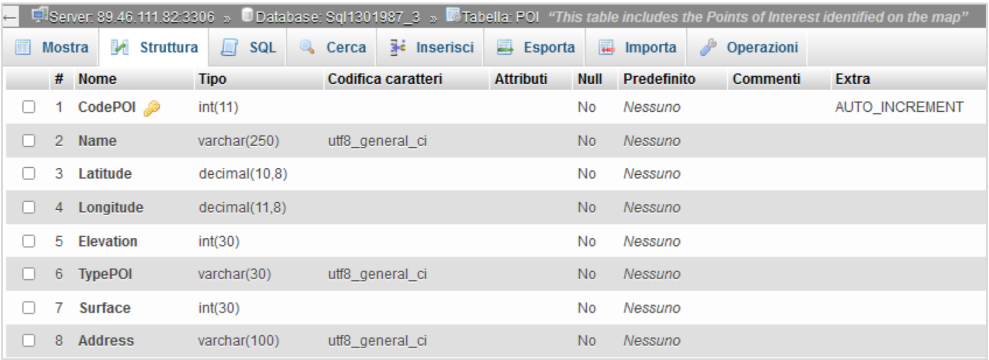
Figure 35 - Details table Metadata

POI

The table POI includes references to the points of interest identified on the geographic map (Figure 36). CodePOI is the auto-incremental identifier of the points of interest, and it is the foreign-key attribute included in the table Metadata.

For each location of interest, the main information stored is related to its Name, the location in terms of Latitude, Longitude and Elevation, the Type of location (e.g., building, area etc.) and additional information like the Surface in case of an area and the Address in case of a punctual location.

The data type of the attributes Latitude and Longitude is decimal, which has a higher precision compared to the floating-point. Latitudes range from -90 to +90 (degrees), so *decimal(10,8)* is ok for that, but longitudes range from -180 to +180 (degrees) so that *decimal(11,8)* is needed [112]. The first number in brackets is the total number of digits stored, and the second is the number after the decimal point. In the two attributes, North and East directories are represented in positive numbers, while the negative sign represents West and South directions.



Server: 89.46.111.82:3306 » Database: Sql1301987_3 » Tabella: POI "This table includes the Points of Interest identified on the map"
Mostra
Struttura
SQL
Cerca
Inserisci
Esporta
Importa
Operazioni

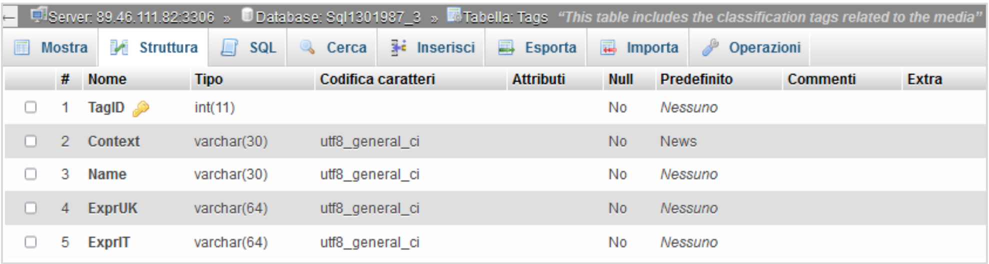
	#	Nome	Tipo	Codifica caratteri	Attributi	Null	Predefinito	Commenti	Extra
□	1	CodePOI	int(11)			No	*Nessuno*		AUTO_INCREMENT
□	2	Name	varchar(250)	utf8_general_ci		No	*Nessuno*		
□	3	Latitude	decimal(10,8)			No	*Nessuno*		
□	4	Longitude	decimal(11,8)			No	*Nessuno*		
□	5	Elevation	int(30)			No	*Nessuno*		
□	6	TypePOI	varchar(30)	utf8_general_ci		No	*Nessuno*		
□	7	Surface	int(30)			No	*Nessuno*		
□	8	Address	varchar(100)	utf8_general_ci		No	*Nessuno*		

Figure 36 - Details table POI

Tags

The third table is called Tags and includes the classification tags related to the media (Figure 37). Each row of the table defines one single tag, that can be associated with several multimedia. Together with the primary key TagID, the table includes the name of the tag both in Italian (ExprIT) and in English language (ExprUK): this is to represent the connection of different linguistic expressions in a simplified way.

Moreover, the Context the tag belongs to is reported, to identify the domain of the tag hierarchy and associated content items. All the attributes are stored in the *varchar* data type, except for the identifier that is an integer number.


←
Server: 89.46.111.82:3306 » Database: Sql1301987_3 » Tabella: Tags
Mostra
Struttura
SQL
Cerca
Inserisci
Esporta
Importa
Operazioni

"This table includes the classification tags related to the media"

	#	Nome	Tipo	Codifica caratteri	Attributi	Null	Predefinito	Commenti	Extra
□	1	TagID	int(11)			No	*Nessuno*		
□	2	Context	varchar(30)	utf8_general_ci		No	News		
□	3	Name	varchar(30)	utf8_general_ci		No	*Nessuno*		
□	4	ExprUK	varchar(64)	utf8_general_ci		No	*Nessuno*		
□	5	ExprIT	varchar(64)	utf8_general_ci		No	*Nessuno*		

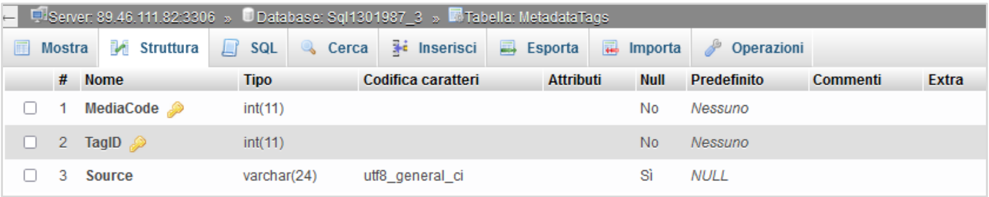
Figure 37 - Details table Tags

MetadataTags

The fourth table is the bridge table connecting multimedia to the correspondent tags. As said, such table is needed as the linkage between the two entities is mapped as a many-to-many relationship, which means that one media can be associated with more than one tag, and one tag can be associated with different media. Without this table, it would not be possible to retrieve what media are assigned to which tag and vice versa. Again, even though in the specific case of the prototype the classification algorithm is trained to return one single tag per text (meaning that a foreign key could be enough), this table has been implemented anyway to make the prototype closer to the final implementation application and to allow the user to manually associate media contents to multiple topics.

The table MetadataTags is composed of three attributes (Figure 38). MediaCode and TagID are foreign keys coming from the previous tables and are both primary keys for this table. In this way, it is possible to map the interdependencies and connections between media content items and their tags. The third attribute, Source, is used to trace the origin of the tag/topic assignment to the item. In the case of the prototype, it will always be the same ML model but in general, it could be assigned manually by a user or there may be more

ML models for different groups of items coming from different sources.

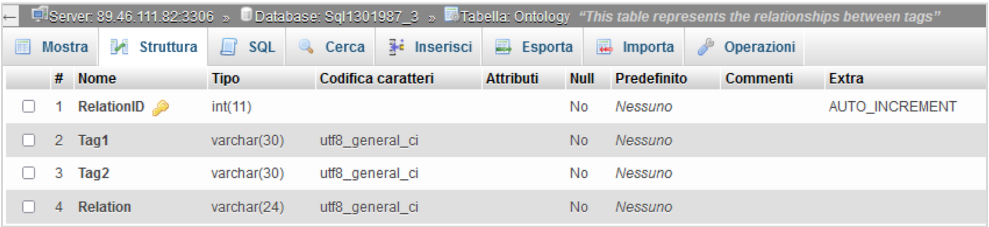

Server: 89.46.111.82:3306 » Database: Sql1301987_3 » Tabella: MetadataTags
Mostra
Struttura
SQL
Cerca
Inserisci
Esporta
Importa
Operazioni

	#	Nome	Tipo	Codifica caratteri	Attributi	Null	Predefinito	Commenti	Extra
□	1	MediaCode	int(11)			No	*Nessuno*		
□	2	TagID	int(11)			No	*Nessuno*		
□	3	Source	varchar(24)	utf8_general_ci		Sì	*NULL*		

Figure 38 - Details table MetadataTags

Ontology

The last table, Ontology, is used to map the relationships between tags (Figure 39). Each tuple is identified by a unique, auto-incremental RelationID and includes the type of connection between a couple of tags, identified as Tag1 and Tag2, via the attribute Relation. The most common types of relations can be "*is_a*", which represents the classical hierarchical connection, or "*related_to*". For example, the tag "Cinema" is a sub-topic of "Culture", so that the two tags are linked by an "*is_a*" relationship; instead, the "New opening" of a shop can be combined with a "Public Event" like an inauguration ceremony, so that the two tags can be linked by a "*related_to*" type of relationship.


Server: 89.46.111.82:3306 » Database: Sql1301987_3 » Tabella: Ontology "This table represents the relationships between tags"
Mostra
Struttura
SQL
Cerca
Inserisci
Esporta
Importa
Operazioni

	#	Nome	Tipo	Codifica caratteri	Attributi	Null	Predefinito	Commenti	Extra
□	1	RelationID	int(11)			No	*Nessuno*		AUTO_INCREMENT
□	2	Tag1	varchar(30)	utf8_general_ci		No	*Nessuno*		
□	3	Tag2	varchar(30)	utf8_general_ci		No	*Nessuno*		
□	4	Relation	varchar(24)	utf8_general_ci		No	*Nessuno*		

Figure 39 - Details table Ontology

4.1.3 Otero's extractor

Supervised ML algorithms need a large quantity of input data to perform adequate training and testing phases. However, entering so much data and registering all their metadata one by one manually is a very long and burdensome job, and on top of that, it is a non-value-added activity. Therefore, having a tool able to automatically extract the key features from a link and upload them to the database is desirable, if not necessary.

The Spanish digital designer Oscar Otero encoded a php library called *Embed* [113] able to extract information from any web page (using oembed, opengraph, twitter-cards, scrapping the html, etc.). It is compatible with any web service (YouTube, Vimeo, Flickr, Instagram, etc.) and has adapters to some sites like Archive.org, Github, Facebook, etc. An online demo can be found at the following web address:

https://oscarotero.com/embed3/demo/

When an URL is entered in the specific taskbar, the tool can extract the main information about it, including a title, a description, the URL itself, possible images included on the website and information about the provider of the web page.

In this prototype scenario, a copy of the Embed library has been made and connected to the relational database and to the following page:

http://www.mixmap.it/magis/home/upload.php

which is the page where the graphical interface of MAGIS map application will be shown (Figure 40). In this way, web articles, pictures, and other data taken as input for the prototype can be easily parsed by Otero's library and most of their metadata are automatically uploaded in the SQL database.

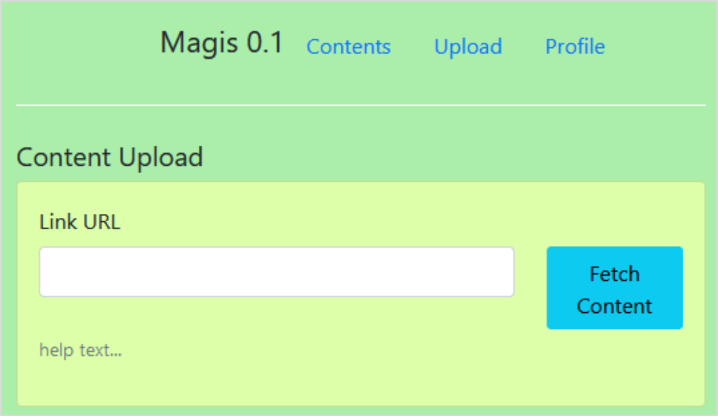
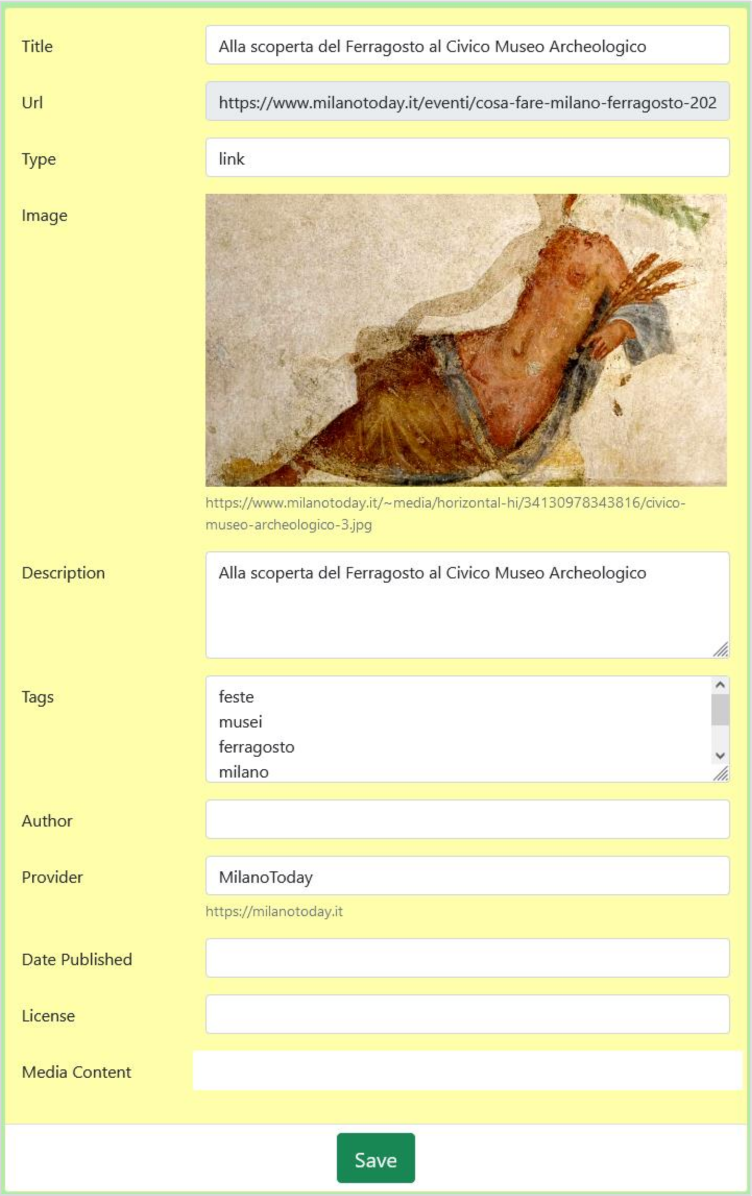


Figure 40 - Interface of Otero's library "Embed" linked to MAGIS application

In Figure 41 an example of the extraction is displayed. Not all pieces of information are always extracted; depending on the website, some fields may remain empty from the automatic parsing, but the user can manually fill the missing data. At the end of the process, the button

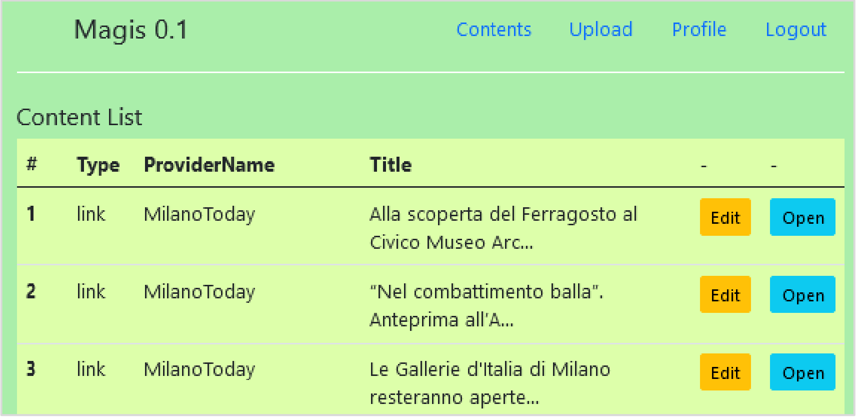
"*Save*" commands the upload of the extracted data into the correspondent attributes of the database. What is interesting in this library is that in many cases some tags are extracted from the web page keywords and can be used to match the labels defined in the ontology.

When data are saved in the database, the tab *Contents* of the web interface shows the list of data that are currently stored; here the type of element, the provider's name and the title are displayed, as shown in Figure 42. The option *Open* is used to open the link attached to the content element, while the option *Edit* opens again Otero's "Embed" webpage of the article allowing the user to modify the related information.



Title	Alla scoperta del Ferragosto al Civico Museo Archeologico
Url	https://www.milanotoday.it/eventi/cosa-fare-milano-ferragosto-202
Type	link
Image	https://www.milanotoday.it/~media/horizontal-hi/34130978343816/civico-museo-archeologico-3.jpg
Description	Alla scoperta del Ferragosto al Civico Museo Archeologico
Tags	feste musei ferragosto milano
Author	
Provider	MilanoToday https://milanotoday.it
Date Published	
License	
Media Content	

Figure 41 - Example of data extraction with Otero's "Embed" library



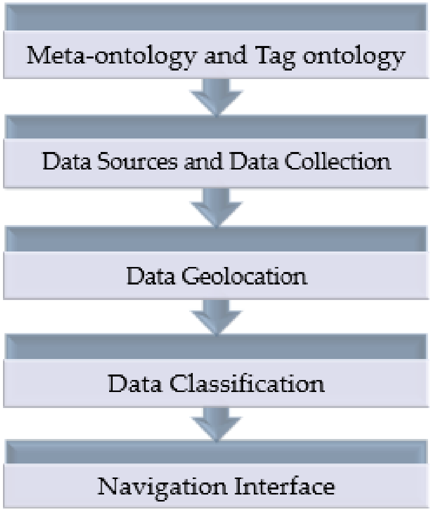
Content List

#	Type	ProviderName	Title	-	-
1	link	MilanoToday	Alla scoperta del Ferragosto al Civico Museo Arc...	Edit	Open
2	link	MilanoToday	"Nel combattimento balla". Anteprima all'A...	Edit	Open
3	link	MilanoToday	Le Gallerie d'Italia di Milano resteranno aperte...	Edit	Open

Figure 42 - Overview MAGIS Content List

4.2 The Prototype – Citizen Journalism

The prototyped thematic area is the context called *Citizen Journalism* (CJ), also referred to as *News*. It is focused on the collection of news and journalistic articles describing a certain geographic area, to get an overview of the evolution/events of that area. It includes local news, citizen journalism, cultural reports, events that can be related to a specific location. The geographic boundary selected for the scope of the prototype is the area of Milan and its surroundings, including the adjacent Lombard provinces if needed.



