

TIN2009-11005
DAMASK

Data-Mining Algorithms with Semantic Knowledge

PROYECTO DE INVESTIGACIÓN
PROGRAMA NACIONAL DE INVESTIGACIÓN FUNDAMENTAL,
PLAN NACIONAL DE I+D+i 2008-2011
ÁREA TEMÁTICA DE GESTIÓN: Tecnologías informáticas

Internal project report Subtask T-3.2 Data matrix construction

Authored by

Ferran Mata, Universitat Rovira i Virgili
Aïda Valls, Universitat Rovira i Virgili



Document information

project name:	DAMASK	
Project reference:	TIN2009-11005	
type of document:	Internal Report	
file name:	DAMASK report T3-2.pdf	
version:	1.0	
authored by:	F. Mata, A. Valls	10/04/2012
co-authored by		
released by:	A.Valls	15.04.2012
approved by:	Co-ordinator	Antonio Moreno

Document history

version	date	reason of modification
1.0	23.April.2012	First release of the data matrix description.
1.1	10.May.2012	Revised document.

Table of Contents

1	Introduction	3
2	Data matrix by data type	5
2.1	Categorical and numerical data	5
2.2	Semantic data	8
2.2.1	Selecting the semantic attributes	11
2.3	The DAMASK data matrix	14
3	References	16

1 Introduction

This document is the result of Task T3.2 “DAMASK data matrix”, according to the planning shown in Figure 1. The document explains how the DAMASK data matrix has been constructed.

The goal of the DAMASK (Data Mining Algorithms with Semantic Knowledge) project is the use of semantic domain knowledge, represented in the form of ontologies, to define new methods for extracting and integrating information from heterogeneous Web resources with varying degrees of structure, performing an automatic classification, and making a semantic interpretation of the results.

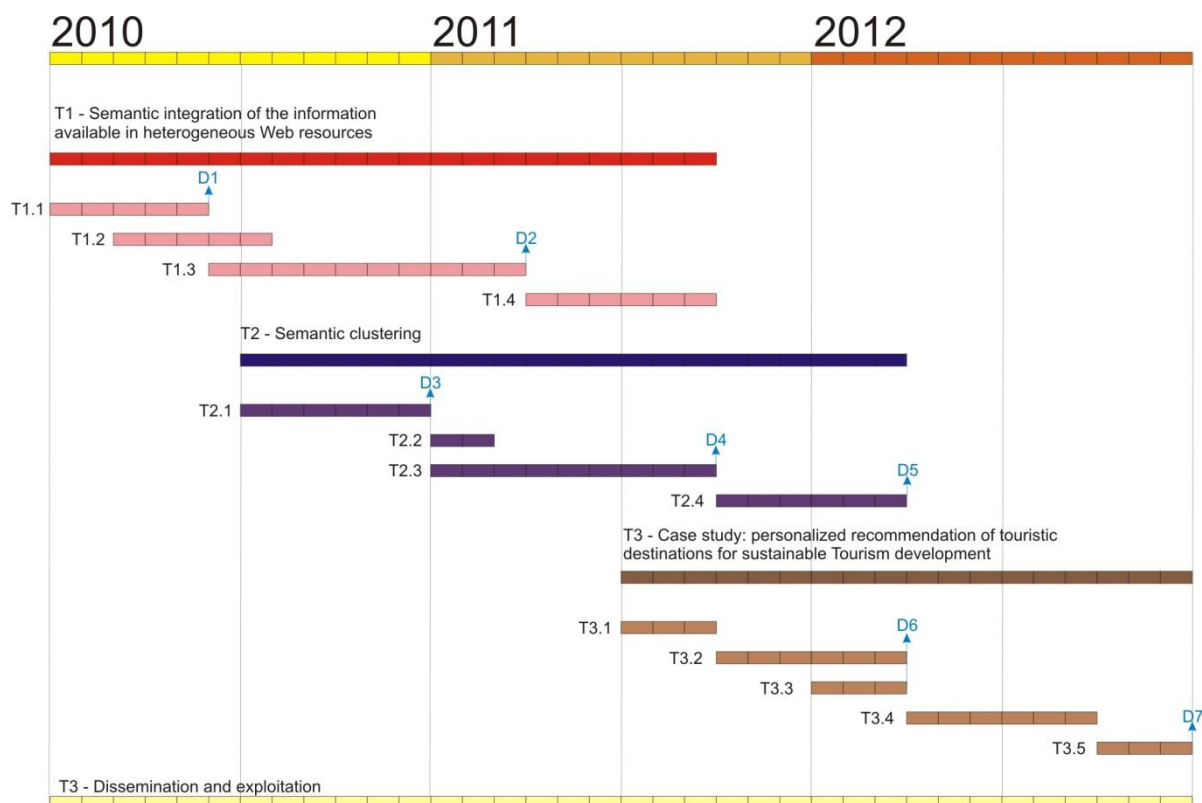


Figure 1: Tasks of DAMASK

The methods developed in tasks T1 (Semantic integration of information from the Web) and T2 (Semantic clustering) are domain-independent. Then, in Task T3, a demonstrator system in the field of Tourism and Leisure is being designed and implemented. In particular, we will construct a Web-based personalized recommender system of touristic destinations for sustainable Tourism development of the most suitable destinations for tourists is studied. Subtask T3-1 was devoted to the creation of a domain ontology specific for this case study. In subtask T3-2, we have applied the methods of semantic extraction and integration of information (developed in task T1) in order to obtain a data matrix that gathers all the information available in Web resources about some touristic destinations.

The architecture of the recommender system is given in Figure 2.