Meta-ontology and Tag ontology
Data Sources and Data Collection
Data Geolocation
Data Classification
Navigation Interface

Figure 43 - Steps for CJ Prototype

Figure 43 summarises the steps that have been followed to build the prototype. As the first step, the general ontological structure of the framework was adapted to the specific case of the prototype, to define the main classes describing contents and their properties. A further specification of the ontology has been created to map the domain-specific topic ontology, to define the relationships between tags describing the articles. Once defined the data structure, data sources were selected and web articles were collected and stored in the database, after passing through Otero's library. Each web item was associated with a precise location on the map, constituting a point of interest. Each article was then manually associated with a set of tags representing its topic and briefly summarising the content, to be used as a training dataset for the development of the classification algorithm. Subject to some comparative tests, the short-text classification algorithm was chosen, and the articles were classified according to their tags; a part of the data was used as training set, another part as test set, as supervised ML algorithms require. Finally, the ontology of tags was used to implement the filters that discriminate content items in the navigation interface, so to display a subset of articles at a time.

The different phases are well described in the next paragraphs.

4.2.1 Meta-ontology and Tag ontology

The decision to deploy an ontological data structure for this project was inspired by the already mentioned *Ontology-Based Data Access* (OBDA) approach [96], which offers access to stored data according to a three-level architecture:

- The *Ontology* layer provides a formal and high-level description of a domain-specific knowledge base; it does not simply represent the specific data sources, but it is an explicit representation of the domain, which fosters the reusability of acquired knowledge.
- The *Data Sources* layer is made of existing databases that are interrogated by the users.
- The *Mapping* layer connects the previous levels of the architecture, defining the relationships between ontological elements and the data sources. This tier is especially useful when information is spread in several locations so that traditional data access is difficult to achieve without a specific mapping.

Concerning the general ontology of MAGIS framework presented in Chapter 3 (Figure 19), the data structure of this prototype does not include ontological classes about map elements, because the web-based application has been developed relying on a ready-to-use map layer. The ontology developed in the demonstrative project, therefore, includes only classes related to content elements, their properties, and their classifications according to the topic.

As a further specification of content items, a tag ontology was developed, which elements (i.e., classes) are made by the tags used to classify multimedia contents, interconnected by sets of relationships.

Figure 44 can be considered the *meta-ontology* of the prototype, namely the general data structure representing content items and their semantics. In this case, the key classes of the meta-structure are the ones highlighted in colour in the figure:

- Point of Interest represents the geographic location the content item is related to. It is the key class to geo-refer content items.
- Content is the class enclosing the content elements collected in the data acquisition phase.
- Topic is the heart of the tag ontology and the basis of content classification.
- Metadata encloses the different properties and characteristics of content items.
- Time is the metadata property that manages the temporal dimension of the framework. It is related to the time the media was published or created, but also the timeframe or single date the event in the article refers to.

A detailed discussion on the class Topic is something that merits consideration. Not only the meta-ontology of the general structure is required, but also a domain-specific *Tag Ontology* showing the tags needed to classify content items and their relations. This corresponds to the second-level ontology mentioned in section *3.2.3 Tag ontology and meta-ontology*, which is an instance of the first-level meta-ontology in Figure 44. The structure that has been developed has an evident hierarchical component, but is not purely hierarchical, as several cross-topic relationships have been introduced.

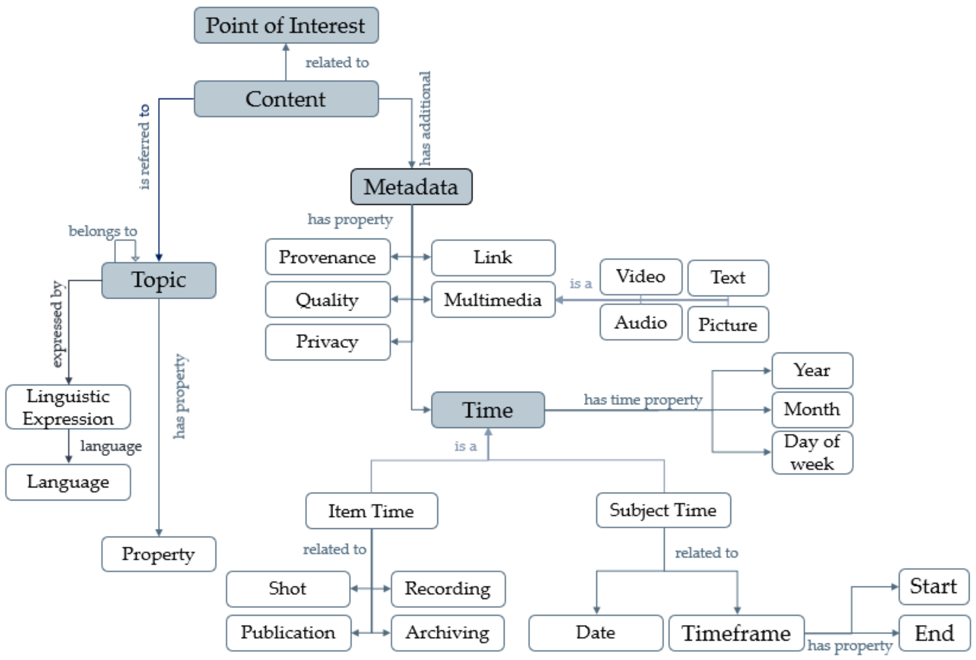


Figure 44 - Meta-ontology of MAGIS Prototype

The web articles taken as input data were chosen within three main topic categories: *Culture*, *Infrastructures & Transports*, *Local News*. These are the topics that are part of the context of *Citizen Journalism* (in brief *News*). For each of the three topics, a set of tags were defined to describe the content of the different articles, shown in Figure 45.

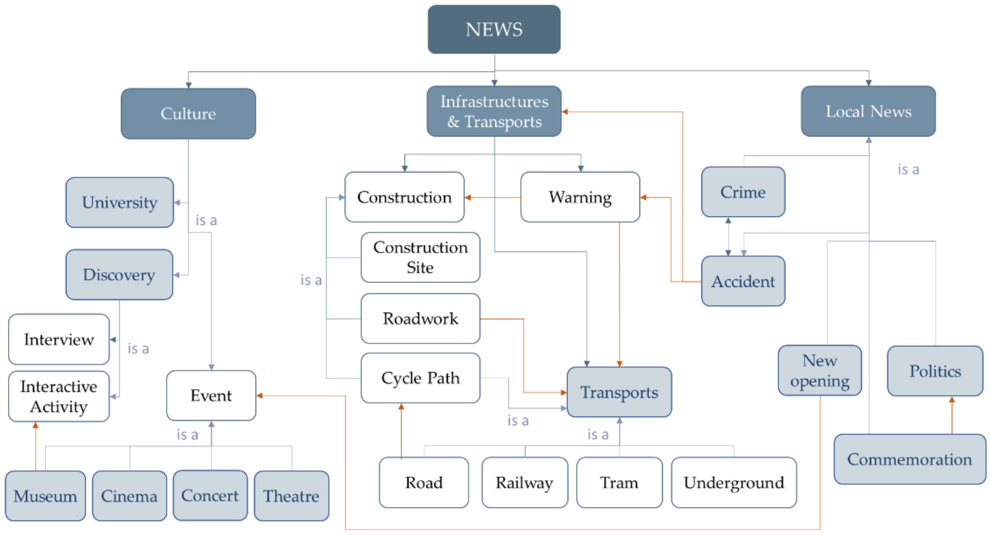


Figure 45 - Tag Ontology Citizen Journalism

The topic called *Culture* includes articles related to cultural initiatives and events, that can regard concerts, cinema, or theatre exhibitions, either temporary or permanent. The tag *Museum* summarises all kinds of exhibitions, including traditional expositions in city museums, but also open-air art installations, murals, photography collections etc. *Interactive Activity* refers to initiatives where people can interact with an artistic or cultural environment, such as laboratories for adults and children organised by a museum, guided visits in cultural locations, etc. *University* regards news about the main universities in Milan, like the opening of a new space for students and researchers, new degree courses, rankings about the best universities at national and international levels. *Discovery* is used to identify a neighbourhood, an area of the city that is worth to be discovered through guided visits or scenic tours; it is often associated with *Interview*, which is used to mention an interview to a person who lives in a specific district and tells the historical background and the points of interest of that area.

The topic *Infrastructures & Transports* collects news related to public transports (including *Railway, Tram, Underground*), warnings about roadworks or construction sites in some locations of the city, and advancements about the construction of cycle paths across the districts of Milan.

Local News instead identifies web articles about what happens daily in the city. Most news is crime-related episodes (like thefts, homicides, violations, arsons, etc.) or accidents (e.g., car crash, fire). Other tags identify commemoration events, political news, and new openings (e.g., a new shop, a new hospital department etc.).

The figure shows different types of relationships between the different tags. The lightest arrows represent the simplest “is_a” relationship, where the leaves are the categorisations of a higher-level tag. Darker arrows indicate more general “related_to” relations; for instance, a *Culture* is a very generic tag, that can be related to multiple sub-topics of different nature. Eventually, orange arrows represent the cross-topic relationships. To mention some examples, as mentioned some paragraphs above, the new opening of a shop can be combined with an event like an inauguration ceremony, or a public exhibition, so that the tags *New opening* and *Event* can be linked by a “related_to” type of relationship, even though they belong to separate hierarchies. In other cases, an accident can cause inconveniences and delays in traffic or on public transport, so that correlated news can also inform about transportation conditions.

To clarify the colouring notation used in the figure, blue cells represent the higher-level topic classes, while grey cells represent the tags specifically used in the prototype as leaves of the three hierarchies.

To recall the structure of the table Metadata in the database, the attribute TagAlgorithm is populated with one single tag among the grey ones, as the short-text classification algorithm has been trained to provide one single tag per analysed text. However, the hierarchy of tags is maintained in the attribute TrainingTags, which can be manually valorised: all the links report the upper-level tag *News* which identifies the CJ context, then the topic (*Culture*, *Infrastructures & Transports*, *Local News*) is specified, and eventually, one or more leaves of the tag tree are assigned. The reader should also remember that multiple tags from different hierarchies can also be assigned to each web article: in this case, they will be reported in the attribute TrainingTags.

4.2.2 Data sources and data collection

The typology of data collected for the CJ prototype includes links to Italian web articles that can be freely accessed online. Most data sources are non-professional Italian online journals and blogs publishing news about Milan and adjoining areas; other links regard news from regional or national journals. In all cases, websites requiring a subscription to read the news were discarded.

Here is a summary of the websites used as data sources:

- Milano Today: https://www.milanotoday.it/
- YesMilano: https://www.yesmilano.it/
- Il Sole 24 Ore: https://www.ilsole24ore.com/
- A Milano Puoi: https://amilanopuoi.com/it/
- Milano Events: https://www.milanoevents.it/
- Repubblica: https://milano.repubblica.it/
- Metro News: https://metronews.it/
- Prima Milano Ovest: https://primamilanoovest.it/
- La Stampa: https://www.lastampa.it/milano
- ATM: https://www.atm.it/it/Pagine/default.aspx
- Info Milano: https://www.infomilano.news/
- Milano All News: https://www.milanoallnews.it/

- Milano Fanpage: https://www.fanpage.it/milano/
- Lecco Today: https://www.leccotoday.it/
- Corriere: https://milano.corriere.it/
- RFI – Rete Ferroviaria Italiana: https://www.rfi.it/it.html
- Trenord: https://www.trenord.it/
- Serravalle: https://www.serravalle.it.html
- Il Giorno: https://www.ilgiorno.it/milano
- Il Giornale: https://www.ilgiornale.it/sezioni/milano.html
- Sky TG 24: https://tg24.sky.it/milano
- Ansa: https://www.ansa.it/lombardia/
- In Topic: https://www.intopic.it/lombardia/milano/
- Mi-Lorenteggio: https://www.mi-lorenteggio.com/
- Mediaset Play: https://www.mediasetplay.mediaset.it/
- Il Giornale della Birra: https://www.giornaledellabirra.it
- YouTube: https://www.youtube.com/
- Ristorante Web: https://www.ristorantiweb.com
- Affari Italiani: https://affaritaliani.it
- Cool in Milan: https://coolinmilan.it
- I Like Milano: https://www.ilikemilano.com
- Milano In Movimento: https://milanoinmovimento.com
- Amici di Casa: https://amicidicasa.it
- Università Statale di Milano: https://lastatalenews.unimi.it
- Università degli Studi di Milano-Bicocca: https://unimib.it
- Panbianco News: https://www.pambianconews.com

511 articles were collected from the mentioned websites and saved in the database. All of them were parsed using Otero's Embed library to extract the key features, while the missing pieces of information were filled manually in the editing form or were entered directly in the table Metadata of the database. In Table 2, an example of data definition in the database is reported.

Metadata

MediaCode	152
Title	Al Museo della Scienza e della Tecnologia arrivano le Collezioni di Studio
Description	Da sabato 24 luglio, i depositi saranno aperti al pubblico con visite guidate settimanali per tutto il periodo estivo. Per la prima volta sarà possibile scoprire le migliaia di oggetti

	custoditi nel cuore del Museo che fanno parte delle sue Collezioni di Studio
URL	https://www.milanoevents.it/2021/08/05/collezioni-studio-museo-scienza/
Type	Link
PublicationDate	2021-08-05 11:43:20
StartDate	2021-07-24 00:00:00
EndDate	2021-08-28 00:00:00
Location	Museo Nazionale della Scienza e della Tecnologia, Via San Vittore, 21
TagsFound	MilanoEvents agenzia eventi agenzie eventi milano agenzia organizzazione eventi agenzie organizzazione eventi milano eventi milano
TrainingTags	news; culture; event; museum; interactive activity
ImageURL	https://www.milanoevents.it/wp-content/uploads/2020/08/museo-scienza.jpg
ProviderName	MILANOEVENTS.IT | News 2.0 ed Eventi a Milano
ProviderURL	https://www.milanoevents.it
ProviderIcon	
CodePOI	26
TrainingText	Al Museo della Scienza e della Tecnologia arrivano le Collezioni di Studio. Da sabato 24 luglio i depositi saranno aperti al pubblico con visite guidate settimanali per tutto il periodo estivo. Per la prima volta sarà possibile scoprire le migliaia di oggetti custoditi nel cuore del Museo che fanno parte delle sue Collezioni di Studio. MilanoEvents agenzia eventi agenzie eventi milano agenzia organizzazione eventi agenzie organizzazione eventi milano eventi milano
Usage	Training
TagsAlgoithm	Museum

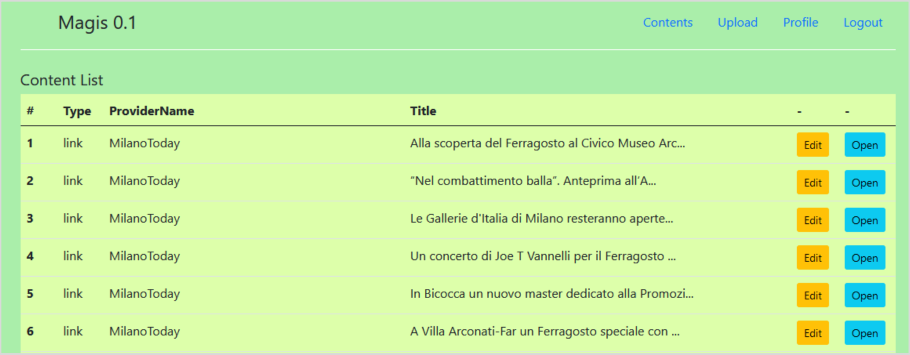
Table 2 - Example data definition

4.2.3 Data Geolocation

When new data are collected and loaded into the database, they must be assigned to a specific geographic location that becomes a *point of interest*. This means geolocating the contents used in the prototype, that become georeferenced data. To do this, a proper extension has been added to the mixmap application interface[4]. This extension is

[4] http://www.mixmap.it/magis/home/list.php

accessible through the tab *Edit* aside each content element, as shown in Figure 46.



Content List

#	Type	ProviderName	Title	-	-
1	link	MilanoToday	Alla scoperta del Ferragosto al Civico Museo Arc...	Edit	Open
2	link	MilanoToday	"Nel combattimento balla". Anteprima all'A...	Edit	Open
3	link	MilanoToday	Le Gallerie d'Italia di Milano resteranno aperte...	Edit	Open
4	link	MilanoToday	Un concerto di Joe T Vannelli per il Ferragosto ...	Edit	Open
5	link	MilanoToday	In Bicocca un nuovo master dedicato alla Promozi...	Edit	Open
6	link	MilanoToday	A Villa Arconati-Far un Ferragosto speciale con ...	Edit	Open

Figure 46 - MAGIS Interface: Content List

In the content editing form, a section *Georeference* shows a geographic map where the position of the content can be defined. When a data content has not been assigned a location yet, the map is displayed as empty (Figure 47).

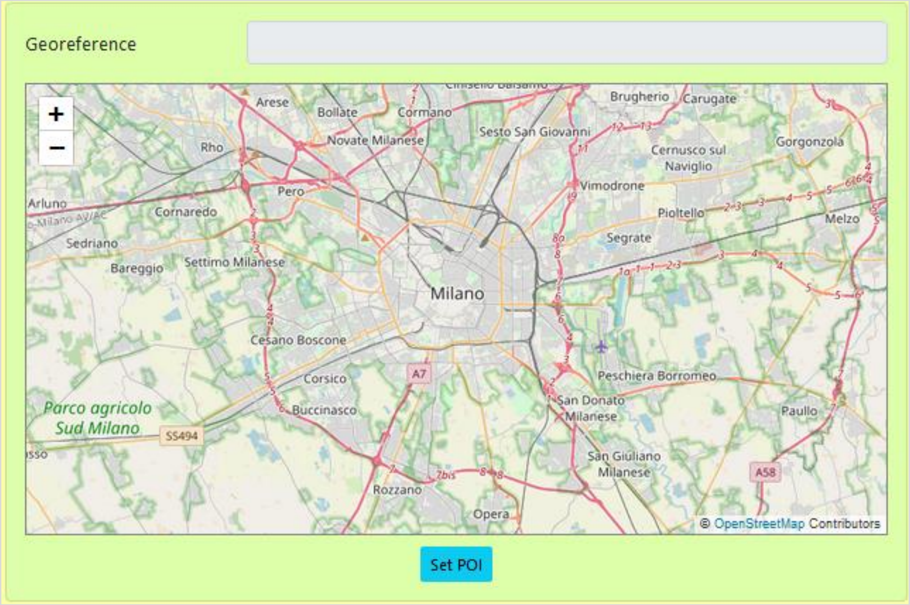


Figure 47 - MAGIS Interface: Georeference Tab

Clicking on the option *Set POI* below the map, a new tab *Point of Interest* pops up with a bigger map displaying the points of interest already created as yellow markers (Figure 48).