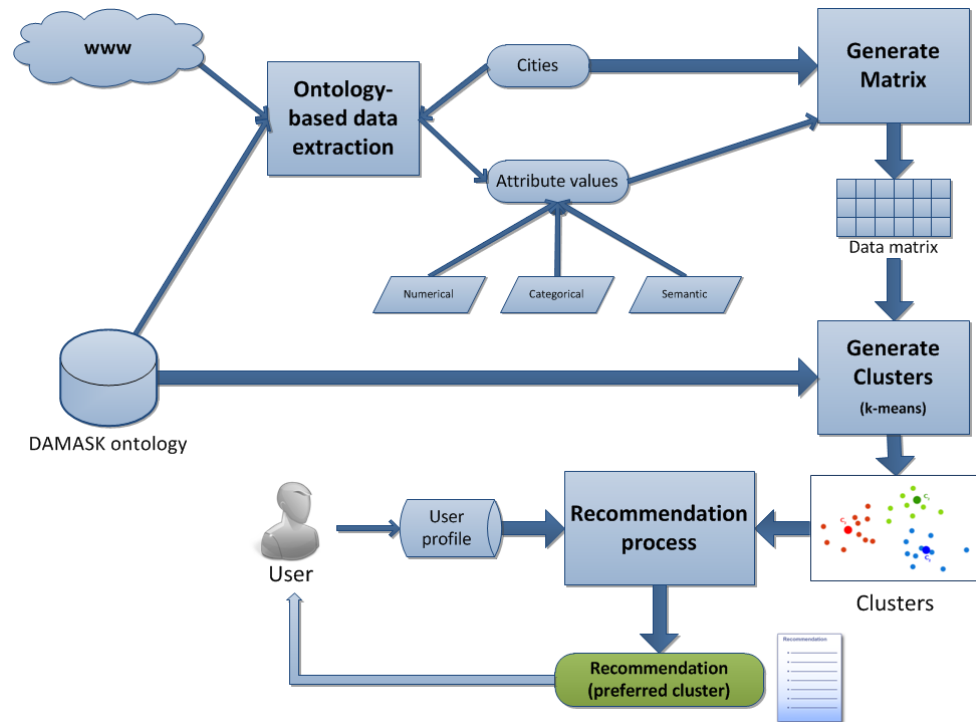


Figure 2: Recommender steps scheme

The process starts with the generation of a data matrix with a diversified sample of cities from all over the world. For each city a set of attributes has been considered. They can be numerical, categorical or semantic (see Table 1). Depending on the nature of the values, different data extraction methods are used to automatically obtain the descriptions of each city from the data available in the Web. In this document we explain how the values of each type of attribute have been obtained.

A set of 150 cities has been selected. These cities are the 150 leading and most dynamic cities in terms of tourist arrivals, according to the ranking made by Euromonitor International in 2006 (Bremner, 2007).

The cities are the following: Aberdeen, Abu Dhabi, Agra, Amsterdam, Antwerp, Atlanta, Bahrain, Bangkok, Barcelona, Bath (Somerset), Beijing, Benidorm, Berlin, Bilbao, Birmingham, Boston, Bratislava, Bregenz, Brighton, Bristol, Bruges, Budapest, Buenos Aires, Cairo, Cambridge, Cancun, Cape Town, Cardiff, Chengdu, Chennai, Chester, Chicago, Chongqing, Copenhagen, Dalian, Dijon, Dresden, Dubai, Dublin, Edinburgh, Florence, Florianopolis, Fortaleza, Foz do Iguaçu, Geneva, Genoa, Ghent, Glasgow, Goa, Gothenburg, Granada, Graz, Guangzhou, Guilin, Hamburg, Hangzhou, Havana, Heidelberg, Helsinki, Hong Kong, Honolulu, Houston, Innsbruck, Inverness, Istanbul, Jerusalem, Krakow, Kuala Lumpur, Kunming, Las Vegas (Nevada), Leeds, Linz, Lisbon, Liverpool, London, Los Angeles, Luxembourg, Lyon, Macau, Madrid, Malmö, Manchester, Marrakech, Marseille, Mecca, Melbourne, Mexico City, Miami, Milan, Monaco, Montreal, Moscow, Mumbai, Munich, Nanjing, Naples, New Delhi, New York City, Newcastle upon Tyne, Nice, Nottingham, Nuremberg, Orlando (Florida), Oslo, Oxford, Paris, Prague, Qingdao, Reading (Berkshire), Reading (Pennsylvania), Reykjavík, Rheims, Rio de Janeiro, Rome, Saint Petersburg, Salvador (Bahia), Salzburg, San Diego, San Francisco, San Jose (California), São Paulo, Seattle, Seoul, Seville, Shanghai, Shenzhen, Singapore, Stockholm, Suzhou, Sydney, Taipei, Tallinn, Tarragona, Tianjin, Tokyo, Toronto, Turku, Valencia, Spain, Varadero, Venice, Vienna, Warsaw, Washington D.C., Wuxi, Xiamen, Xi'an, York, Zaragoza, Zhuhai, Zürich.

2 Data matrix by data type

2.1 Categorical and numerical data

After the analysis of Web resources made in task T1, the following set of attributes has been selected as significant from a tourist point of view:

- Population – Numerical
- Elevation – Numerical
- Continent code – Categorical
- Climate – Categorical

For each city, the *Population* and *Elevation* data were searched by means of the methods developed in task T1, based on using semi-structured resources like Wikipedia in an automatic way (Vicent, 2009). However, for some cities there was information that has been unable to obtain with these techniques (see Deliverable D2 for more details).

To complete the data matrix other sources have been considered, such as the use of specific APIs of different Websites. The *population* and *continent code* were extracted from geonames API among many other data that was finally discarded for irrelevant or incomplete. One of the attributes from Geonames that was incomplete is the *elevation* of each city. Then, to get the elevation, the Google Maps Elevation API was used. This API does not return results by city names, but by coordinates instead. These coordinates were extracted from geonames API for each city. So, at the end the *elevation* of the city was found using its coordinates, as shown in Figure 3 and 4.

```

▼<geonames style="MEDIUM">
  <totalResultsCount>983</totalResultsCount>
  ▼<geoname>
    <toponymName>Tarragona</toponymName>
    <name>Tarragona</name>
    <lat>41.11667</lat>
    <lng>1.25</lng>
    <geonameId>3108288</geonameId>
    <countryCode>ES</countryCode>
    <countryName>Spain</countryName>
    <fcl>P</fcl>
    <fcode>PPLA2</fcode>
  </geoname>
</geonames>

```

Figure 3: XML result to a Geonames request

```

▼<ElevationResponse>
  <status>OK</status>
  ▼<result>
    ▼<location>
      <lat>41.1166700</lat>
      <lng>1.2500000</lng>
    </location>
    <elevation>34.0122147</elevation>
    <resolution>152.7032318</resolution>
  </result>
</ElevationResponse>

```

Figure 4: XML result to a G. Maps elevation request

Finally, the *climate* was get from a list available at the Köppen-Geiger website. A file with the correspondences between coordinates in Geonames was used. So, the *climate* for each city was extracted using the Geonames coordinates with the climate information list. The Köppen-Geiger climate classification uses a pattern of characters to indicate the climate of each zone, and those characters indicate the *main climate*, the *precipitation* and the *temperature*. See figure 5 for a global view of the classification.