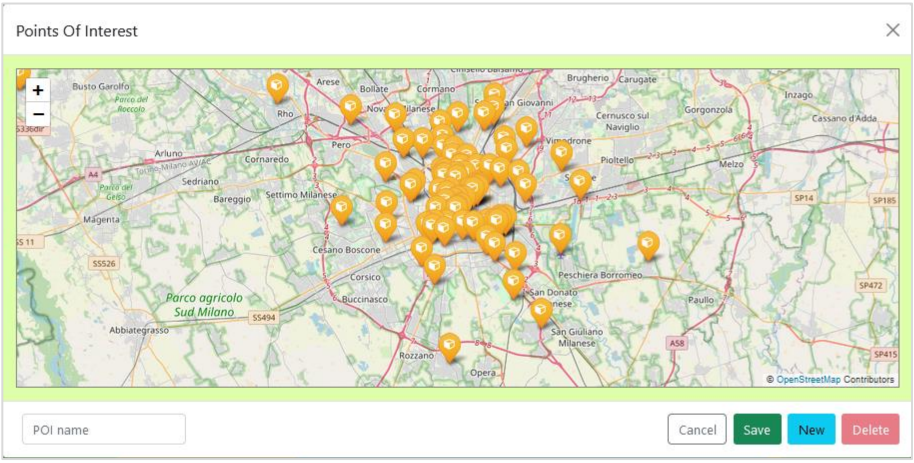


Figure 48 - MAGIS Interface: Points of Interest Tab

From this tab, it is possible to either associate the content to an existing POI or create a new POI. In the first case, the user can click on the existing POI and then select the option *Set* displayed in the popup of the location (Figure 49).

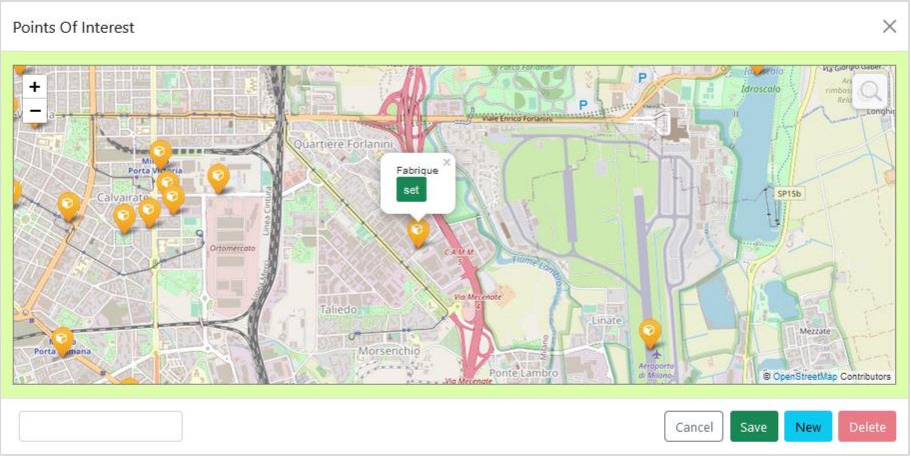
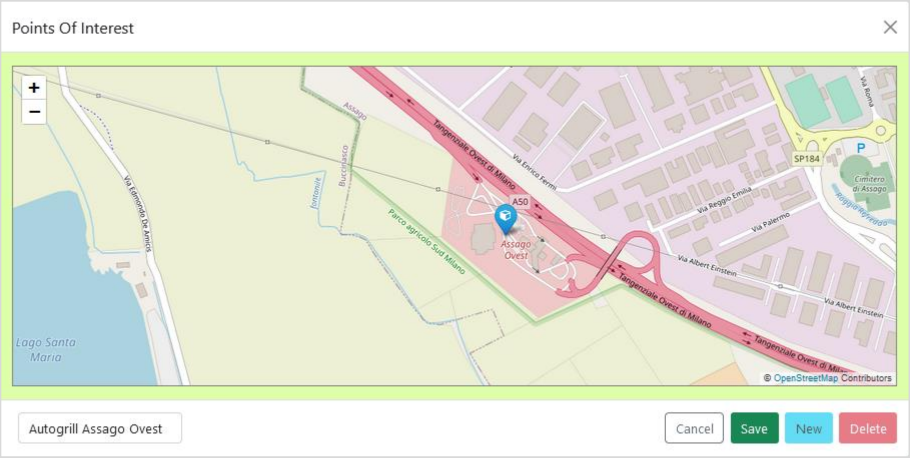


Figure 49 - MAGIS Interface: Setting an existing POI

To create a new point of interest, the name of the new POI needs to be entered in the proper space in the bottom-left corner of the tab and the option *New* creates a new marker, that is displayed in blue (Figure 50); this can be moved on the map to the desired position, even zooming the map as much as it is required. Then, the button *Save* confirms the creation of the new POI.

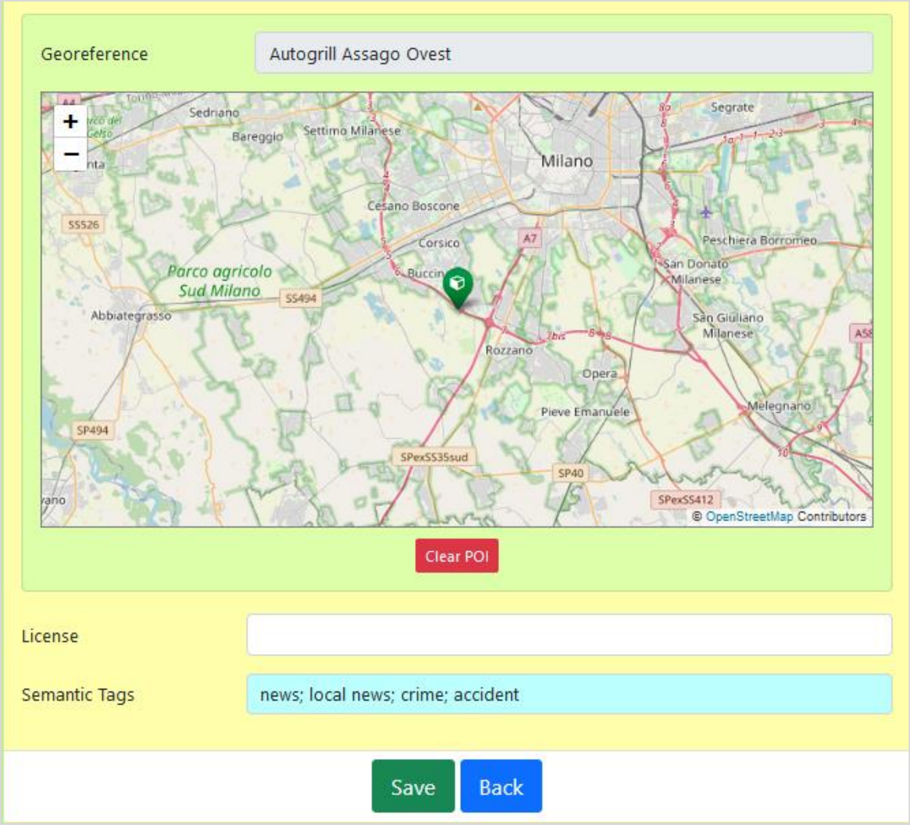
When the assignment of the content to the location is finalised, a green marker is displayed on the map indicating the position of the current piece of content. Above the map, the name of the POI is also shown (Figure 51). To save the new point of interest in the database, it is important to click the proper *Save* button. Clicking on the option *Back*, the user comes back to the content list.



Points Of Interest

Autogrill Assago Ovest

Figure 50 - MAGIS Interface: Creation of a new POI



Georeference: Autogrill Assago Ovest

License:

Semantic Tags: news; local news; crime; accident

Figure 51 - MAGIS Interface: Visualisation POI

When a new POI is entered, it is saved in the table POI of the database (previously shown in Figure 35). The identification code is automatically and incrementally created and saved in the primary key CodePOI. The attribute Name is filled with the name assigned by the user. The attributes Latitude and Longitude are determined by the geographical position of the marker on the map.

To complete the pieces of information related to the points of interest, the type of POI and the address of each location can be manually typed into the database. As anticipated in the general description of the database, the attribute TypePOI indicates the typology of point of interest, such as *Building, Street, Square, Park,* or a general *Area*, like a district or a neighbourhood. When the POI is not an extended surface but can be identified by a specific location, a proper Address may be entered as well; the name of a street is also accepted as an address, even though the exact position on the road may not be known. In Table 3, an example of POI is shown.

POI	
POICode	26
Name	Museo Nazionale della Scienza e della Tecnologia
Latitude	45.46169587
Longitude	9.17076015
Elevation	0
TypePOI	Building
Surface	0
Address	Via San Vittore, 21, 20123 Milano MI

Table 3 - Example POI definition

4.2.4 Data Classification

Once all data are collected in the database, the training of the ML classification algorithm can start. The model required for the prototype is a topic classifier, able to analyse a text and associate a label based on its meaning.

The classification algorithm selected for the project is the *Short Text Classifier* developed by *MonkeyLearn*. The short text used by the algorithm is made of a concatenation of title of the media and the description and tags extracted by Otero's library when the content item is parsed in it.

Being a supervised-learning algorithm, its application involves a first training phase where the model learns the classification procedure thanks to a manual assignment of the classes (in this case, the tags); this is done using a dedicated dataset (training set) which output is known. A second step involves the testing of the learnt procedure using a different dataset (test set), which output is still known and is compared to the output provided by the model.

Before proceeding with the classification process, some comparative tests have been run to compare the algorithm's setting options and define the most performant combination. As following step, the definitive training was carried out.

Chapter 5 is entirely dedicated to describing in detail the approach to select the most suitable algorithm and setting options. The Chapter reports the comparative tests and the final classification of uploaded content items.

4.2.5 Navigation Interface

This section aims to present the appearance and the usage of the navigation interface of MAGIS web prototype. The application can be found at the following web address:

http://www.mixmap.it/magis/home/view.php

The most functionalities of the web prototype are given after logging on the platform through the *"Login"* option on the top right of the interface. Having users logged on the website is necessary to classify user profiles, which is at the basis of the most common recommendation systems. Besides, it is useful to discriminate grants and privileges of the different user's types and it is also helpful to keep track of the provenance of new content items, with benefits in the data quality and validation. To log in the system, a *Username* and a *Password* are required, as shown in Figure 59.

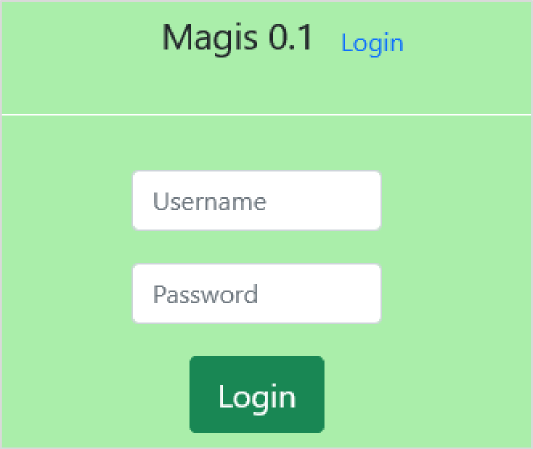


Figure 52 - Navigation Interface: Login page

Landing on the home page, with the option *View*, the user is shown the interface in Figure 60. The website shows the geographic map of Milan, on which green markers are displayed, which represent the points of interest where content items are stored. Clicking on each marker, a white window pops up showing the name of the point of interest, which can be either the name of a building or an attraction (e.g., *Museo Civico di Storia Naturale di Milano*) or the name of a road or an address (e.g., *Via Torino, Milano*). The user can zoom the map in and out through the buttons on the left side of the map ("+" and "-"), navigate with the cursor or enter a specific location through the lens icon on the top left of the map.

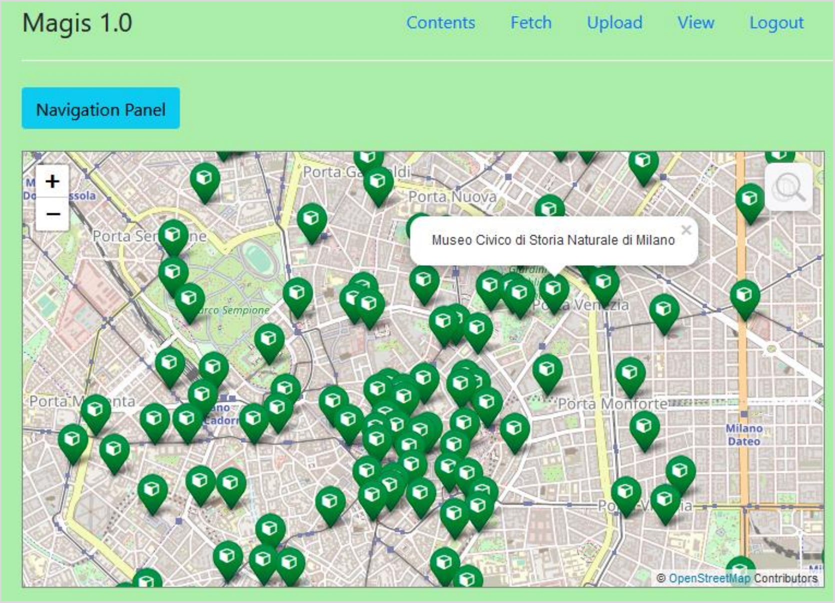
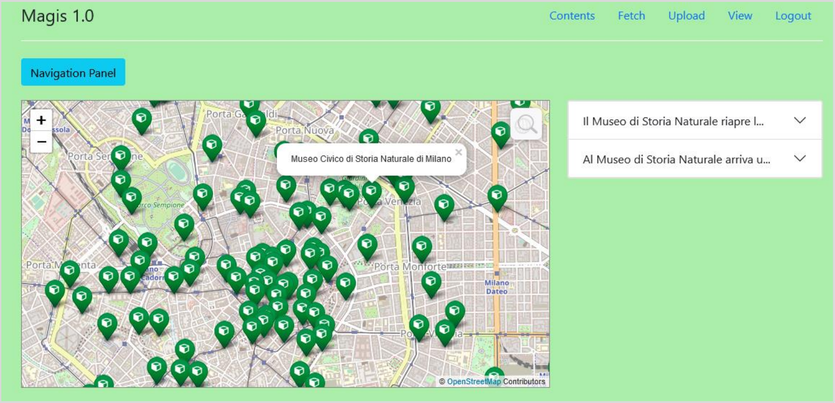


Figure 53 - Navigation Interface: Home

When the user selects one of the available points of interest, the content items connected to that POI are shown in form of list on the right side of the map (Figure 61).


Magis 1.0
Contents
Fetch
Upload
View
Logout
Navigation Panel
Museo Civico di Storia Naturale di Milano
© OpenStreetMap Contributors

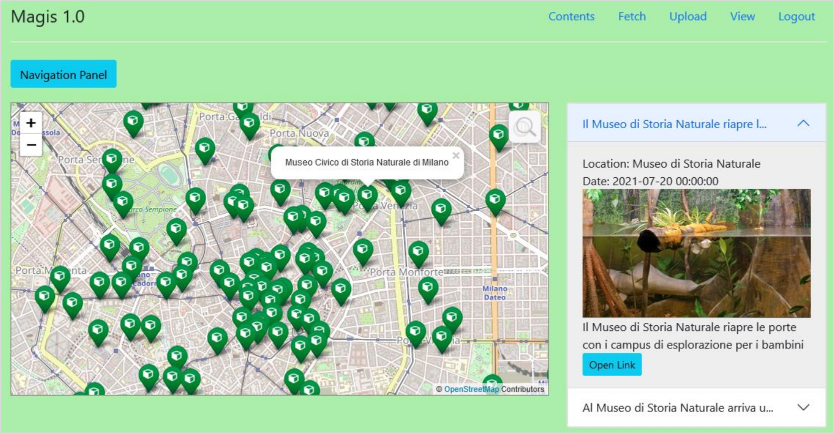
Il Museo di Storia Naturale riapre l...

Al Museo di Storia Naturale arriva u...

Figure 54 - Navigation Interface: Contents list

Clicking on one element on the list, that specific element is expanded, and some of its metadata are shown, including a picture, if available, representing the news (Figure 62). The button "*Open Link*" allows to open the original web page of the news article, so that the user can read the news connected to the point of interest.

The visualisation of the map as shown in the previous figures appears quite confusing, as tens of markers are shown together making it difficult to distinguish the elements of interest. To address this problem, the "*Navigation Panel*" on top of the screen is used to filter elements on the map. This is the key functionality of the interface, as it is the way to reduce information overload by selecting which subset of elements to display on the map.



Location: Museo di Storia Naturale
Date: 2021-07-20 00:00:00

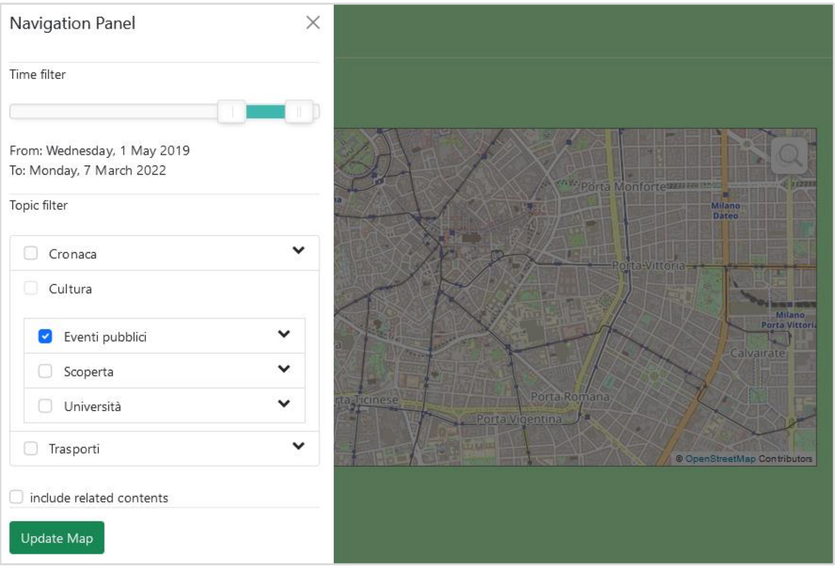
Il Museo di Storia Naturale riapre le porte con i campus di esplorazione per i bambini

Figure 55 - Navigation Interface: Content details

The navigation panel popping up on the left side of the page shows the list of filters that the user can select to discriminate the contents (Figure 63). The highest filtering level classifies contents according to the three main topic hierarchies shown in the domain-specific ontology in Figure 45, namely *Culture, Infrastructures & Transports* and *Local News* (in Italian *Cultura, Traporti, Cronaca*). These classes represent the three main topic layers discussed in the previous chapters, which aim is to segregate content items. Clicking on each class, a drop-down list appears showing the related tag hierarchy. The user can select one or more tags and through the button "*Update Map*" the map visualisation is refreshed, and the selected contents are displayed. As explained in the previous sections, the topic tree is dynamically generated starting from the ontology and the contents collected for each topic category.

Filters are not limited to the semantic dimension, but they also regard the temporal one; indeed, it is possible to select a time span of interest and the contents referred to that period are shown. The time filter considers as priority the StartDate and EndDate reported as metadata, which represent the time which the news in the web element is referred to. If this is not available, the publication date is taken into account.

An example of filtered visualisation in provided in Figure 64, which shows the subset of content elements related to public events (*Eventi pubblici*) in the period between 2019, May 1st and 2022, March 7.



Navigation Panel

Time filter

From: Wednesday, 1 May 2019
To: Monday, 7 March 2022

Topic filter

- □ Cronaca
- □ Cultura
  - ☑ Eventi pubblici
  - □ Scoperta
  - □ Università
- □ Trasporti

□ include related contents

Figure 56 - Navigation Interface: Navigation Panel

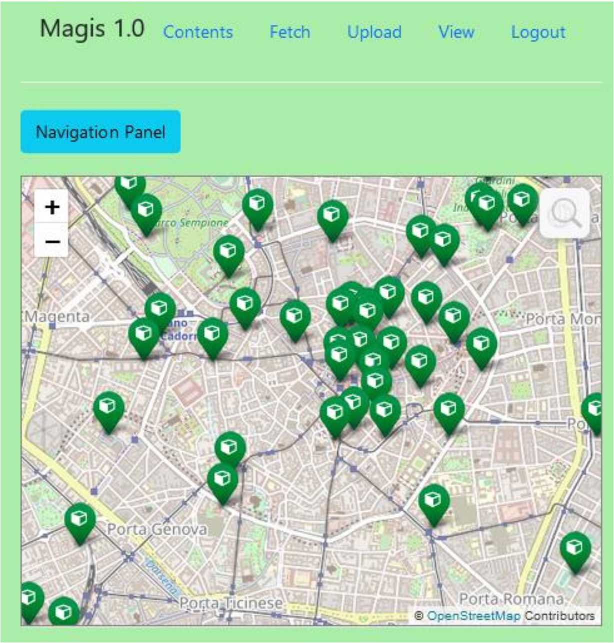


Figure 57 - Navigation Interface: Filtered Contents

While the navigation panel is used to select tags that are subclasses of the three main topic categories ("*Is a*" relationship), the option "*Include Related Contents*" can be flagged to display the contents belonging to related classes in the ontology, i.e., connected by a "*Related to*" type of relationship. This creates dynamism in the content navigation, as filters are not exclusively hierarchical.

5 MAGIS Approach for Data Classification

This Chapter aims to propose an approach to evaluate the most suitable algorithm and its setting options to classify content items for MAGIS applications. The objective is not to propose the best algorithm for the purpose, but to illustrate a structured process to select a performant classification algorithm for MAGIS implementations.

Overall, the approach is structured in the following phases, which are well described in the next paragraphs:

1. *Understand the model*: investigate how the algorithm works and what are its main setting options.
2. *Run comparative tests*: using some sample data, simulate the training and testing phases of the algorithm multiple times, changing some options to compare the output performances of the algorithm.
3. *Select the proper setting environment*: define some metrics to evaluate the performances in the different comparative tests and select the most performant combination of setting options.
4. *Set up the final model*: train and test the algorithm using the selected settings to make the algorithm finally functional.

As anticipated, the classification algorithm selected for the project is the *Short Text Classifier* developed by *Monkey Learn*. It is a ready-to-use online tool that can be accessed through a free account on the website. The free offer allows creating one custom model, with a maximum of 1.000 textual data, and 300 testing queries per month. The tool can be found on the following webpage:

https://monkeylearn.com/short-text-classification/

In the prototype, the training phase of the supervised algorithm has been performed using a 350-articles dataset coming from the three major topics (*Culture, Infrastructures & Transports, Local News*). About 120 additional links were kept as test dataset, used to test whether the model was trained enough to recognise tags in new texts.

Before applying the algorithm to the selected dataset, it is necessary to understand how the program works and which parameters are required to have a well-performant application. To understand the correct functioning of the algorithm and the right settings to be used for the purpose of this project, some comparative tests were carried out using a limited part of the collected dataset. In the following sections, a general description of the steps of the procedure is presented, followed by the comparative tests.

5.1 The model

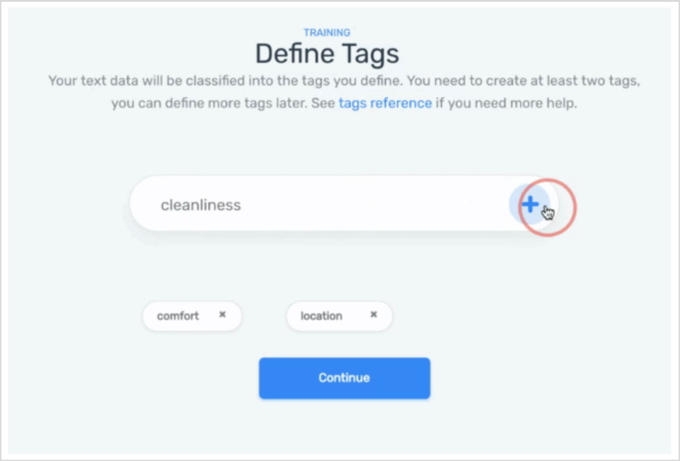
In this section, the general functioning of the short-text classification model is presented in its main phases and setups. Overall, when a new custom model is created on *MonkeyLearn* platform, data are required to be imported and the model needs to be set with the selected tags; then the training phase can start and eventually the model can be tested.

5.1.1 Data Import

As soon as a new custom model is created, MonkeyLearn platform requires to import the data that are going to be used in the training phase. The training dataset can be uploaded in csv or excel format and the attribute (column) to be analysed is asked to be selected. It is therefore necessary to export the table Metadata from the database into a csv/excel file, to be then imported into the classification platform. To facilitate the import in MonkeyLearn, it is advisable to discard the attributes that are not useful for the analysis (e.g., time attributes, URL etc.). Moreover, to get a more comprehensive text for the training, it is useful to concatenate the Title of the article with its Description and TagsFound, which are the text attributes that need to be analysed. This can be done either by concatenating the fields in the csv file opened in excel, or directly exploiting the attribute TrainingText in the database that joins the mentioned fields.

5.1.2 Setting the model

Once initialised the model by loading the input data, the first tags need to be defined (Figure 52). The scope of a classifier is defined by the list of possible tags. The advantage of this specific algorithm is that the user can define his/her classification tags, without simply relying on a predefined dictionary of labels. Tags should be distinct enough so that the model can avoid overlaps (or confusion). At the same time, the number of tags should be limited enough so that you have sufficient texts to assign to each tag [114].



TRAINING

Define Tags

Your text data will be classified into the tags you define. You need to create at least two tags, you can define more tags later. See tags reference if you need more help.

Figure 58 - Defining tags

For best results, it is advisable to keep the following in mind when defining tags for the first time [115].

- Limit the number of tags as much as possible. It is best to work with less than 10, at least initially. New tags can always be defined during the training phase.
- Make sure to have a sufficiently wide text sample per tag, with at least four texts. This is a critical aspect to accurately training a model. In general, with less than four texts per tag, it is difficult to get insight (and statistics) into how the classifier is performing until there are more texts. If the user is not sure to have enough, the tag can be created later.
- Avoid situations where one tag might be confused with another.

- Use one classification criteria per model, each classifier should have its explicit purpose.
- Start with broad tags.

One aspect to keep in mind is that the latest version of MonkeyLearn does not allow for the creation of hierarchies in custom classifiers. This implies that it is not possible to include subcategories or parent and child tags to build a hierarchy in the tag list. If a second level of tags needs to be built, the suggestion is to build a model for the highest level first, which will help build the necessary accuracy in the first training stages. After the first working model has been built, new classifiers can be trained, or new tags can be entered by including the name of the parent tag in the label (e.g., for various tags like *Movies* and *Music* under *Entertainment*, create *Entertainment_Movies* and *Entertainment_Music*). Anyways, in the case of MAGIS, the use of a tag ontology makes the need of a hierarchical classification algorithm unnecessary, as hierarchical levels can be directly defined in the ontology.

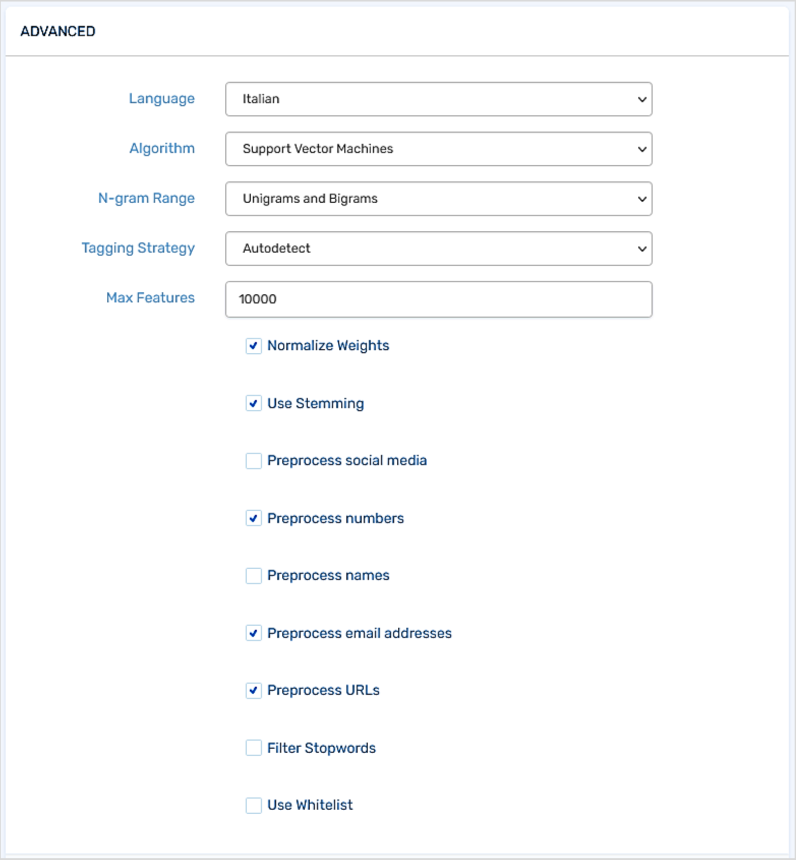
Preparing the model also includes adjusting the operating settings (Figure 53). It is important to set the following parameters:

- *Language* [116]. This setting should match the language in the text data. Selecting the correct language is important, as MonkeyLearn uses this information for the stemming and tokenization process and the default stopwords selection. In the CJ prototype, the language is Italian, as the articles taken into consideration come from Italian online journals.
- *Algorithm* [117]. This setting regulates the classifier algorithm that powers the model. Currently, there are two options: on one side, Multinomial Naïve Bayes (MNB) is a very simple and fast algorithm that has very good performance in most cases; on the other side, Support Vector Machine (SVM) is a more complex algorithm, slightly slower than Naive Bayes but delivers a higher accuracy in general. The latter is the algorithm set by default, but it is interesting to carry out a comparison of the performances of the two.

- *N-Gram Range* [118]. N-gram range sets if features to be used to characterize texts will be unigrams (i.e., single words), bigrams (i.e., terms compound by two words) or trigrams (i.e., terms compound by three words). With this setting, different combinations are possible: Unigrams; Unigrams and Bigrams (default); Unigrams, Bigrams and Trigrams; Bigrams; Bigrams and Trigrams; Trigrams. Changing this setting means taking into account different types of complex expressions. To start with, the default value may be sufficient.
- *Max Features* [119]. It sets the maximum number of features to be used to characterise texts in the training/classification process. This number affects how many computation resources are needed to train the model, as well as the amount of time needed to classify new texts. Including more features means that more computation time is necessary, and hopefully improved results are obtained, even if it is not always the case, as more features could decrease accuracy. The default value is set in *10,000* features which is a reasonable value for text mining applications.
- *Normalise Weights*. The setting for normalising weights informs the classifier whether it should take into account the number of texts for each tag when defining its probability. This is helpful when there are way more texts in some tags than others. If tags are imbalanced, with one tag having most texts, then you might consider normalising the weights to equal them out and see if it helps the classifier performance. The option is set by default, and it is kept active in the prototype.
- *Stemming* [120]. The stemming process transforms words into their root form, so inflected and derived words are grouped together. For example, the words *fishing*, *fished* or *fisher* are transformed to the root word *fish*. This is enabled by default if a particular language is selected; usually, it will help the classifier generalise and improve the classification, but it depends on the data and the purpose of the classification.
- *Preprocessing*. The different pre-processing options are used to replace the related elements with some keywords to exclude

them from the learning, signalling that there is a certain element, but that its specific meaning is not relevant. For example, when the *Preprocess Numbers* [121] parameter is selected, all numbers will be replaced by a special word __*number*__. This will allow the model to learn about number mentions in general rather than about specific number mentions. The same applies to URLs, email addresses, names and social media. In the prototype, names may be important to recognise topics, so that the preprocessing-names option remains unselected; the other options are not important for the classification scope, and they can be ticked.

- *Stopwords* [122]. Stopwords are words that usually do not contribute as classification features. Usually, stopwords are high-frequency words like articles, connectors, etc. Stopwords are usually selected from a predefined set of words that depends on the chosen language, but new ones may be added when needed. This feature is useful when some wrong keywords are used as features in the classifier (found by looking at the keyword cloud). In that case, they can be filtered by adding them to the list of stopwords. This is not an essential feature; thus, it can remain deactivated.
- *Whitelist* [123]. The whitelist parameter is basically a list of words that the model will *always* use as features. Including words in the whitelist will force classifiers to learn from those words regardless of the frequency with which they appear or how important they are for each category. This is useful when you know there is a word that should always be included in a certain category. If a word is included in the classifier's whitelist, a few texts will only need to be tagged for the classifier to be able to predict most of the texts containing that word correctly. In the specific case of the CJ prototype, if the words *theft, arrest, criminal,* etc. are found in the title of an article, it is quite sure that the text needs to be tagged under the label *Crime*, so those words can be included in the whitelist to facilitate the classification.



ADVANCED

Language: Italian

Algorithm: Support Vector Machines

N-gram Range: Unigrams and Bigrams

Tagging Strategy: Autodetect

Max Features: 10000

☑ Normalize Weights

☑ Use Stemming

□ Preprocess social media

☑ Preprocess numbers

□ Preprocess names

☑ Preprocess email addresses

☑ Preprocess URLs

□ Filter Stopwords

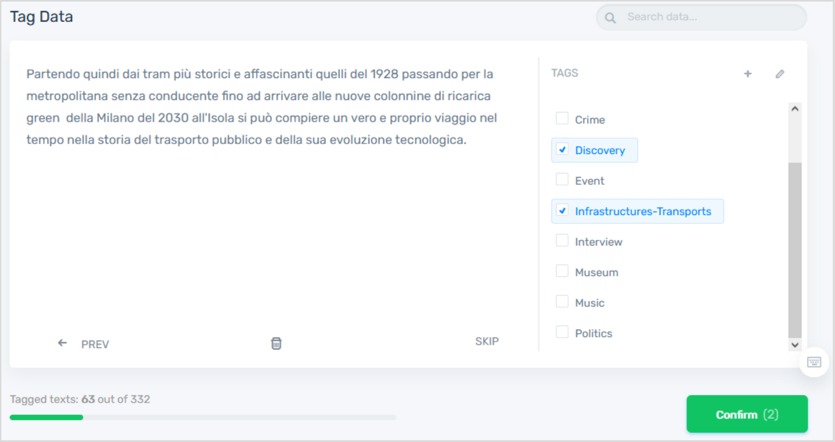
□ Use Whitelist

Figure 59 - Model Settings

5.1.3 Training the model

After the tag definition, the training is an interactive activity where the system randomly picks a sample data from the training data set and the user manually selects the tags that best describe that text (Figure 54). More than one tag can be selected if needed. If a description appears where the classification is not clear, it is possible to skip that text; having a relatively low number of texts for the training, it is better to discard the ones with a difficult or unclear classification. Tags can be modified in a second moment, by going back in the training and visualising past descriptions or editing previous choices.

Going on with the training, the ML algorithm starts recognising elements in the text and suggesting the tags to assign to successive descriptions. Clearly, at the beginning, the model makes many errors in the tagging, but the more the training goes on, the higher the precision in the identification of tags.



Partendo quindi dai tram più storici e affascinanti quelli del 1928 passando per la metropolitana senza conducente fino ad arrivare alle nuove colonnine di ricarica green della Milano del 2030 all'Isola si può compiere un vero e proprio viaggio nel tempo nella storia del trasporto pubblico e della sua evoluzione tecnologica.

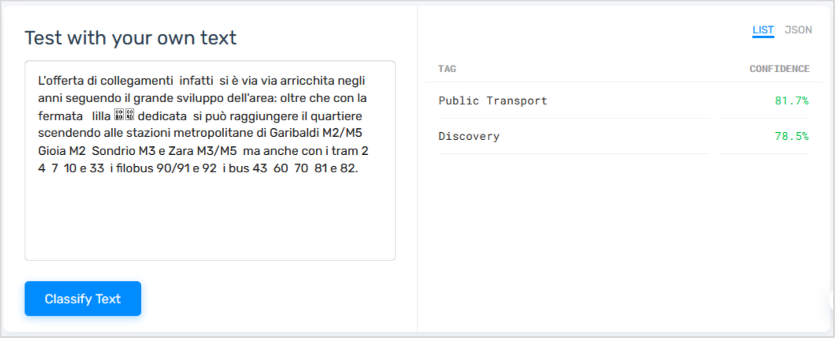
- ☐ Crime
- ☑ Discovery
- ☐ Event
- ☑ Infrastructures-Transports
- ☐ Interview
- ☐ Museum
- ☐ Music
- ☐ Politics

Tagged texts: **63** out of 332

Figure 60 - Training the model

5.1.4 Testing the model

Once the model is completed, it can be tested with the second part of the dataset (the test set). As it is shown in Figure 55, on the left-hand side of the web interface, the new description to be tested is entered, while on the right the tags are identified with the correspondent confidence level.



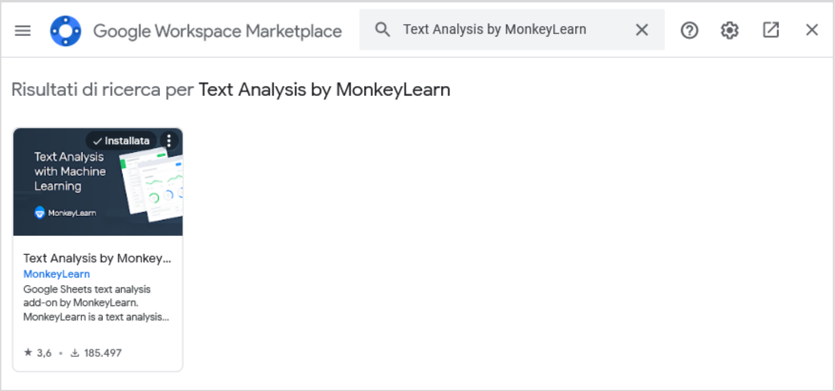
L'offerta di collegamenti infatti si è via via arricchita negli anni seguendo il grande sviluppo dell'area: oltre che con la fermata lilla ▯▯ dedicata si può raggiungere il quartiere scendendo alle stazioni metropolitane di Garibaldi M2/M5 Gioia M2 Sondrio M3 e Zara M3/M5 ma anche con i tram 2 4 7 10 e 33 i filobus 90/91 e 92 i bus 43 60 70 81 e 82.