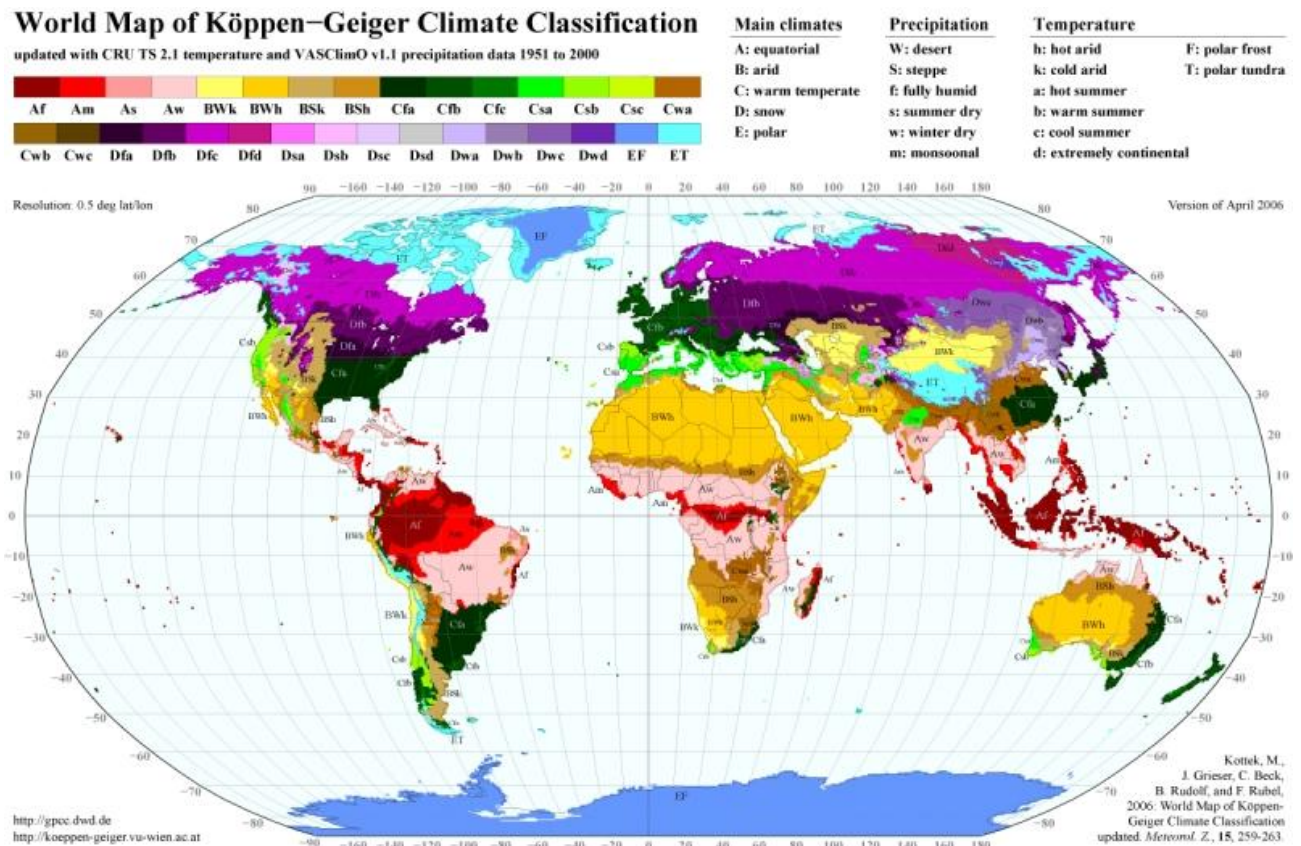


Figure 5: World map of Köppen-Geiger climate classification

As seen in figure 5, there are 31 possible climates, which is far too much for our sample of 150 cities. So, this classification can be divided into subgroups that are much more adequate for the DAMASK data matrix, and even more semantically coherent. So, the final classification taken is the following:

- Tropical rainforest: Af
- Tropical monsoon: Am
- Tropical savannah: Aw, As
- Desert: BWh, BWk, BWn
- Semi-arid: BSh, BSk
- Humid sub-tropical: Cfa, Cwa
- Oceanic: Cfb, Cwb, Cfc
- Mediterranean: Csa, Csb
- Humid continental: Dfa, Dwa, Dfb, Dwb, Dsa, Dsb
- Subarctic: Dfc, Dwc, Dfd, Dwd, Dsc, Dsd
- Polar: ET, EF

After collecting the information, basic descriptive statistics of the variables have been computed using the SPSS software.

For the numerical values, a few statistics have been calculated and are shown here. These values have been used later in the normalization step performed for calculating distances between cities.

Statistics		
	Population	Elevation
N	150	150
Avg.	2088459,11	138,915933
Tip. Dev.	3017418,78	283,177017
Min	20000	1,67
Max	14608512	2227,88
Percentiles 25	333414,25	12,605
Percentiles 50	726002	33,14
Percentiles 75	2596330,25	156,49

Table 1. Basic descriptive statistics of the numerical variables

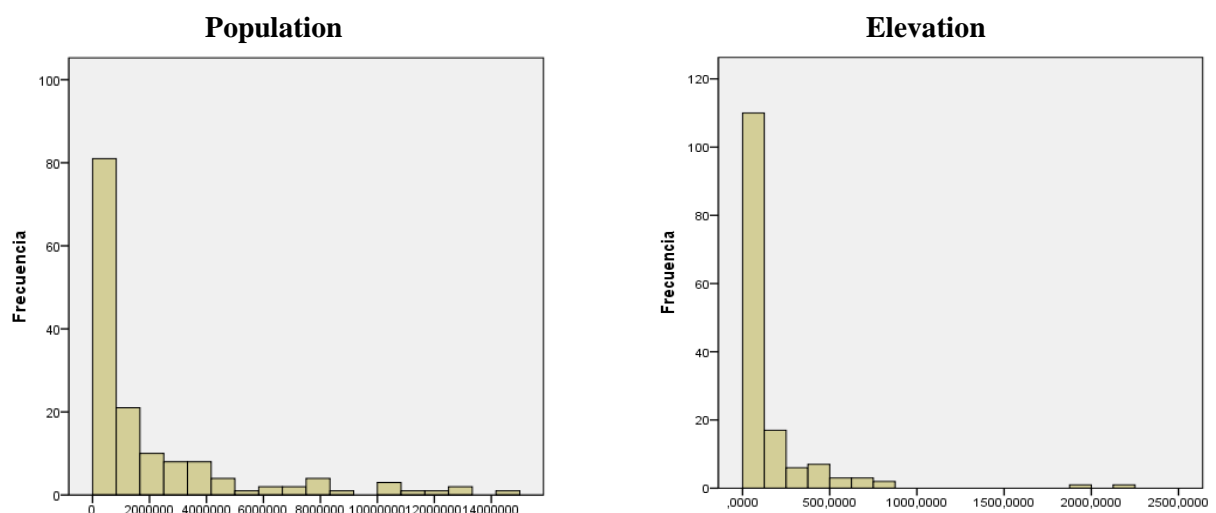


Figure 6. Frequency distribution of the numerical variables

This frequency distribution shows a large concentration of values in a very small range of the domain. This indicates that the variables are not very discriminant when used in clustering processes. Small differences must be taken into account, while minimizing the impact of large distances. This consideration has been taken when defining the comparison measure in the clustering process.