TAG	CONFIDENCE
Public Transport	81.7%
Discovery	78.5%

Figure 61 - Testing the model

The classification model can also be tested automatically, with the support of a Google Sheets add-on. Once the list of texts is uploaded on a Google Drive Spreadsheet, the algorithm can be associated with that file as an external component (Figure 56).

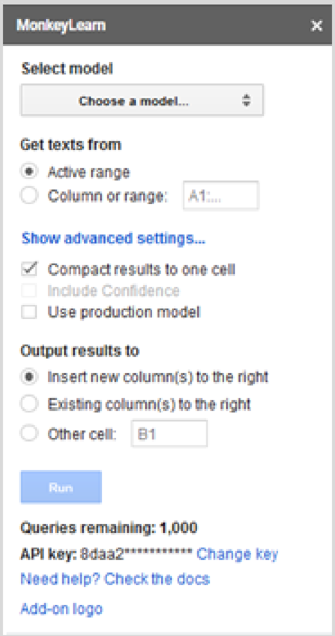
When the extension is installed and activated, the *API Key* present in MonkeyLearn account is required to be entered in the add-on interface as an authentication method. In this way, it is possible to retrieve the classification model created on the website and apply it to the list of texts in the spreadsheet (Figure 57). Based on the settings, the algorithm will process the texts in the sheet and return the ML output in form of tags and confidence levels.


Google Workspace Marketplace
Text Analysis by MonkeyLearn
Installata
Text Analysis with Machine Learning
MonkeyLearn

Risultati di ricerca per Text Analysis by MonkeyLearn

Text Analysis by Monkey...
MonkeyLearn
Google Sheets text analysis add-on by MonkeyLearn. MonkeyLearn is a text analysis...
★ 3,6 • 185.497

Figure 62 - Text Analysis by MonkeyLearn: Google Sheets add-on



Select model
Choose a model...

Get texts from
● Active range
○ Column or range: A1:...

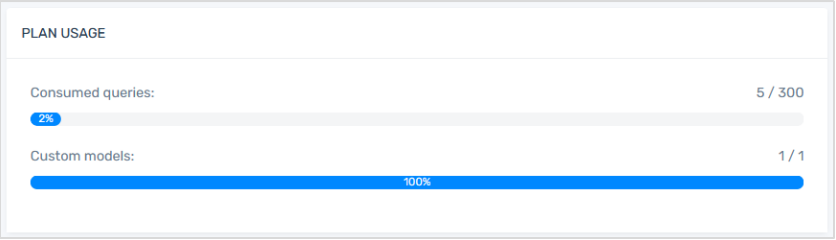
☑ Compact results to one cell
□ Include Confidence
□ Use production model

Output results to
● Insert new column(s) to the right
○ Existing column(s) to the right
○ Other cell: B1

Queries remaining: 1,000
API key: 8daa2************

Figure 63 - Text Analysis by MonkeyLearn: settings

The free account on the website allows for a limited number of test instances (300 per month), whether they are performed singularly on the website or massively with the add-on. In the section “My Account” of the website the *Plan Usage* can be checked (Figure 58). With the free offer, only one model can be created at a time; however, if the model is deleted, the counter goes back to zero and a new model can be defined. On the contrary, the counter of the 300 test queries is always progressive.



PLAN USAGE

Consumed queries: 5 / 300

2%

Custom models: 1 / 1

100%

Figure 64 - Plan Usage counter

5.2 Comparative tests

To understand the power of the classification tool and design the most suitable model for the case, some comparative tests have been carried out to confront the differences between specific setting options. In this project, some trials have been discussed with the following objectives:

1. Compare the performances of the Support Vector Machine and the Multinomial Naïve Bayes as classification algorithms (Trials 1 and 2).
2. Compare the performances of the classification algorithm using tags in English or Italian language (Trial 3).
3. Compare the performances of the classification algorithm analysing Unigrams and Bigrams or including Trigrams as well (Trial 4).
4. Compare the performances of the classification algorithm with and without the use of a proper whitelist (Trial 5).

All the tests have been carried out with the same dataset to have the most homogeneous testing environment possible. Six tags were chosen for the trials: *Concert, Museum, Discovery* and *University* for the *Culture* topic, *Transport* for the topic *Infrastructures & Transports* and *Crime* for the topic *Local News*. For each tag, twenty articles were selected as training dataset and five more for the testing phase, for a total of 120 training articles and 30 testing links[5].

For each trial, some statistics were generated about the training model and the testing effectiveness. Being able to understand the classifier statistics is a key part of improving the model's

[5] The list of texts used for the comparative tests are not reported here. The Reader may retrieve the articles from the complete list in the Attachment document through the number reported in the tables of the following paragraphs.

performance. MonkeyLearn offers two groups of statistics for the training: one group applies to the classifier overall, while the other refers to the single tags [124].

The *overall statistics* for the training phase are the following:

- *Accuracy*. Accuracy is the percentage of texts that were predicted with the correct tag. It is the total number of correct predictions divided by the total number of texts in the dataset. While providing a good indication, accuracy may not take into account large imbalances in the number of texts between tags, or other issues that might exist at a tag level.
- *F1 Score*. F1 Score is another measure of how well the classifier is doing its job, by combining both Precision and Recall for all the tags (see tag-level statistics). Unlike accuracy, it does a better job of accounting for any imbalances in the distribution of texts among tags.

Tag-specific statistics for the training phase are the following:

- *Precision*. Precision refers to the percentage of texts the classifier got right out of the total number of texts that is predicted for a given tag. If a tag has low precision, that means that texts from other tags are getting confused with the tag in question.
- *Recall*. Recall refers to the percentage of texts the classifier predicted for a given tag out of the total number of texts it should have predicted for that given tag. If a tag has a low recall, that means that texts from that tag are getting predicted for other tags.

Tag-level statistics are generated based on how well the algorithm classifies the data. This depends on four possible outcomes: true positives, true negatives, false positives and false negatives. A *true positive* (TP) is an outcome where the model correctly predicts the right tag. Similarly, a *true negative* (TN) is an outcome where the model correctly predicts the tags that don't apply. A *false positive* (FP) is an outcome where the model incorrectly predicts the right tag. And a *false negative* (FN) is an outcome where the model incorrectly predicts the tags that don't apply.

Precision and Recall are evaluated as:

$$Precision = \frac{True\ positives}{True\ positives + False\ Positives}$$

$$Recall = \frac{True\ positives}{True\ positives + False\ Negatives}$$

Considering the testing phase, instead, the main statistic provided by the algorithm is the confidence level. *Confidence* tells how sure one can be about the result. It is expressed as a percentage and represents how often the true percentage of the population who would pick an answer that lies within the confidence interval. The 95% confidence level means that one can be 95% certain; the 99% confidence level means one can be 99% certain [125]. It is important to keep in mind that the confidence level depends on the sample size: the wider the sample, the higher the confidence. The comparative tests explained in this section have been carried out with a limited number of articles, therefore it is reasonable to expect a quite low confidence level or anyway an imperfect performance of the algorithm. What is important anyway is to compare the confidence levels among the trials, rather than the absolute number *per se*, to evaluate different performances when some setting options are changed.

In the following paragraphs, the trials executed are described one by one.

5.2.1 Trial 1 – Support Vector Machine

The first trial is meant to evaluate the performance of the Support Vector Machine (SVM) algorithm. Table 4 summarises the settings for the model:

Trial 1 Settings

Algorithm	Support Vector Machine
Language	Italian
Tags Language	English
N-gram Range	Unigrams and Bigrams
Tagging Strategy	Autodetect
Max Features	10.000
Normalise weights	Yes
Use Stemming	Yes
Preprocess social media	No

Preprocess numbers	Yes
Preprocess names	No
Preprocess email addresses	Yes
Preprocess URLs	Yes
Filter Stopwords	No
Use Whitelist	No

Table 4 - Settings Model Trial 1

After setting the model and once completed the training for all 120 articles, the following statistics were obtained (Tables 5 and 6). Considering the overall statistics, the model got a satisfactory level of performance, as more than eight out of ten tags were correctly predicted. Considering the tag-level statistics, the algorithm resulted best-performant with the tags *Concert* and *University*, where very few false positives and false negatives were observed. For the tag *Transport*, all true negatives were correctly predicted, but some false negatives occurred. The tag *Discovery* recorded several false positives instead. To improve these performances, more training could be done on these tags.

Trial 1 – Overall Statistics

Texts	Accuracy	F1 score
120	85%	86%

Table 5 - Overall Statistics Trial 1

Trial 1 – Tag-Level Statistics

Tag	Texts	Precision	Recall	TP	TN	FP	FN
Concert	20	90%	90%	18	98	2	2
Crime	20	89%	80%	16	98	2	4
Discovery	20	61%	95%	19	88	12	1
Museum	20	94%	80%	16	99	1	4
Transport	20	100%	75%	15	100	0	5
University	20	95%	90%	18	99	1	2

Table 6 - Tag-level Statistics Trial 1

Concerning the testing of the model, the algorithm was able to recognise the correct label for 77% data used. The confidence level is very high for most tests, although errors in recognising the tags obtained high confidence as well. The highest confidence levels were obtained for the tags *Discovery*, *Museum* and *Concert*. The exact figures are reported in Table 7.

Trial 1 Testing Results

Category	Code	Recognised Tag	Confidence
Concert	39	Transport	1
	7	Concert	0,72
	17	Concert	1
	221	Concert	1
	343	Concert	0,964
Crime	160	Transport	0,838
	195	Crime	0,693
	104	Crime	1
	88	Transport	0,649
	209	Crime	1
Discovery	65	Discovery	0,844
	69	Discovery	0,796
	243	Discovery	0,649
	127	Discovery	1
	130	Discovery	1
Museum	151	Museum	1
	184	Museum	1
	13	Museum	1
	180	Museum	0,935
	186	Discovery	0,946
Transport	73	Transport	0,287
	226	Concert	0,43
	259	Transport	0,508
	227	Transport	1
	80	Crime	0,523
University	264	University	0,514
	268	Transport	0,622
	263	University	1
	344	University	1
	345	University	1

Table 7 - Testing Results Trial 1

5.2.2 Trial 2 – Multinomial Naïve Bayes

The second trial was meant to evaluate the performance of the algorithm Multinomial Naïve Bayes in comparison with the Support Vector Machine of Trial 1. Table 8 summarises the settings for the custom model:

Trial 2 Settings

Algorithm	Multinomial Naïve Bayes
Language	Italian
Tags Language	English
N-gram Range	Unigrams and Bigrams
Tagging Strategy	Autodetect
Max Features	10.000

Normalise weights	Yes
Use Stemming	Yes
Preprocess social media	No
Preprocess numbers	Yes
Preprocess names	No
Preprocess email addresses	Yes
Preprocess URLs	Yes
Filter Stopwords	No
Use Whitelist	No

Table 8 - Settings Model Trial 2

After setting the model and once completed the training for all the 120 articles, the following statistics were obtained (Tables 9 and 10). Considering the overall statistics, the model got a slightly less satisfactory level of performance compared to SVM algorithm, but still comparable, as about four out of five tags were correctly predicted. The same occurred with the tag-level statistics: again, the algorithm resulted best-performant with the tags *Concert* and *University*, where very few false positives and false negatives were observed. Like Trial 1, for the tags *Transport*, all true negatives were correctly predicted, but some false negatives occurred. The tag *Discovery* recorded several false positives instead, while all true positives were correctly predicted. the tags *Concert* and *Transport* got the worst results in terms of false negatives.

Trial 2 – Overall Statistics

Texts	Accuracy	F1 score
120	83%	84%

Table 9 - Overall Statistics Trial 2

Trial 2 – Tag-Level Statistics

Tag	Texts	Precision	Recall	TP	TN	FP	FN
Concert	20	90%	90%	18	98	2	2
Crime	20	93%	70%	14	99	1	6
Discovery	20	59%	100%	20	86	14	0
Museum	20	89%	80%	16	98	2	4
Transport	20	100%	70%	14	100	0	6
University	20	95%	90%	18	99	1	2

Table 10 - Tag-level Statistics Trial 2

Concerning the testing of the model, the algorithm was able to recognise the correct label for 70% of data used, lowering its performances compared to Trial 1. One more time, the confidence level is quite high; in some cases, it overcomes the one of Trial 1, in some other cases it is slightly worsened, even if still comparable. The highest confidence levels were obtained for the tags *Discovery*, *Museum* and *University*. The algorithm failed in recognising the tag *Transport* three times out of five. The exact figures are reported in Table 11, where a new column is added for the comparison of the confidence level with Trial 1. From this comparison, it is possible to notice that most texts for which the algorithm failed in recognising the correct tag are the same in Trial 1 and Trial 2; this rises the suspect that those texts might be somehow ambiguous.

Trial 2 Testing Results

Category	Code	Recognised Tag	Confidence	Comparison Trial 1
Concert	39	Transport	1	-
	7	Concert	0,476	0,72
	17	Concert	1	1
	221	Concert	1	1
	343	Concert	1	0,964
Crime	160	Transport	1	-
	195	Crime	0,855	0,693
	104	Crime	1	1
	88	Transport	0,527	-
	209	Crime	1	1
Discovery	65	Discovery	1	0,844
	69	Discovery	1	0,796
	243	Discovery	0,843	0,649
	127	Discovery	1	1
	130	Discovery	1	1
Museum	151	Museum	1	1
	184	Museum	1	1
	13	Museum	1	1
	180	Museum	1	0,935
	186	Discovery	1	-
Transport	73	University	0,552	0,287
	226	Concert	0	-
	259	Transport	0,235	0,508
	227	Transport	1	1
	80	Crime	0,508	-
University	264	Discovery	0,72	0,514
	268	Transport	0,235	-
	263	University	1	1
	344	University	1	1
	345	University	1	1

Table 11 - Testing Results Trial 2

5.2.3 Trial 3 – Italian Tags

The third trial was meant to evaluate the performances of the model when the tags are defined in Italian, which is the language for the input texts. The objective was to understand whether the correspondence between the two languages would improve the topic-recognition capacity of the algorithm. The model was run using the Support Vector Machine algorithm, as Trials 1 and 2 have proved to have better performances. Table 12 summarises the settings for the custom model:

Trial 3 Settings

Algorithm	Support Vector Machine
Language	Italian
Tags Language	Italian
N-gram Range	Unigrams and Bigrams
Tagging Strategy	Autodetect
Max Features	10.000
Normalise weights	Yes
Use Stemming	Yes
Preprocess social media	No
Preprocess numbers	Yes
Preprocess names	No
Preprocess email addresses	Yes
Preprocess URLs	Yes
Filter Stopwords	No
Use Whitelist	No

Table 12 - Settings Model Trial 3

The tags chosen for the trial were the followings: *Arte* (in English *Art*, corresponding to the tag *Museum*); *Cronaca* (corresponding to the tag *Crime*), *Musica* (corresponding to *Concert*); *Trasporti* (i.e., *Transports*); *Turismo* (i.e., *Tourism*, as the correspondent of *Discovery*); *Università* (i.e., *University*).

After setting the model and once completed the training for all the 120 articles, the following statistics were obtained (Tables 13 and 14). Considering the overall statistics, the model performance was very similar to the one of Trial 1, as more than eight out of ten tags were again correctly predicted. Concerning the tag-level statistics, the algorithm resulted best-performant with the tags *Musica, Cronaca* and *Trasporti*; good performances were also recorded for tags *Arte* and

Università. The tag *Turismo* was the less performant one, with eleven false positives recorded.

Trial 3 – Overall Statistics

Texts	Accuracy	F1 score
120	86%	86%

Table 13 - Overall Statistics Trial 3

Trial 3 – Tag-Level Statistics

Tag	Texts	Precision	Recall	TP	TN	FP	FN
Arte	20	89%	80%	16	98	2	4
Cronaca	20	94%	85%	17	99	1	3
Musica	20	90%	85%	17	98	2	3
Trasporti	20	100%	80%	16	100	0	4
Turismo	20	65%	100%	20	89	11	0
Università	20	94%	85%	17	99	1	0

Table 14 - Tag-level Statistics Trial 3

Concerning the testing of the model, the confidence levels in recognising tags were aligned to the ones in Trial 1: in some cases, Trial 3 performed slightly better, sometimes slightly worse, but it was still comparable to the first trial, without improving the performance in a significant way. The highest confidence levels were obtained for the tags *Musica*, *Turismo* and *Arte*. Similar to Trial 2, the texts that the algorithm in Trial 1 was not able to recognise, recorded a failure in Trial 3 as well; this supports the previous hypothesis of ambiguity in the texts. The exact figures are reported in Table 15. Overall, Trial 3 shows that setting tags in Italian language does not bring any significant improvement to the performance of the algorithm. This is probably because, although the algorithm can process texts in several languages, the language of tags is not particularly relevant[6].

Trial 3 Testing Results

Category	Code	Recognised Tag	Confidence	Comparison Trial 1
Musica	39	Trasporti	1	-
	7	Musica	0,762	0,72
	17	Musica	1	1
	221	Musica	0,859	1
	343	Musica	1	0,964
Cronaca	160	Trasporti	1	-

[6] In this dissertation, the code defining MonkeyLearn classification algorithm was not analysed, as it is out of scope. Therefore, how the tags are analysed in the algorithm is not known. Potentially, the algorithm could ignore the text of the tags, and considered them with an identification code; in this case, the language of the tags would be irrelevant.

	195	Cronaca	0,936	0,693
	104	Cronaca	1	1
	88	Trasporti	0,897	-
	209	Cronaca	1	1
Turismo	65	Turismo	1	0,844
	69	Turismo	0,763	0,796
	243	Turismo	0,844	0,649
	127	Turismo	1	1
	130	Turismo	1	1
Arte	151	Arte	0,915	1
	184	Arte	1	1
	13	Arte	1	1
	180	Arte	1	0,935
	186	Turismo	1	-
Trasporti	73	Trasporti	0,103	0,287
	226	Musica	0,649	-
	259	Trasporti	0,738	0,508
	227	Trasporti	0,822	1
	80	Cronaca	0,751	-
Università	264	Università	0,241	0,514
	268	Trasporti	0,909	-
	263	Università	1	1
	344	Università	1	1
	345	Università	1	1

Table 15 - Testing Results Trial 2

5.2.4 Trial 4 – Trigrams

The fourth trial was meant to evaluate the performance of the classification algorithm when the model is set to analyse Trigrams besides Unigrams and Bigrams. The objective was to establish whether analysing groups of three words as well would make the model more performant than analysing only single words and groups of two. In this case, two trials were carried out:

- Trial 4a was carried out analysing Unigrams and Bigrams.
- Trial 4b was carried out analysing Unigrams, Bigrams and Trigrams as well.