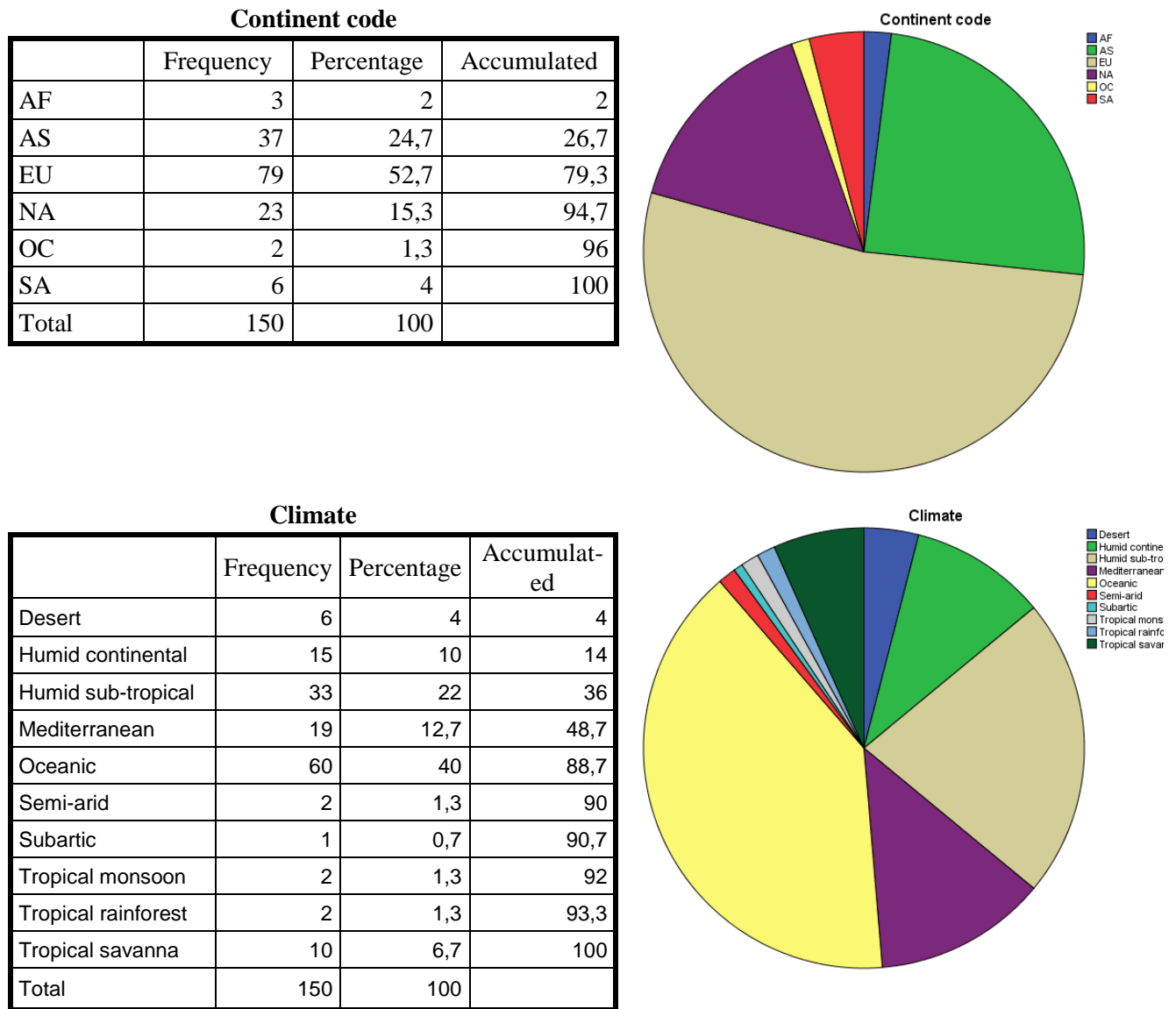


Figure 7. Frequency distribution of the categorical variables

Although we can observe a high concentration of European cities and of Oceanic climate, we consider that the distribution is quite diverse for clustering purposes.

2.2 Semantic data

The extraction methods developed in task T1 have also been used to obtain the concepts that correspond to each semantic attribute for a given city. The method is based on the representation of a subset of concepts of the DAMASK ontology that were selected as appropriate to be included as features to describe the touristic destinations. For this purpose, a tailoring of the DAMASK ontology was done, generating a simplified version (see Figure 8). The procedure is explained in (Vicient, Sánchez, & Moreno, 2011).

The Damask ontology is the result of merging and combining the following ontologies:

- **TourismOWL.owl:** models touristic points of interest for different kinds of tourist profiles. It was designed in (Vicent, 2009) based on information extracted through Wikipedia articles. It consists of 315 classes and a depth of 5 hierarchical levels. Its main classes represent concepts related with administrative divisions, buildings, festivals, landmarks, museums and sports.
- **Space.owl:** consists of 188 classes and a depth of 6 hierarchical levels. It contains concepts related with three main topics: geographical features, geopolitical entities and places.
- **PCTTO.owl:** It is focused on tourist activities. The ontology represents up to 203 connected concepts in 5 hierarchy levels. It is structured around eight main concepts, which constitute the first level of the hierarchy: “Events”, “Nature”, “Culture”, “Leisure”, “Sports”, “Towns”, “Routes” and “ViewPoints”.

The DAMASK ontology consists of 538 classes connected in 9 hierarchy levels. It is structured around 4 main concepts that constitute the first level of the hierarchy: “geopolitical division”, “activity”, “point of interest” and “geographical feature”. The Damask Ontology is not a pure taxonomy, as it contains multi-inheritance between concepts.