In both cases, the Support Vector Machine algorithm was used and a list of Italian stopwords was included to reduce the number of words combinations that the algorithm is supposed to analyse. The stopwords, reported in Table 16 below, have been selected from the list created by the software architect Alireza Savand and released on the development platform GitHub [126].

Trial 4 - Stopwords
a, abbia, abbiamo, abbiano, abbiate, ad, adesso, agl, agli, ai, al, all, alla, alle, allo, allora, altre, altri, altro, anche, ancora, avemmo, avendo, avere, avesse, avessero, avessi, avessimo, aveste, avesti, avete, aveva, avevamo, avevano, avevate, avevi, avevo, avrai, avranno, avrebbe, avrebbero, avrei, avremmo, avremo, avreste, avresti, avrete, avrà, avrò, avuta, avute, avuti, avuto, c, che, chi, ci, coi, col, come, con, contro, cui, da, dagl, dagli, dai, dal, dall, dalla, dalle, dallo, degl, degli, dei, del, dell, della, delle, dello, dentro, di, dov, dove, e, ebbe, ebbero, ebbi, ecco, ed, era, erano, eravamo, eravate, eri, ero, essendo, faccia, facciamo, facciano, facciate, faccio, facemmo, facendo, facesse, facessero, facessi, facessimo, faceste, facesti, faceva, facevamo, facevano, facevate, facevi, facevo, fai, fanno, farai, faranno, fare, farebbe, farebbero, farei, faremmo, faremo, fareste, faresti, farete, farà, farò, fece, fecero, feci, fino, fosse, fossero, fossi, fossimo, foste, fosti, fra, fu, fui, fummo, furono, giù, gli, ha, hai, hanno, ho, i, il, in, io, l, la, le, lei, li, lo, loro, lui, ma, me, mi, mia, mie, miei, mio, ne, negl, negli, nei, nel, nell, nella, nelle, nello, no, noi, non, nostra, nostre, nostri, nostro, o, per, perché, però, più, pochi, poco, qua, quale, quanta, quante, quanti, quanto, quasi, quella, quelle, quelli, quello, questa, queste, questi, questo, qui, quindi, sarai, saranno, sarebbe, sarebbero, sarei, saremmo, saremo, sareste, saresti, sarete, sarà, sarò, se, sei, senza, si, sia, siamo, siano, siate, siete, sono, sopra, sotto, sta, stai, stando, stanno, starai, staranno, stare, starebbe, starebbero, starei, staremmo, staremo, stareste, staresti, starete, starà, starò, stava, stavamo, stavano, stavate, stavi, stavo, stemmo, stesse, stessero, stessi, stessimo, stesso, steste, stesti, stette, stettero, stetti, stia, stiamo, stiano, stiate, sto, su, sua, sue, sugl, sugli, sui, sul, sull, sulla, sulle, sullo, suo, suoi, te, ti, tra, tu, tua, tue, tuo, tuoi, tutti, tutto, un, una, uno, vai, vi, voi, vostra, vostre, vostri, vostro, è

Table 16 - Trial 4 Stopwords

Table 17 summarises the settings for the custom model in both cases:

	Trial 4a Settings	Trial 4b Settings
Algorithm	Support Vector Machine	Support Vector Machine
Language	Italian	Italian
Tags Language	English	English
N-gram Range	Unigrams and Bigrams	Unigrams, Bigrams, Trigrams
Tagging Strategy	Autodetect	Autodetect
Max Features	10.000	10.000
Normalise weights	Yes	Yes
Use Stemming	Yes	Yes
Preprocess social media	No	No
Preprocess numbers	Yes	Yes
Preprocess names	No	No
Preprocess email addresses	Yes	Yes
Preprocess URLs	Yes	Yes
Filter Stopwords	Yes	Yes
Use Whitelist	No	No

Table 17 - Settings Model Trial 4

After setting the model and once completing the training for all the 120 articles for both cases, the following statistics were obtained (Tables 18, 19 and 20). Considering the overall statistics, both models are quite aligned to Trial 1, as more than eight out of ten tags were correctly predicted; moreover, the model of Trial 4a performed slightly better than Trial 4b, as the analysis of trigrams led to almost 90% rate of success in recognising tags. A similar scenario occurred in the tag-level statistics: in Trial 4a, the algorithm resulted best-performant with the tag *Museum*, where only two false negatives were observed, and a similar performance regarded the tag *Concert*. Categories *Transports, University* and *Discovery* recorded some errors in recognising tags, instead. In Trial 4b, some mistakes occurred, especially for tags *Discovery* and *University*, that recorded some false positives, and for tags *Museum* and *Transports* concerning the false negatives. Overall, the performances of the algorithm are very similar the one to the other, and still comparable to the ones of Trial 1.

Trial 4 – Overall Statistics

	Texts	Accuracy	F1 score
Trial 4a	120	88%	88%
Trial 4b	120	85%	85%

Table 18 - Overall Statistics Trial 4

Trial 4a – Tag-Level Statistics

Tag	Texts	Precision	Recall	TP	TN	FP	FN
Concert	20	95%	90%	18	99	1	2
Crime	20	90%	90%	18	98	2	2
Discovery	20	77%	85%	17	95	5	3
Museum	20	100%	90%	18	100	0	2
Transport	20	83%	75%	15	97	3	5
University	20	83%	95%	19	96	4	1

Table 19 - Tag-level Statistics Trial 4a

Trial 4b – Tag-Level Statistics

Tag	Texts	Precision	Recall	TP	TN	FP	FN
Concert	20	90%	85%	17	98	2	3
Crime	20	95%	90%	18	99	1	2
Discovery	20	69%	90%	18	92	8	2
Museum	20	94%	80%	16	99	1	4
Transport	20	89%	80%	16	98	2	4
University	20	81%	85%	17	96	4	3

Table 20 - Tag-level Statistics Trial 4b

Concerning the testing of the model, the confidence levels in recognising tags were aligned to the ones in Trial 1: in some cases, Trial 4 performed slightly better, sometimes slightly worse, but it was still comparable to the first trial. The best performance regards the tag *University*, where no errors were made in recognising the correct topic. Good results also regard tags *Museum* and *Concert*, which obtained very confidence levels, especially in Trial 4a. Again, however, several mistakes occurred in labelling the texts: compared to the previous trials, the algorithm improved in recognising topics *University* and *Transport*, while worsening its performance on tag *Discovery*. Overall, the performance of Trial 4a and Trial 4b was comparable to Trial 1, as the percentage of success was respectively 77% and 80%. The exact figures are reported in Table 21.

Trial 4 Testing Results

Category	Code	Recognised Tag Trial 4a	Confidence Trial 4a	Recognised Tag Trial 4b	Confidence Trial 4b
Concert	39	Transport	1	Transport	1
	7	Concert	1	Concert	0,648
	17	Concert	1	Concert	1
	221	Concert	1	Concert	0,589
	343	Concert	1	Concert	1
Crime	160	Transport	0,803	Transport	0,866
	195	Crime	0,712	Crime	0,243
	104	Crime	1	Crime	1
	88	Transport	0,435	Transport	0,729
	209	Crime	1	Crime	1
Discovery	65	Transport	0,625	Discovery	0,793
	69	Discovery	1	Discovery	0,764
	243	Crime	0,232	Crime	0,91
	127	Discovery	1	Discovery	1
	130	Discovery	1	Discovery	1
Museum	151	Museum	1	Museum	0,582
	184	Museum	1	Museum	1
	13	Museum	1	Museum	1
	180	Museum	0,778	Museum	0,926
	186	Discovery	1	Discovery	1
Transport	73	Transport	1	Transport	1
	226	Concert	1	Concert	0,648
	259	Concert	1	Concert	1
	227	Concert	1	Concert	0,589
	80	Concert	1	Concert	1
University	264	University	0,799	University	0,794
	268	University	0,908	University	0,54
	263	University	1	University	1
	344	University	1	University	1
	345	University	1	University	1

Table 21 - Testing Results Trial 4

5.2.5 Trial 5 – Whitelist

The fifth and last trial was meant to evaluate the performance of the classification algorithm when a whitelist is entered in the settings. As anticipated, the whitelist parameter is a list of words that the model will *always* use as features. It is useful when we know there is a word that should always be included in a certain category. The objective was to establish whether adding some keywords in the whitelist would improve the ability of the model in recognising topics and labelling texts. The model was still set with the Support Vector Machine algorithm. The N-gram Range was set at analysing unigrams and bigrams only, to compare the new model with Trial 1 taken as a basic scenario. Table 22 summarises the settings for the custom model:

Trial 5 Settings

Algorithm	Support vector Machine
Language	Italian
Tags Language	English
N-gram Range	Unigrams and Bigrams
Tagging Strategy	Autodetect
Max Features	10.000
Normalise weights	Yes
Use Stemming	Yes
Preprocess social media	No
Preprocess numbers	Yes
Preprocess names	No
Preprocess email addresses	Yes
Preprocess URLs	Yes
Filter Stopwords	No
Use Whitelist	Yes

Table 22 - Settings Model Trial 5

When the option is activated, if a word is included in the classifier's whitelist, a few texts will only need to be tagged for the classifier to be able to predict most of the texts containing that word correctly. This means that the list should include non-ambiguous words that for sure indicate that a text belongs to a certain topic. In this case, the following whitelist was created:

- The keywords *Mostra* and *Museo* (in English *Exposition* and *Museum*) were added for the *Museum* tag.
- The keywords *Concerto*, *Musica* and *Spettacoli* (in English *Concert*, *Music* and *Shows*) were added for the *Concert* tag.

- The keywords *Bus*, *ATM* (i.e., identifying the public transport agency of Milan), *Trasporto Pubblico* and *Stazione* (i.e., *Public Transport* and *Station*) were added for the *Transport* tag.
- The keywords *Università, Politecnico, Bocconi*, *Laurea* (i.e., *Degree*) and *Master* were added for the *University* tag.
- The keywords *Accoltellato, Furto, Arrestato, Ladro, Aggredito, Rapina, Sequestro, Scassinare* (in English *Stabbed, Theft, Arrested, Thief, Assaulted, Requisition, to Force*) were added for *Crime* tag.
- The keywords *Viaggio, Scoprire, Tour, Quartiere* (in English *Journey, to Discover, Tour, Neighbourhood*) were added for the Discovery tag.

These keywords were added in such way that all the articles selected for testing the algorithm would include at least one word from the "whitelist".

After setting the model and completed the training for all the 120 articles, the following statistics were obtained (Tables 23 and 24). Considering the overall statistics, the model improved compared to Trial 1, where no whitelist was used, as almost nine out of ten tags were correctly predicted during the training. A similar improvement occurred with the tag-level statistics: the algorithm managed to correctly recognise all the true positives for tag Discovery, even if nine false positives were detected as well. Good performances also concerned the other tags.

Trial 5 – Overall Statistics

Texts	Accuracy	F1 score
120	88%	88%

Table 23 - Overall Statistics Trial 5

Trial 5 – Tag-Level Statistics

Tag	Texts	Precision	Recall	TP	TN	FP	FN
Concert	20	90%	90%	18	98	2	2
Crime	20	94%	85%	17	99	1	3
Discovery	20	69%	100%	20	91	9	0
Museum	20	90%	90%	18	98	2	2
Transport	20	100%	70%	14	100	0	6
University	20	95%	90%	18	99	1	2

Table 24 - Tag-level Statistics Trial 5

Concerning the testing of the model, the confidence levels in recognising tags were aligned to the ones in the previous trials: in some cases, Trial 5 performed slightly better than Trial 1, sometimes slightly worse, but it was still comparable to the first trial. The best performance regarded the tag *Discovery*, where the topic of all texts was correctly predicted with high confidence. One more time, the algorithm made mistakes in recognising the same texts that were critical in Trial 1 as well, despite all the texts used for testing the model included at least one word from the whitelist. This supports once again the hypothesis of ambiguity in the texts. For these cases, manual tagging is suggested. With this trial, the evidence is that the whitelist does not bring any particular contribution to the correct identification of the topics when new texts are analysed. The exact figures are reported in Table 25.

Trial 5 Testing Results

Category	Code	Recognised Tag	Confidence	Comparison Trial 1
Concert	39	Transport	1	-
	7	Concert	1	0,72
	17	Concert	0,675	1
	221	Concert	0,777	1
	343	Concert	1	0,964
Crime	160	Transport	1	-
	195	Crime	0,658	0,693
	104	Crime	1	1
	88	Transport	1	-
	209	Crime	1	1
Discovery	65	Discovery	1	0,844
	69	Discovery	0,558	0,796
	243	Discovery	0,748	0,649
	127	Discovery	1	1
	130	Discovery	1	1
Museum	151	Museum	0,493	1
	184	Museum	1	1
	13	Museum	1	1
	180	Museum	1	0,935
	186	Discovery	1	-
Transport	73	Transport	0,205	0,287
	226	Concert	0,36	-
	259	Transport	0,397	0,508
	227	Transport	1	1
	80	Crime	0,411	-
University	264	University	0,544	0,514
	268	Transport	0,834	-
	263	University	1	1
	344	University	1	1
	345	University	1	1

Table 25 - Testing Results Trial 5

5.3 Evaluation and final model

To select the most performant settings for the classification algorithm to use as final model, the different comparative tests were evaluated according to two main metrics, that are based on the performances of the algorithm during the testing phase.

1. *Percentage of correct tag predictions*. It is evaluated as the ratio between the number of times the algorithm has predated the correct tag for each testing element and the total number of tested articles.

$$\%\, Correct\ predictions = \frac{Number\ of\ correct\ predictions}{Number\ of\ total\ predictions}$$

2. Average confidence of correct predictions. It is evaluated as the average confidence level recorded on correct tag predictions.

$$Average\ confidence = \frac{\sum Confidence\ of\ correct\ predictions}{Number\ of\ total\ predictions}$$

Applying those metrics to all trials, an average has been applied to obtain a single comparable number scoring the value of each trial. The two metrics were assigned equal weights (0,5 each), as they are equally important.

Therefore, the scoring formula used to evaluate the overall value of each trial is the following:

$$Score = 0{,}5 * \%\, Correct\ predictions + \\ 0{,}5 * Average\ confidence$$

Table 26 shows the evaluation of the two metrics on the different comparative trials, including the final score of each model.

Comparative Metrics

Trial	% Correct predictions	Average confidence	Score
Tral 1	0,767	0,866	0,816
Trial 2	0,7	0,924	0,812
Trial 3	0,767	0,869	0,818
Trial 4a	0,767	0,925	0,845
Trial 4b	0,8	0,849	0,724
Trial 5	0,767	0,828	0,797

Table 26 - Comparative metrics and overall scoring

As shown in the table, the different trials present similar performances on the two metrics. Trial 2 is the worst-performant in terms of predicting the correct tag, while Trial 4b is the best-performant, even though the other trials are very close to it. The highest average confidence was recorded for Trial 2 and Trial 4a, but all other trials present very high levels as well.

Considering the total scoring, that is the average between the two metrics, Trial 4a ends up being the best-performant test, even if the other trials are not far from it. It is not surprising, as both metrics presented quite high values. The evidence of this evaluation proves that to the extent of the comparative tests run on the selected 150 content items, the best-performant algorithm to be applied is the *Support Vector Machine* algorithm, which objective is to analyse *unigrams and bigrams* to predict *English tags*, by *filtering stopwords*. In other words, the introduction of a stopwords filter adds value to the classification model, while other parameters such as the analysis of trigrams or the introduction of a whitelist are not so relevant from the performance point of view.

Even though Trials 3 to 5 provide better results in the training statistics, Trial 4a proved to be the best attempt in terms of testing. What is important in this prototype is the ability to correctly tag the new texts uploaded by users in the system, rather than suggesting the right tags during the training, as the training is a background activity carried out *before* the release of the prototype. In other words, what matters is that the algorithm works after the training of the model is completed. For these reasons, the model used for comparative **Trial 4a** was selected as the final classification algorithm.

Using the same settings described in Table 17, the definitive model was built using a wider dataset compared to the comparative models. Starting from the dataset described in paragraph "*Data sources and Data Collection*", 12 tag categories were used to classify the data: *Accident, Cinema, Commemoration, Concert, Crime, Discovery, Museum, New opening, Politics, Theatre, Transport, University*. For each tag, 40 articles were selected as data base, of which 30 was used as training set and 10 were employed as *test set* to test the correct functioning of the model. The only exception is the category *Commemoration*, for which 20 articles were used as training set instead of thirty. Hereafter, the key results are reported.

Tables 27 and 28 report the usual statistics for the training phase. The overall statistics show that more than 8 articles out of 10 were correctly labelled while training the model, which means that the algorithm was able to learn fast how to recognise the topic of analysed texts. Tag-specific statistics instead compare the training performances of the different tags. The best performant topics are *Museum* and *University*, for which very few false positives and false negatives were recorded, leading both precision and recall close to 100%. The less performant are instead the tags *Politics*, with less than 70% precision, and New Opening, with just 53% recall. The correspondent high number of false positives and negatives is probably due to a certain ambiguity in the specific texts, or to the fact that the topics could be easily merged into other topics.

Definitive Model – Overall Statistics

Texts	Accuracy	F1 score
350	82%	81%

Table 27 - Overall Statistics Definitive Model

Definitive Model – Tag-Level Statistics

Tag	Texts	Precision	Recall	TP	TN	FP	FN
Accident	30	71%	90%	27	309	11	3
Cinema	30	87%	87%	26	316	4	4
Commemoration	20	74%	70%	14	325	5	6
Concert	30	81%	83%	25	314	6	5
Crime	30	83%	63%	19	316	4	11
Discovery	30	76%	93%	28	311	9	2
Museum	30	94%	97%	29	318	2	1
New opening	30	76%	53%	16	315	5	14
Politics	30	65%	67%	20	309	11	10
Theatre	30	85%	77%	23	316	4	7
Transport	30	73%	73%	22	312	8	8
University	30	94%	100%	30	318	2	0

Table 28 - Tag-level Statistics Definitive Model

Table 29 is instead dedicated to the results of the testing phase. The classification algorithm was best performant in *Accident* category, where all articles were correctly labelled, followed by categories *Cinema* and *New opening*, where one single mistake was made. Crime was instead the worst performant category, as several articles were labelled with a different tag. In particular, four articles were tagged as *Accident*, meaning that probably the two topics are quite similar and that there may be some keywords representing crimes that can be assimilated to accidents. The algorithm also failed in recognising four museum-related texts, labelling them as *Commemoration*; again, this means that the two categories may have similar keywords.

Definitive Model Testing Results

Category	Code	Recognised Tag	Confidence
Accident	359	Accident	0,815
	360	Accident	0,718
	361	Accident	1
	362	Accident	1
	363	Accident	1
	364	Accident	0,871
	365	Accident	0,718
	366	Accident	1
	367	Accident	1
	368	Accident	0,815
Cinema	389	Cinema	0,707
	390	Cinema	1
	391	Cinema	0,899
	392	New opening	0,732
	393	Cinema	0,995
	394	Cinema	1
	395	Cinema	1
	396	Cinema	1
	397	Cinema	0,924
	398	Cinema	0,872
Commemoration	493	Commemoration	0,819
	494	Commemoration	1
	495	Commemoration	1
	496	Commemoration	0,626
	497	Crime	0,806
	498	Commemoration	0,486
	499	Commemoration	0,332
	500	Politics	0,498
	501	Commemoration	0,827
	502	Commemoration	1
Concert	404	Concert	0,997
	405	Concert	1
	406	Concert	1
	407	Concert	0,821
	408	Concert	0,851
	409	Concert	1
	410	Concert	0,958
	411	Concert	0,984
	412	New opening	0,833
	413	Theatre	1
Crime	194	Crime	0,900
	195	Accident	0,447
	199	Commemoration	0,759
	200	Crime	0,295
	202	Crime	0,757
	208	Crime	0,731
	209	Accident	0,624
	211	Politics	0,615
	214	Accident	0,636

	216	Accident	0,548
Discovery	141	Discovery	0,810
	142	Discovery	1
	143	Discovery	0,921
	144	Discovery	0,951
	145	Discovery	1
	146	Discovery	1
	173	Commemoration	0,337
	244	New opening	0,651
	313	Discovery	0,633
	314	Discovery	0,817
Museum	124	Museum	0,643
	147	Museum	0,868
	148	Museum	0,634
	149	Museum	0,967
	150	Commemoration	0,944
	151	Commemoration	0,297
	152	Commemoration	0,607
	153	Commemoration	0,720
	155	Museum	0,926
	156	Museum	0,718
New opening	448	New opening	0,905
	449	New opening	0,856
	451	New opening	1
	453	Concert	0,700
	454	New opening	0,963
	455	New opening	1
	456	New opening	0,858
	457	New opening	0,753
	458	New opening	1
	459	New opening	1
Politics	476	Politics	0,731
	477	Politics	0,803
	478	Politics	0,871
	479	Politics	0,871
	480	Politics	1
	481	Commemoration	0,216
	482	Politics	0,268
	483	New opening	0,155
	484	Concert	0,443
	485	Politics	0,303
Theatre	524	Theatre	1
	525	Theatre	0,833
	526	Concert	0,292
	527	Theatre	0,947
	528	Politics	0,713
	529	Concert	0,830
	530	Theatre	0,834
	531	Theatre	0,851
	532	Theatre	0,700
	533	Theatre	0,361

Transports	238	Concert	0,256
	250	Transports	0,901
	252	Transports	0,120
	259	Transports	0,338
	261	Transports	0,950
	414	Transports	0,441
	415	Transports	0,472
	416	Politics	0,172
	417	Transports	0,969
	418	Politics	0,294
University	425	University	0,344
	426	University	0,911
	427	Accident	0,214
	428	University	0,900
	429	Accident	0,296
	430	University	0,915
	431	University	0,914
	432	New opening	0,335
	433	University	0,909
	434	University	1

Table 29 - Testing Results Definitive Model

Table 30 summarises the performance of the classification algorithm. For each topic category, the number of correctly predicted texts is represented on a total of ten texts per tag, together with the average confidence. In this condensed view, it is easier to compare performances among topics. It is interesting to notice that the category *Crime*, which as mentioned above, had a lower performance in recognising the correct tag, also recorded a quite low average confidence level, which is a sign that the algorithm would require further training to better learn to identify that topic. The same occurred with category Transport, which recorded the lowest average confidence.

Overall Results

	Correct	**Average confidence**
Accident	10	0,894
Cinema	9	0,933
Commemoration	8	0,933
Concert	8	0,951
Crime	4	0,671
Discovery	8	0,892
Museum	6	0,793
New opening	9	0,926
Politics	7	0,692
Theatre	7	0,789

Transport	7	0,599
University	7	0,842
TOTAL	**90**	**0,827**

Table 30 - Overall Results Definitive Model

Overall, the classification model can be considered as satisfying, as 75% of articles were correctly predicted with an average confidence level above 80%. Again, it is important to remind that the testing goal was to propose an approach to the proper algorithm selection, rather than identify the best algorithm to be used.

6 Concluding Remarks

This Thesis has presented the MAGIS framework (*Multimedia Adaptive Geographic Information System*) based on a data model to associate semantic content to cartographic elements. Multimedia management, temporal references and adaptability are the key features. This conceptual structure can be adaptively instantiated by specific applications in different domain contexts.

In fact, several companies have developed different applications, some focusing on the geographic dimension, some on the temporal one, some on multimedia contents or specific recommendation analyses. Some features are still missing from existing applications, such as the automatic classification of new contents and the dynamic grouping of contents based on their topics. Another issue is to support collaborative work in applications such as urban planning or collaborative research, or in highly dynamic contexts, like citizen journalism, where the contents vary in a seamless way. Moreover, existing applications are typically single-purpose applications, developed for a specific target segment. What is still lacking is a dynamic integrated management of the three main dimensions – geographic, temporal, thematic – in a way that a seamless visualisation is offered to different user categories.

To test the features of the framework, a prototype application has been developed. The intent is to obtain a dynamic adaptive system, designed to adapt not only to different thematic contexts but also to contents that vary over time, therefore not entirely available in advance. This is achieved via *self-adaption*, based on automatic content classification and a set of semantic tags. A hybrid approach combining human intervention and ML algorithm is used to classify contents into topic ontologies, that lay the foundations of a layer-based organisation. The navigation interface is equipped with adaptive filtering based on the topic ontology, which helps retrieve contents of interest and reduce information overload.

The development of MAGIS prototype in Citizen Journalism domain context successfully tested the feasibility and the effectiveness of the proposed solution.

Future implementations could deepen the following aspects:

1. Automating content collection and fetching, with the objective of massively importing large datasets of geo-referenced multimedia contents.
2. Exploring other types of classification algorithms to improve performances in the classification of new contents.
3. Investigating automatic tools for content validation and data quality control.
4. Improving interactive data enrichment and enable an effective citizens' participation.

Acknowledgments

This work was partially supported by the European Union's Horizon 2020 research and innovation program under grant agreement No. 826232, project WorkingAge (SmartWorking environments for all Ages). It was also partially supported by the Seamless Project of Regione Lombardia.

We acknowledge the ILAUD [4] participants in the Dergano Workshop held in Milano, July 19-23, 2021, where urban design participatory design has been debated with administrators and citizens, also on the idea of a digital framework for Milano map enrichment towards the participation of people to urban assessment. We are thankful to Proff. P. Guerrieri and S. Comai for common work on maps and participatory design.

This project work was published in:

1. Proceedings of the 15th European Conference on Software Architecture (ECSA 2021) with the title *"A Framework for Adaptive Context and User-Related Management of Multimedia Contents (short paper)"* [127].
2. "*Advances in Information and Communication*" as Proceedings of the 2022 Future of Information and Communication Conference (FICC) (Springer, Volume 1), with the title "*Semantic Adaptive Enrichment of Cartography for Intangible Cultural Heritage and Citizen Journalism*" [128].

References

[1] C. D. Cobb, 'Geospatial Analysis: A New Window Into Educational Equity, Access, and Opportunity', *Rev. Res. Educ.*, vol. 44, no. 1, pp. 97–129, Mar. 2020, doi: 10.3102/0091732X20907362.

[2] L. McNabb, R. S. Laramee, and R. Fry, 'Dynamic Choropleth Maps – Using Amalgamation to Increase Area Perceivability', in *2018 22nd International Conference Information Visualisation (IV)*, Jul. 2018, pp. 284–293. doi: 10.1109/iV.2018.00056.

[3] T. Blaschke, H. Merschdorf, P. Cabrera-Barona, S. Gao, E. Papadakis, and A. Kovacs-Györi, 'Place versus Space: From Points, Lines and Polygons in GIS to Place-Based Representations Reflecting Language and Culture', *ISPRS Int. J. Geo-Inf.*, vol. 7, no. 11, Art. no. 11, Nov. 2018, doi: 10.3390/ijgi7110452.

[4] 'ILAUD WEEK - Cities under Shocks & Stresses 2021 - Ilaud'. https://www.ilaud.org/ilaud-week-cities-under-shocks-stresses-2021/ (accessed Feb. 12, 2022).

[5] S. Gordillo and R. Laurini, 'Conceptual Modeling of Geographic Applications', in *Advanced Geographic Information Systems -Volume I*, vol. 1, EOLSS Publications, 2009.

[6] 'OpenStreetMap', *Wikipedia*. Feb. 11, 2022. Accessed: Feb. 12, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=OpenStreetMap&oldid=1071136889

[7] 'GEOFABRIK // Home'. https://www.geofabrik.de/ (accessed Feb. 12, 2022).

[8] 'BBBike extracts OpenStreetMap'. https://extract.bbbike.org/ (accessed Feb. 12, 2022).

[9] 'Database - Eurostat'. https://ec.europa.eu/eurostat/data/database (accessed Feb. 12, 2022).

[10] 'Set di dati - data.europa.eu'. https://data.europa.eu/data/datasets?catalog=european-union-open-data-portal&showcatalogdetails=true&minScoring=0&locale=it (accessed Feb. 12, 2022).

[11] 'resources.data.gov A repository of Federal Enterprise Data Resources'. https://resources.data.gov/ (accessed Feb. 12, 2022).

[12] 'Volunteered Geographic Information (VGI) | U.S. Geological Survey'. https://www.usgs.gov/center-of-excellence-for-geospatial-information-science-%28cegis%29/volunteered-geographic-information (accessed Feb. 12, 2022).

[13] M. Parentini, 'Definizione di "Cartografia digitale"', *TesiOnline*. https://www.tesionline.it/glossario/3002/cartografia-digitale (accessed Feb. 12, 2022).

[14] 'Cartografia Digitale: Cos'è e come si reperisce', Univerità degli Studi di Firenze, Mar. 06, 2017. Accessed: Feb. 12, 2022. [Online]. Available: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiFle2tp_r1AhWKyKQKHaU3AREQFnoECAMQAQ&url=https%3A%2F%2Fe-l.unifi.it%2Fmod%2Fresource%2Fview.php%3Fid%3D77589&usg=AOvVaw2ycUvFRsPEgY1lEOOE1ofg

[15] 'Chapter 2: Spatial Concepts and Data Models 2.1 Introduction 2.2 Models of Spatial Information 2.3 Three-Step Database Design 2.4 Extending ER with Spatial - [PPT Powerpoint]', *vdocument.in*. https://vdocument.in/chapter-2-spatial-concepts-and-data-models-21-introduction-22-models-of.html (accessed Feb. 12, 2022).

[16] 'What is GIS? | Geographic Information System Mapping Technology', *Esri*. https://www.esri.com/en-us/what-is-gis/overview (accessed Feb. 12, 2022).

[17] 'History of GIS | Timeline of Early History & the Future of GIS', *Esri*. https://www.esri.com/en-us/what-is-gis/history-of-gis (accessed Feb. 12, 2022).

[18] 'About ArcGIS | Mapping & Analytics Software and Services', *Esri*. https://www.esri.com/en-us/arcgis/about-arcgis/overview (accessed Jul. 12, 2021).

[19] 'ArcGIS Online | Web GIS Mapping Software for Everyone', *Esri*. https://www.esri.com/en-us/arcgis/products/arcgis-online/overview (accessed Feb. 12, 2022).

[20] P. U. S. P. Italia, 'GISMaker, programma per l'elaborazione e la manipolazione di dati geometrici georeferenziati', *GEOmedia*, vol. 19, no. 4, Art. no. 4, Oct. 2015, Accessed: Feb. 12, 2022. [Online]. Available: https://mediageo.it/ojs/index.php/GEOmedia/article/view/1234

[21] 'CAD GIS. Il software GIS con un CAD nativo integrato', *ProgeCAD*. http://www.progesoft.com/it/l/cad-gis/ (accessed Feb. 12, 2022).

[22] P. G. D. Group, 'PostgreSQL', *PostgreSQL*, Feb. 12, 2022. https://www.postgresql.org/ (accessed Feb. 12, 2022).

[23] 'PostGIS — PostGIS Feature List'. http://postgis.net/features/ (accessed Jul. 12, 2012).

[24] 'What does point of interest mean?', *Definitions.net*. https://www.definitions.net/definition/point%20of%20interest (accessed Feb. 12, 2022).

[25] B. Liu, Y. Fu, Z. Yao, and H. Xiong, 'Learning geographical preferences for point-of-interest recommendation', in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, Agosto 2013, pp. 1043–1051. doi: 10.1145/2487575.2487673.

[26] C. Cheng, H. Yang, M. R. Lyu, and I. King, 'Where you like to go next: successive point-of-interest recommendation', in *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, Beijing, China, Agosto 2013, pp. 2605–2611.

[27] 'Point of Interest (POI) - Are these points really necessary for Mapping?? | Ceinsys', *Ceinsys Tech Ltd*, May 15, 2017. https://www.ceinsys.com/blog/point-of-interest-really-necessary-for-mapping/ (accessed Feb. 12, 2022).

[28] 'Mappe satellitari HD (Earth Maps Street View)', *Earth It*. https://earth-it.com/ (accessed Feb. 12, 2022).

[29] 'Street View - Satellite Maps', *Street View*. https://www.street-view.net/ (accessed Feb. 12, 2022).

[30] M. Yuan, 'Temporal GIS and Spatio-Temporal Modeling Abstract I. Introduction II. The Trend of Temporal Data Modeling in GIS', Department of Geography, The University of Oklahoma. Accessed: Feb. 12, 2022. [Online]. Available: https://scholar.googleusercontent.com/scholar?q=cache:s-LSw9zEi84J:scholar.google.com/+Temporal+GIS+and+Spatio-Temporal+Modeling&hl=it&as_sdt=0,5

[31] M. P. Armstrong, 'Temporality in Spatial Databases - University of Iowa', in *GIS/LIS 88 Proceedings: Accessing the World*, 1988, p. pp.880-889. Accessed: Feb. 12, 2022. [Online]. Available: https://iro.uiowa.edu/esploro/outputs/conferenceProceeding/Temporality-in-Spatial-Databases/9983557341802771

[32] G. Langran and N. R. Chrisman, 'A Framework For Temporal Geographic Information', *Cartogr. Int. J. Geogr. Inf. Geovisualization*, vol. 25, no. 3, pp. 1–14, Oct. 1988, doi: 10.3138/K877-7273-2238-5Q6V.

[33] M. F. Worboys, 'A Unified Model for Spatial and Temporal Information', *Comput. J.*, vol. 37, no. 1, pp. 26–34, Gennaio 1994, doi: 10.1093/comjnl/37.1.26.

[34] W. Siabato, C. Claramunt, S. Ilarri, and M. A. Manso-Callejo, 'A Survey of Modelling Trends in Temporal GIS', *ACM Comput. Surv.*, vol. 51, no. 2, p. 30:1-30:41, Apr. 2018, doi: 10.1145/3141772.

[35] L. Múzquiz, 'Atlante Storico Mondiale Interattivo dal 3000 aC | GeaCron', *GeaCron*, 2011. http://geacron.com/home-it/?lang=it (accessed Jul. 05, 2021).

[36] EPFL, 'Unleashing Big Data of the Past – Europe builds a Time Machine', Mar. 2019, Accessed: Feb. 12, 2022. [Online]. Available: https://actu.epfl.ch/news/unleashing-big-data-of-the-past-europe-builds-a-ti/

[37] 'Venice Time Machine Project – Current state of affairs', *Time Machine Europe*. https://www.timemachine.eu/venice-time-machine-project-current-state-of-affairs/ (accessed Feb. 12, 2022).

[38] 'MyTravelMap', *MyTravelMap*. https://www.mytravelmap.xyz/?0 (accessed Feb. 12, 2022).

[39] M. Carminati, 'MyTripMap', *mytripmap*. https://www.mytripmap.it/ (accessed Feb. 12, 2022).

[40] S. T. Rahim, K. Zheng, S. Turay, and Y. Pan, 'Capabilities of Multimedia GIS', *Chin. Geogr. Sci.*, vol. 9, no. 2, pp. 159–165, Jun. 1999, doi: 10.1007/BF02791367.

[41] 'ArcGIS Insights | Documentazione — Aggiungere testo ed elementi multimediali', *Esri*. https://doc.arcgis.com/it/insights/latest/share/add-text-and-media.htm (accessed Feb. 12, 2022).

[42] 'ArcGIS StoryMaps | Digital Storytelling with Maps', *Esri*. https://www.esri.com/en-us/arcgis/products/arcgis-storymaps/overview (accessed Feb. 12, 2022).

[43] R. GEOmedia, 'ArcGIS StoryMaps: il nuovissimo strumento di Esri per creare storie con le mappe', *Rivistageomedia*. https://rivistageomedia.it/2019082916561/BIM-CAD-GIS/arcgis-storymaps-il-nuovissimo-strumento-di-esri-per-creare-storie-con-le-mappe (accessed Feb. 12, 2022).

[44] A. Azzini and F. C. Pavesi, 'Introduzione ai GIS - Università degli studi di Bergamo', presented at the Corso base di ArcView - ArcGIS Desktop 10, Università degli Studi di Bergamo, May 06, 2011. Accessed: Feb. 12, 2022. [Online]. Available: https://studylibit.com/doc/7527126/1.-introduzione-ai-gis---università-degli-studi-di-bergamo

[45] 'LearnOSM Impara ad usare OpenStreetMap passo dopo passo', *LearnOSM*, Jul. 25, 2017. https://learnosm.org/it/osm-data/data-overview/ (accessed Feb. 12, 2022).

[46] 'Geodatabase - GIS Wiki | The GIS Encyclopedia', *wiki.gis.com*, Sep. 27, 2009. http://wiki.gis.com/wiki/index.php/Geodatabase (accessed Feb. 12, 2022).

[47] 'Guida pratica ai principali formati CAD', *Planstudio Software Innovation*. https://blog.planstudio.it/guida-pratica-ai-principali-formati-cad (accessed Mar. 27, 2022).

[48] P. Shoval, R. Danoch, and M. Balaban, 'Hierarchical entity-relationship diagrams: The model, method of creation and experimental evaluation', *Requir. Eng.*, vol. 9, pp. 217–228, Gennaio 2004, doi: 10.1007/s00766-004-0201-9.