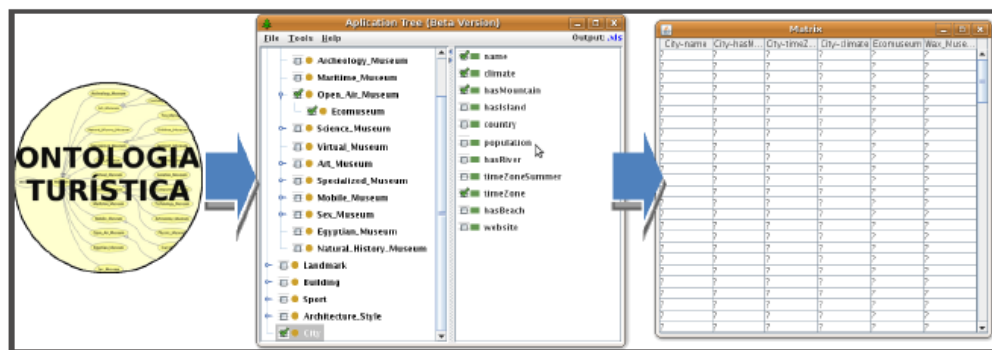


Figure 8: Procedure for the ontology-based feature extraction

A tool for visualizing and manipulating this ontology has been created, named Tree (Figure 9). This tool permits to select a subset of concepts to be used as attributes. If the concept is a leaf on the taxonomy, then the attribute is Binary (a city may have this concept or not). Otherwise the concept generated a “semantic attribute”, whose possible values are all the concepts that descend from it.

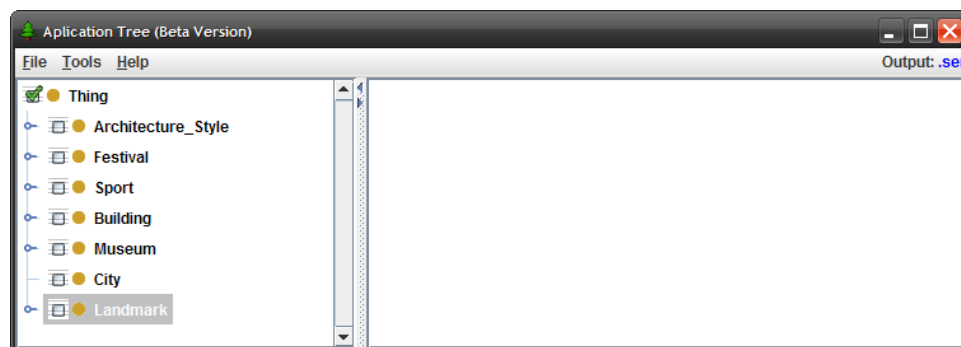


Figure 9: Visualization and manipulation tool for generating the semantic attributes

The main attributes that appear in figure 10 can be unfolded to show subconcepts that can be selected to extract more specific results. The system generates a new attribute for each of the classes (i.e. nodes) selected, and searches if the concepts that are below can be found in the Web pages of each city.

It is crucial to select the most appropriate nodes, at a level that the corresponding attributes that are generated are adequately populated. That means, for instance, that we cannot select categories that are leafs of the taxonomy, because at the lowest level the selected nodes will become a Boolean attribute in the resulting matrix. This type of attribute is not allowed in this system because they provide poor information for making clusters. So, the user must select some of the intermediate nodes of the hierarchy. For example, if *Christian Building* is selected, the resulting attribute will take as values any of its descendants (e.g. chapel, church...), but if the selected node is *Religious Building*, then the resulting attribute can take as values any of its descendants, like chapel, church, mosque, synagogue, Christian building, etc. Notice that the attributes are multi-valued because one city may have more than one Religious Building. The symbol ‘#’ is used to separate the values in each cell of the matrix.

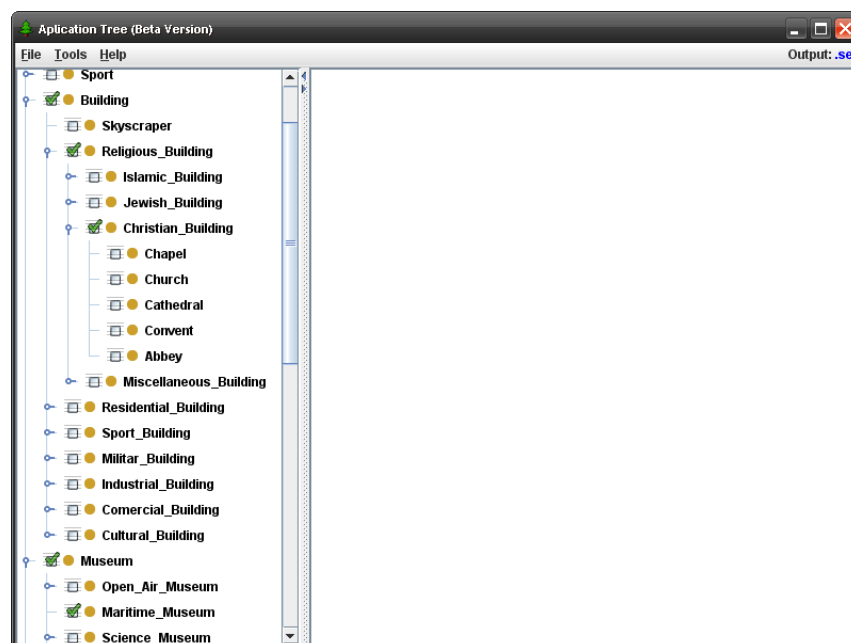
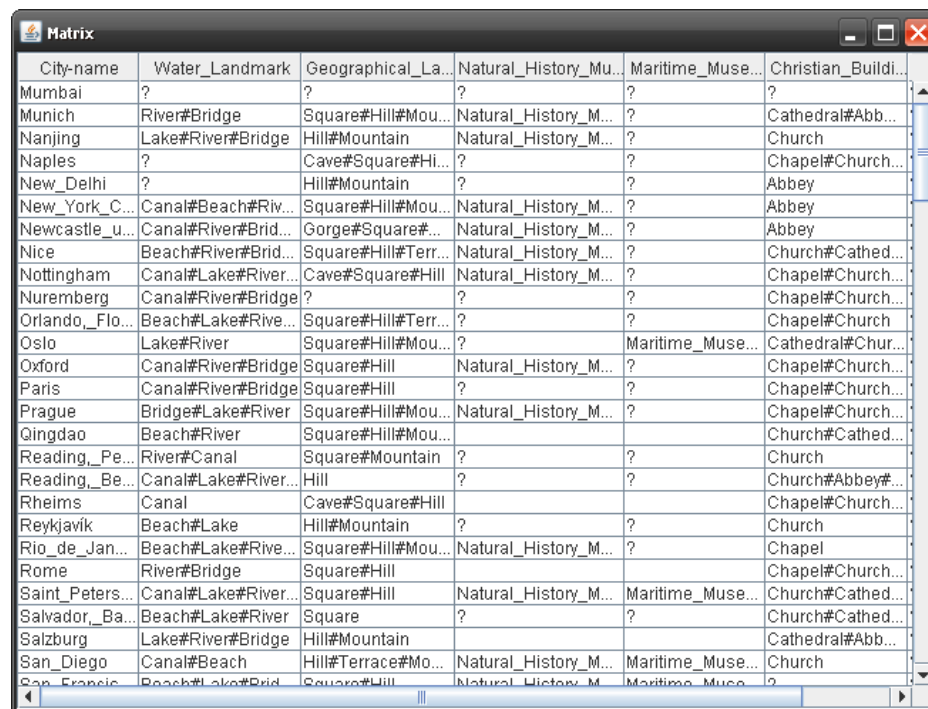