[49] Y. Huang, M. Yuan, Y. Sheng, X. Min, and Y. Cao, 'Using Geographic Ontologies and Geo-Characterization to Represent Geographic Scenarios', *ISPRS Int. J. Geo-Inf.*, vol. 8, p. 566, Dicembre 2019, doi: 10.3390/ijgi8120566.

[50] 'What is the Data, Information, Knowledge, Wisdom (DIKW) Pyramid?', *Ontotext*. https://www.ontotext.com/knowledgehub/fundamentals/dikw-pyramid/ (accessed Feb. 12, 2022).

[51] C. Vercellis, *Business intelligence. Modelli matematici e sistemi per le decisioni.* Milano: McGraw-Hill, 2006. Accessed: Feb. 12, 2022. [Online]. Available: http://hdl.handle.net/11311/505699

[52] A. Twin, A. Drury, and M. Reeves, 'Data Mining Definition', *Investopedia*, Sep. 17, 2021. https://www.investopedia.com/terms/d/datamining.asp (accessed Feb. 12, 2022).

[53] A. L. Samuel, 'Some studies in machine learning using the game of checkers', *IBM J. Res. Dev.*, vol. 44, no. 1.2, pp. 206–226, Jan. 2000, doi: 10.1147/rd.441.0206.

[54] 'Tutto il valore dei Big Data: cosa sono e perché sono così importanti!', *Osservatori.net Digital Innovation*. https://blog.osservatori.net/it_it/big-data-cosa-sono (accessed Feb. 12, 2022).

[55] M. Kanevski, L. Foresti, C. Kaiser, A. Pozdnoukhov, V. Timonin, and D. Tuia, 'Machine learning models for geospatial data', in *Handbook of Theoretical and Quantitative Geography*, Faculty of Geosciences and Environment, University of Lausanne, Switzerland: François Bavaud, Christophe Mager, 2009, pp. 175–227. [Online]. Available: https://www.researchgate.net/publication/261551597_Machine_learning_models_for_geospatial_data

[56] D. Feldmeyer, C. Meisch, H. Sauter, and J. Birkmann, 'Using OpenStreetMap Data and Machine Learning to Generate Socio-Economic Indicators', *ISPRS Int. J. Geo-Inf.*, vol. 9, p. 498, Agosto 2020, doi: 10.3390/ijgi9090498.

[57] 'Meet Picterra - Geospatial cloud-based platform', *Picterra*. https://picterra.ch/mission/ (accessed Feb. 12, 2022).

[58] M. Altaweel, 'Machine Learning and Object Detection in Spatial Analysis', *GIS Lounge*, Dec. 09, 2020. https://www.gislounge.com/machine-learning-and-object-detection-in-spatial-analysis/ (accessed Feb. 12, 2022).

[59] M. Camacho-Collados and F. Liberatore, 'A Decision Support System for predictive police patrolling', *Decis. Support Syst.*, vol. 75, pp. 25–37, Luglio 2015, doi: 10.1016/j.dss.2015.04.012.

[60] 'GeoRaster in Oracle Databas'. Oracle, Mar. 2017. Accessed: Feb. 12, 2022. [Online]. Available: https://docs.oracle.com/database/121/GEORS/geor_intro.htm#GEORS100

[61] 'PL/SQL for Developers | Oracle Italia', *Oracle*. https://www.oracle.com/it/database/technologies/appdev/plsql.html (accessed Feb. 12, 2022).

[62] R. R. Vatsavai, A. Ganguly, V. Chandola, A. Stefanidis, S. Klasky, and S. Shekhar, 'Spatiotemporal data mining in the era of big spatial data: algorithms and applications', in *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data*, New York, NY, USA, Nov. 2012, pp. 1–10. doi: 10.1145/2447481.2447482.

[63] J. Han, K. Koperski, and N. Stefanovic, 'GeoMiner: a system prototype for spatial data mining', *ACM SIGMOD Rec.*, vol. 26, no. 2, pp. 553–556, Giugno 1997, doi: 10.1145/253262.253404.

[64] A. Papandreou, 'Development and utility of georeferenced analytical tools in rural areas', *Manag. Econ. Eng. Agric. Rural Dev.*, vol. 18, no. 4, pp. p225-228, 2018.

[65] A. Baviera-Puig, J. Buitrago-Vera, and C. Escriba-Perez, 'Geomarketing models in supermarket location strategies', *J. Bus. Econ. Manag.*, vol. 17, no. 6, pp. 1205–1221, Nov. 2016, doi: 10.3846/16111699.2015.1113198.

[66] S. Kang, S. Heo, M. J. Realff, and J. H. Lee, 'Three-stage design of high-resolution microalgae-based biofuel supply chain using geographic information system', *Appl. Energy*, vol. 265, p. 114773, May 2020, doi: 10.1016/j.apenergy.2020.114773.

[67] F. Perazzoni, P. Bacelar-Nicolau, and M. Painho, 'Geointelligence against Illegal Deforestation and Timber Laundering in the Brazilian Amazon', *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 6, p. 398, Jun. 2020.

[68] M. R. Walter, S. M. Hemachandra, B. S. Homberg, S. Tellex, and S. Teller, 'Learning Semantic Maps from Natural Language Descriptions', *Int. J. Robot. Res.*, Jun. 2013, Accessed: Feb. 13, 2022. [Online]. Available: https://dspace.mit.edu/handle/1721.1/87051

[69] A. Ballatore, M. Bertolotto, and D. C. Wilson, 'Geographic knowledge extraction and semantic similarity in OpenStreetMap', *Knowl. Inf. Syst.*, vol. 37, no. 1, pp. 61–81, Oct. 2013, doi: 10.1007/s10115-012-0571-0.

[70] T. Sobral, T. Galvão, and J. Borges, 'An Ontology-based approach to Knowledge-assisted Integration and Visualization of Urban Mobility Data', *Expert Syst. Appl.*, vol. 150, p. 113260, Luglio 2020, doi: 10.1016/j.eswa.2020.113260.

[71] M. Ben Ellefi *et al.*, 'Ontology-based web tools for retrieving photogrammetric cultural heritage models', *ISPRS - Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. XLII-2/W10, pp. 31–38, Apr. 2019, doi: 10.5194/isprs-archives-XLII-2-W10-31-2019.

[72] E. Meyer, P. Grussenmeyer, J. Perrin, A. Durand, and P. Drap, 'A web information system for the management and the dissemination of Cultural Heritage data', *J. Cult. Herit.*, vol. 8, pp. 396–411, Sep. 2007, doi: 10.1016/J.CULHER.2007.07.003.

[73] T. MacWright, 'The difference between XYZ and TMS tiles and how to convert between them', *Github*. https://gist.github.com/tmcw/4954720 (accessed Feb. 19, 2022).

[74] 'Visualizzazioni dei dati: 10 esempi di mappe interattive', *Tableau*. https://www.tableau.com/it-it/learn/articles/interactive-map-and-data-visualization-examples (accessed Feb. 13, 2022).

[75] J. Fung, 'Manhattan Population Explorer', *Manhattan Population Explorer*. http://manpopex.us/ (accessed Feb. 13, 2022).

[76] 'Google Maps'. https://www.google.it/maps (accessed Feb. 13, 2022).

[77] J. Howard, 'Find the Right Hotels & Neighborhoods near Public Transit to Manhattan (NYC)', *NJ Hotels Near NYC*. https://njhotelsnearnyc.com/ (accessed Feb. 13, 2022).

[78] J. Davis, 'At Minimum', *Tableau Public*, Jan. 16, 2019. https://public.tableau.com/app/profile/justindavis/viz/AtMinimum/AtMinimum (accessed Feb. 13, 2022).

[79] S. Afzal, R. Maciejewski, Y. Jang, N. Elmqvist, and D. Ebert, 'Spatial Text Visualization Using Automatic Typographic Maps', *IEEE Trans. Vis. Comput. Graph.*, vol. 18, pp. 2556–2564, 2012, doi: 10.1109/TVCG.2012.264.

[80] G. A. Lee, A. Dünser, S. Kim, and M. Billinghurst, 'CityViewAR: A mobile outdoor AR application for city visualization', in *2012 IEEE International Symposium on Mixed and Augmented Reality - Arts, Media, and Humanities (ISMAR-AMH)*, Nov. 2012, pp. 57–64. doi: 10.1109/ISMAR-AMH.2012.6483989.

[81] C. Parker and M. Tomitsch, 'Data Visualisation Trends in Mobile Augmented Reality Applications', in *Proceedings of the 7th International Symposium on Visual Information Communication and Interaction*, New York, NY, USA, Aug. 2014, pp. 228–231. doi: 10.1145/2636240.2636864.

[82] M.-J. Lobo, C. Appert, and E. Pietriga, 'MapMosaic: Dynamic Layer Compositing for Interactive Geovisualization', *Int. J. Geogr. Inf. Sci.*, vol. 31, no. 9, p. 1818, May 2017, doi: 10.1080/13658816.2017.1325488.

[83] M. D. Wilkinson *et al.*, 'The FAIR Guiding Principles for scientific data management and stewardship', *Sci. Data*, vol. 3, no. 1, Art. no. 1, Mar. 2016, doi: 10.1038/sdata.2016.18.

[84] 'FAIR Principles', *GO FAIR*. https://www.go-fair.org/fair-principles/ (accessed Feb. 13, 2022).

[85] P. Guerrieri, S. Comai, and M. G. Fugini, 'Evolving Experiences of Participation: The e-ILAUD Tool', Jul. 2021. doi: 10.13140/RG.2.2.24786.89282.

[86] L. Tanca, 'Semistructured Data Integration', presented at the Technologies for Information Systems, Politecnico di Milano, 2020.

[87] V. Lavecchia, 'Differenza tra Tassonomia, Thesaurus e Ontologia nel Web Semantico', *Informatica e Ingegneria Online*, May 14, 2020. https://vitolavecchia.altervista.org/differenza-tra-tassonomia-thesaurus-e-ontologia-nel-web-semantico/ (accessed Feb. 13, 2022).

[88] M. T. Biagetti, 'Ontologies (as knowledge organization systems)', *Knowl. Organ.*, vol. 48, no. 2, pp. 152–176, 2021.

[89] 'What is Text Analysis? A Beginner's Guide', *MonkeyLearn*. https://monkeylearn.com/text-analysis/ (accessed Feb. 13, 2022).

[90] J. Chen, Y. Hu, J. Liu, Y. Xiao, and H. Jiang, 'Deep short text classification with knowledge powered attention', in *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, Honolulu, Hawaii, USA, Gennaio 2019, pp. 6252–6259. doi: 10.1609/aaai.v33i01.33016252.

[91] A. I. Kadhim, 'Survey on supervised machine learning techniques for automatic text classification', *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 273–292, Jun. 2019, doi: 10.1007/s10462-018-09677-1.

[92] J. Hartmann, J. Huppertz, C. Schamp, and M. Heitmann, 'Comparing automated text classification methods', *Int. J. Res. Mark.*, vol. 36, no. 1, pp. 20–38, Mar. 2019, doi: 10.1016/j.ijresmar.2018.09.009.

[93] 'Short Text Classification', *MonkeyLearn*. https://monkeylearn.com/short-text-classification/ (accessed Feb. 13, 2022).

[94] M. Horridge, 'A practical guide to building owl ontologies using protégé 4 and co-ode tools edition1. 3', *Univ. Manch.*, vol. 107, Mar. 2009, [Online]. Available: http://mowl-power.cs.man.ac.uk/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_3.pdf

[95] G. Xiao *et al.*, 'Ontology-Based Data Access: A Survey', in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 2018, pp. 5511–5519. doi: 10.24963/ijcai.2018/777.

[96] 'OBDA Systems: Semantic Solutions for Enterprise Data Management', *OBDA Systems*. http://obdm.obdasystems.com/ (accessed Feb. 13, 2022).

[97] I. Cantador, M. Fernández, D. Vallet, P. Castells, J. Picault, and M. Ribière, 'A Multi-Purpose Ontology-Based Approach for Personalised Content Filtering and Retrieval', in *Advances in Semantic Media Adaptation and Personalization*, vol. 93, M. Wallace, M. C. Angelides, and P. Mylonas, Eds. Berlin, Heidelberg: Springer, 2008, pp. 25–51. doi: 10.1007/978-3-540-76361_2.

[98] I. Cantador, A. Bellogín, and P. Castells, 'A multilayer ontology-based hybrid recommendation model', *AI Commun.*, vol. 21, no. 2–3, pp. 203–210, Jan. 2008, doi: 10.3233/AIC-2008-0437.

[99] A. Sieg, B. Mobasher, and R. Burke, 'Improving the effectiveness of collaborative recommendation with ontology-based user profiles', *Proc. 1st Int. Workshop Inf. Heterog. Fusion Recomm. Syst.*, pp. 39–46, Sep. 2010, doi: 10.1145/1869446.1869452.

[100] 'Meta-ontology', *Wikipedia*. Dec. 09, 2021. Accessed: Feb. 13, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Meta-ontology&oldid=1059448577

[101] 'Meta-ontologia', *it.knowledgr.com*. https://it.knowledgr.com/04199683/Metaontologia (accessed Feb. 13, 2022).

[102] 'Time ontology - Italian application profile', *WebVOWL*, 2020. https://ontopia-lodview.agid.gov.it/webvowl/#iri=https://w3id.org/italia/onto/TI (accessed Feb. 13, 2022).

[103] M. N. Mahdi, A. R. Ahmad, R. Ismail, H. Natiq, and M. A. Mohammed, 'Solution for Information Overload Using Faceted Search–A Review', *IEEE Access*, vol. 8, pp. 119554–119585, 2020, doi: 10.1109/ACCESS.2020.3005536.

[104] L. AB, 'What is faceted search and navigation?', *Loop54*. https://www.loop54.com/knowledge-base/what-is-faceted-search-navigation (accessed Feb. 13, 2022).

[105] K. Whitenton, 'Filters vs. Facets: Definitions', *Nielsen Norman Group*, Mar. 16, 2014. https://www.nngroup.com/articles/filters-vs-facets/ (accessed Feb. 13, 2022).

[106] M. Camanes, 'Faceted navigation for SEO best practices', *Builtvisible*, Aug. 29, 2017. https://builtvisible.com/faceted-navigation-seo-best-practices/ (accessed Feb. 13, 2022).

[107] 'Web Ontology Language', *Wikipedia*. Feb. 02, 2022. Accessed: Feb. 13, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Web_Ontology_Language&oldid=1069505816

[108] L. Delucchi, M. Napolitano, and A. Zanol, 'Introduzione ad OpenStreetMap'. Sep. 2017. Accessed: Feb. 13, 2022. [Online]. Available: https://studylibit.com/doc/1706541/introduzione-ad-openstreetmap

[109] A. Williams, 'NoSQL Document-Oriented Database: a detailed overview', *RavenDB*, Mar. 18, 2021. https://ravendb.net/articles/nosql-document-oriented-databases-detailed-overview (accessed Feb. 13, 2022).

[110] A. Williams, 'NoSQL database types explained: Column-oriented databases', *SearchDataManagement*, Sep. 22, 2021. https://searchdatamanagement.techtarget.com/tip/NoSQL-database-types-explained-Column-oriented-databases (accessed Feb. 13, 2022).

[111] 'Home | Aruba.it', *Aruba.it*. https://www.aruba.it/home.aspx (accessed Feb. 13, 2022).

[112] 'Decimal or Point Data Type for storing Geo location data in MySQL', *Database Administrators Stack Exchange*, 2017. https://dba.stackexchange.com/questions/107089/decimal-or-point-data-type-for-storing-geo-location-data-in-mysql (accessed Jul. 20, 2021).

[113] O. Otero, *Embed*. 2022. Accessed: Feb. 13, 2022. [Online]. Available: https://github.com/oscarotero/Embed

[114] R. Garreta, 'How are Text Classifiers Trained?', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2173817-how-are-text-classifiers-trained (accessed Feb. 13, 2022).

[115] R. Maguire, 'Defining Tags for Classifiers', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2173836-defining-tags-for-classifiers (accessed Feb. 13, 2022).

[116] R. Maguire, 'Language Settings', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2174106-language-settings (accessed Feb. 13, 2022).

[117] R. Maguire, 'Changing Algorithms', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2174103-changing-algorithms (accessed Feb. 13, 2022).

[118] R. Maguire, 'N-gram range', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2174105-n-gram-range (accessed Feb. 13, 2022).

[119] R. Maguire, 'Max Features', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2174109-max-features (accessed Feb. 13, 2022).

[120] R. Maguire, 'Stemming', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2174118-stemming (accessed Feb. 13, 2022).

[121] R. Maguire, 'Preprocessing Numbers', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2174114-preprocessing-numbers (accessed Feb. 13, 2022).

[122] Maguire, 'Filter Stopwords', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2174123-filter-stopwords (accessed Feb. 13, 2022).

[123] R. Maguire, 'Whitelisting Words', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2174117-whitelisting-words (accessed Feb. 13, 2022).

[124] R. Maguire, 'Understanding Classifier Statistics', *MonkeyLearn*. http://help.monkeylearn.com/en/articles/2173838-understanding-classifier-statistics (accessed Feb. 13, 2022).

[125] D. Siegle, 'Confidence Intervals and Levels', *Educational Research Basics*, May 22, 2015. https://researchbasics.education.uconn.edu/confidence-intervals-and-levels/ (accessed Feb. 13, 2022).

[126] A. Savand, 'Italian Stop Words', *Github*, Feb. 04, 2022. https://github.com/Alir3z4/stop-words/blob/3366e47dec3153fae90add3fa2a02a498f76e507/italian.txt (accessed Feb. 13, 2022).

[127] M. Fugini, J. Finocchi, and E. Rossi, 'A Framework for Adaptive Context and User-Related Management of Multimedia Contents (short paper)', in *ECSA 2021 Companion Volume*, Virtual (originally: Växjö, Sweden), Sep. 2021, vol. 2978. Accessed: Mar. 19, 2022. [Online]. Available: http://ceur-ws.org/Vol-2978/#casa-paper6

[128] M. Fugini, J. Finocchi, and E. Rossi, 'Semantic Adaptive Enrichment of Cartography for Intangible Cultural Heritage and Citizen Journalism', in *Advances in Information and Communication*, Cham, 2022, pp. 173–185. doi: 10.1007/978-3-030-98012-2_14.

List of Figures

List of Tables

Attachments

Attachment *2022_04_Rossi_02.PDF* includes the Executive Summary of the present dissertation. Following the guidelines provided by Politecnico di Milano, it reports a critical overview of this Thesis with a focus on the main achievements that have emerged from the research.

Attachment *2022_04_Rossi_03.PDF* includes the details about the dataset used in the CJ prototype described in the Thesis. The file presents a collection of the tables of MAGIS Database, namely Metadata, MetadataTags, Ontology, POI, Tags.

Adaptive Management of Multimedia and Georeferenced Contents | Elisa Rossi