Figure 10: Expanded tree with some unfold categories

In case that the system is not able to find any evidence for a given attribute, the symbol ‘?’ is used. Notice that, in this case, the symbol is not exactly representing a missing value as normally understood, because the lack of information about one attribute is telling us that probably the city does not have any instance of this type. For example, in Figure 11, the automatic extraction system has not found any information about Maritime Museums in Munich, because certainly they do not exist. The data matrix construction by means of extraction processes is slanted by the *precision* and *recall* of each method used during the whole process (i.e., the natural language parser, the named entity detection heuristics, the inaccuracy of Web statistics and the relatedness measures). The *precision index* measures the number of correct values among all the values obtained. *Recall* is calculated by dividing the number of correct values by the total of values that could have been found. As explained in deliverable D2, for the purpose of the project, high precision is needed, to ensure that the values that we attach to some city are correct. High precision is achieved at a cost of reducing the recall. In this case, the symbol ‘?’ may appear in the data matrix because we have not been able to re-

trieve the information from the Web page. For example Oslo has a Natural History Museum, but this data has not been found by the system.



City-name	Water_Landmark	Geographical_La...	Natural_History_Mu...	Maritime_Muse...	Christian_Buildi...
Mumbai	?	?	?	?	?
Munich	River#Bridge	Square#Hill#Mou...	Natural_History_M...	?	Cathedral#Abb...
Nanjing	Lake#River#Bridge	Hill#Mountain	Natural_History_M...	?	Church
Naples	?	Cave#Square#Hi...	?	?	Chapel#Church...
New_Delhi	?	Hill#Mountain	?	?	Abbey
New_York_C...	Canal#Beach#Riv...	Square#Hill#Mou...	Natural_History_M...	?	Abbey
Newcastle_u...	Canal#River#Brid...	Gorge#Square#...	Natural_History_M...	?	Abbey
Nice	Beach#River#Brid...	Square#Hill#Terr...	Natural_History_M...	?	Church#Cathed...
Nottingham	Canal#Lake#River...	Cave#Square#Hill	Natural_History_M...	?	Chapel#Church...
Nuremberg	Canal#River#Bridge	?	?	?	Chapel#Church...
Orlando_Flo...	Beach#Lake#Rive...	Square#Hill#Terr...	?	?	Chapel#Church
Oslo	Lake#River	Square#Hill#Mou...	?	Maritime_Muse...	Cathedral#Chur...
Oxford	Canal#River#Bridge	Square#Hill	Natural_History_M...	?	Chapel#Church...
Paris	Canal#River#Bridge	Square#Hill	?	?	Chapel#Church...
Prague	Bridge#Lake#River	Square#Hill#Mou...	Natural_History_M...	?	Chapel#Church...
Qingdao	Beach#River	Square#Hill#Mou...			Church#Cathed...
Reading_Pe...	River#Canal	Square#Mountain	?	?	Church
Reading_Be...	Canal#Lake#River...	Hill	?	?	Church#Abbey#...
Rheims	Canal	Cave#Square#Hill			Chapel#Church...
Reykjavik	Beach#Lake	Hill#Mountain	?	?	Church
Rio_de_Jan...	Beach#Lake#Rive...	Square#Hill#Mou...	Natural_History_M...	?	Chapel
Rome	River#Bridge	Square#Hill			Chapel#Church...
Saint_Peters...	Canal#Lake#River...	Square#Hill	Natural_History_M...	Maritime_Muse...	Church#Cathed...
Salvador_Ba...	Beach#Lake#River	Square	?	?	Church#Cathed...
Salzburg	Lake#River#Bridge	Hill#Mountain			Cathedral#Abb...
San_Diego	Canal#Beach	Hill#Terrace#Mo...	Natural_History_M...	Maritime_Muse...	Church
San_Francis...	Beach#Lake#Brid...	Square#Hill	Natural_History_M...	Maritime_Muse...	?

Figure 11: Resulting matrix from the Tree extracting procedure

2.2.1 Selecting the semantic attributes

Since all the concepts in this tailored ontology are candidates to become attributes, we had to select which ones could better represent the city and facilitate the recommendation process for the users. In order to apply the semantic techniques explained in this project, we were not interested in generating Boolean attributes, so the classes at the leaves of the taxonomy were discarded. Then, we selected a subset of the intermediate concepts formed by:

- Aquatic_Sport
- Park
- Nature_Sport
- Martial_Art
- Residential_Building
- Christian_Building
- Water_Landmark
- Militar_Building
- Field_Sport
- Comercial_Building
- Geographical_Landmark
- Sport_Building
- Conmemorate_Landmark

- Cultural_Building
- Memorial_Landmark
- Miscellaneous_Building
- Tomb
- Museum

We studied the number of terms in the lists of each city for the different attributes, as well as, the number of cities with missing information (blanks). Figure 12 shows the average number of terms per attribute (counting only the ones that are not empty). We can see some attributes having a lot of concepts per city and some others practically empty.

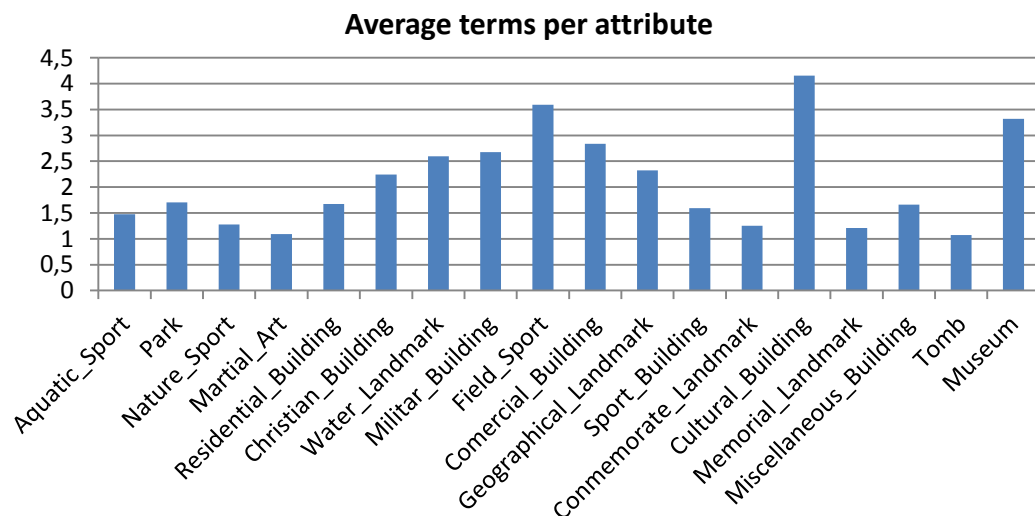


Figure 12: Average number of terms per attribute

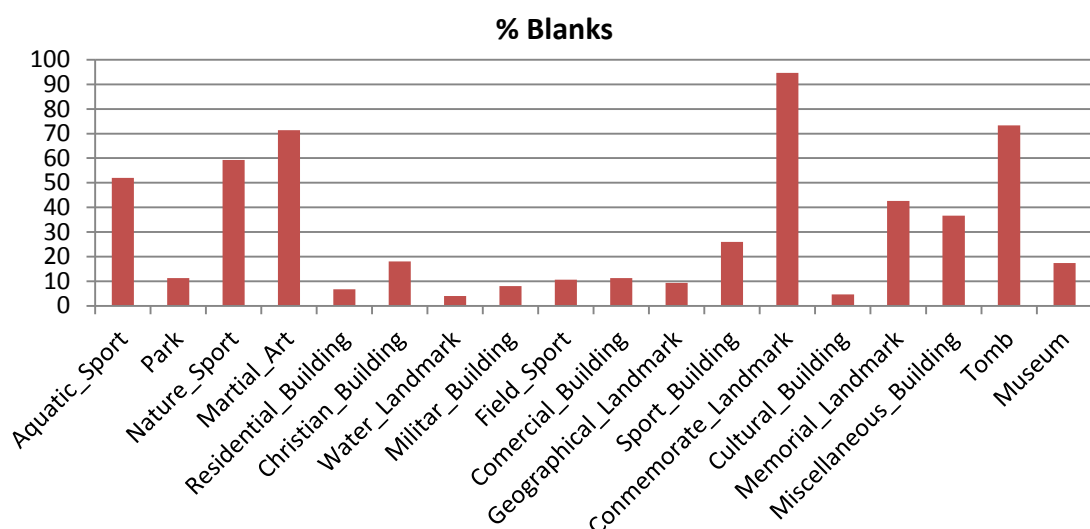


Figure 13: Percentage of blanks (?) per attribute

In Figure 13 we can observe that columns such as *Commemorate_Landmark* or *Tomb* are mostly empty. This is no desired because these attributes are not able to give any information to distinguish the cities.

After studying this distribution, we decided reduce the number of attributes columns, increasing the terms per attribute and avoiding to have attributes with no information. This is important because it affects the user, who will have to choose a prototype of city considering the meaning of the attributes. Hence, in order to not overwhelm the user, the lower the number of attributes, the better. For example, we constructed a new attributed with Water + Geographical landmarks, and we considered this an important issue for the final data matrix. The final set of attributes is:

- Aquatic + Nature Sports
- Other sports (Motor + Martial + Street + Field + Aerial + Dance + Competition)
- Religious_Building
- Other Buildings (Residential + Skyscraper + Industrial + Militar + Comercial + Sport)
- Museum
- Landmark (Water + Geographical)
- Other Landmark (Park + Column + Commemorate + Memorial + Tomb)
- Cultural_Building

City-name	Landmark (Water + Geographical)	Other Landmark (Park + Column + Commemorate + Memorial + Tomb)	Cultural_Building
Aberdeen	Beach#River#Square#Hill#Terrace	Park#Fountain	University#Theatre#Private_School
Abu_Dhabi	Beach	Park	School#Library
Agra	Lake#River	Statue#Mausoleum#Tomb	Public_University#Public_School#University#School
Amsterdam	Canal#Lake#River#Bridge#Polder#Square#Terrace	Nature_Reserve#Zoo	Theatre#School#Opera#University
Antwerp	River#Polder#Bridge#Hill#Mountain	Zoo#Statue#Tomb	Theatre#School#University
Atlanta	Lake#Bridge#Polder	Botanical_Garden#Zoo#Park#Statue	Public_University#Theatre#Public_School#Art_School
Bahrain	Bridge#River	Fountain	University
Bangkok	Canal#River#Beach#Bridge	Green_Zone#Forest_Park#Park#Statue	Theatre#School#University
Barcelona	Beach	Historic_Park#Zoo#Urban_Park#Forest_Park#Park#Statue#Crypt	Theatre#Opera#Forum#University#Ancient_Greek_Theatre
Bath_Somerset	River#Bridge#Beach#Stone_Bridge#Square#Hill#Terrace	Botanical_Garden#Park#Ionic_Column#Column#Statue	University#Theatre#School#Forum#Amphitheatre#Opera
Beijing	River#Stone_Bridge#Pedestrian_Bridge	Botanical_Garden#Zoo#Garden_Park#Obelisk#Mausoleum	School#Opera#University#Library
Benidorm	Beach	Park#Zoo	?
Berlin	Beach#River	Zoo#Botanical_Garden#Column#Fountain#Crypt	University#School#Opera#Theatre#Library
Bilbao	River#Bridge#Pedestrian_Bridge#Square#Hill#Mountain	Park	University#Theatre#Opera
Birmingham	Canal#River	Botanical_Garden#Nature_Reserve	University#Theatre#School#Library#Forum#Music_School
Boston	Canal#River#Square#Hill	Zoo#Park	Public_University#Theatre#Public_School#Opera#Lib
Bratislava	Lake#River#Bridge	Botanical_Garden#Zoo#Forest_Park#Statue#Pyramid	University#Theatre#Opera#Library#Business_School
Bregenz	Lake	?	Theatre#Opera
Brighton	Beach#River	Park#Mausoleum	University#Theatre#School
Bristol	Canal#River#Stone_Bridge#Bridge#Gorge#Square#Hill	Zoo#Nature_Reserve#Park#Statue#Megalithic	University#Theatre#School
Bruges	Canal#Bridge#Square	Park#Statue	Theatre#School#Opera#Forum#University
Budapest	River#Bridge#Lake	Park#Statue#Tomb	University#Opera#Library#Theatre
Buenos_Aires	Lake#River#Square	Botanical_Garden#Zoo#Park#Obelisk#Statue	University#Theatre#School#Opera#Library#Ancient_Greek_Theatre
Cairo	River#Beach#Bridge	Park#Pyramid	University#Theatre#Opera#Library#School
Cambridge	River#Hill	Park#Zoo#Fountain	University#Theatre#School#Library#Opera
Cancún	Canal#Beach#Lake#Bridge	Park	?
Cape_Town	Beach#Cave#Square#Hill#Mountain	Park#Statue	Theatre
Cardiff	Canal#Lake#River#Beach#Hill#Mountain	Nature_Reserve#Park#Pyramid#Megalithic	University#Theatre#Opera#Library#School
Chengdu	River#Bridge	Park#Statue	University#School#Theatre#Music_School#Opera
Chennai	Beach	Park	Theatre#Music_School#University
Chester	Canal#River#Bridge#Stone_Bridge#Hill	Zoo#Park#Statue#Crypt	Theatre#School#Roman_Amphitheatre#Opera#University
Chicago	Canal#Beach#Lake#River#Bridge#Square#Hill	Zoo#Botanical_Garden#Nature_Reserve#Park#Statue#Fountain	University#Theatre#School#Opera#Library#Technology
Chongqing	River#Bridge	Zoo#Statue	University#School
Copenhagen	Canal#Beach#Lake#Bridge#Square#Hill	Botanical_Garden#Zoo#Statue#Fountain#Tomb	University#Theatre#School#Opera#Forum#Music_School

Figure 14: Extract of the resulting matrix

Figure 14 shows an extract of three of those attributes. We can see now that the cities have a good number of terms on each of their attributes, having no attributes empty. With this recoding, the number of attributes has been reduced from 18 to 8.

The same analysis was performed in order to prove that now the average of terms per attribute (counting only the ones that are not empty) is acceptable and no column is almost empty. Figures 15 and 16 show the new results:

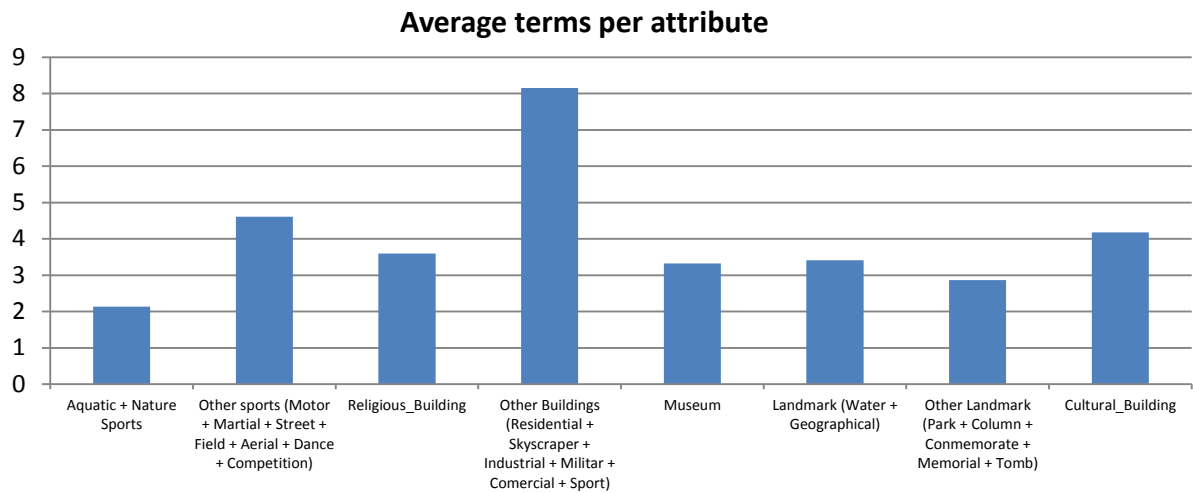


Figure 15: Average number of terms per attribute

We can notice that now almost every attribute has a mean of about 3 or 4 terms per city, which is completely desirable. The counterpart here is that the ‘Other Buildings’ column is a bit overpopulated. Some possible subdivisions were considered but finally we decided to maintain them as a single group.

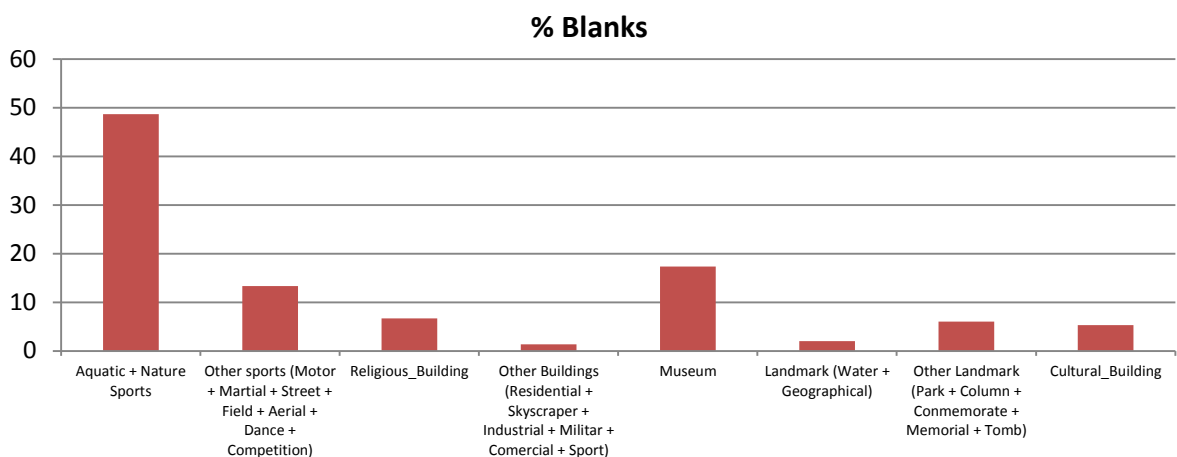


Figure 16: Percentage of blanks (?) per attribute

The percentage of blanks per column shows good results (below 10 – 20 %) in most of the cases. We have an exception in Aquatic + Nature Sports attribute, which has a too large number of missing values, but we decided to maintain it since it has a strong conceptual meaning which is lost if other sports are included in this group.

2.3 The DAMASK data matrix

After the study presented in this document, the data matrix that will be used in the prototype of the project has the following structure:

1. A set of 150 touristic cities distributed all over the world: Aberdeen, Abu Dhabi, Agra, Amsterdam, Antwerp, Atlanta, Bahrain, Bangkok, Barcelona, Bath (Somerset), Beijing, Benidorm, Berlin, Bil-

bao, Birmingham, Boston, Bratislava, Bregenz, Brighton, Bristol, Bruges, Budapest, Buenos Aires, Cairo, Cambridge, Cancun, Cape Town, Cardiff, Chengdu, Chennai, Chester, Chicago, Chongqing, Copenhagen, Dalian, Dijon, Dresden, Dubai, Dublin, Edinburgh, Florence, Florianopolis, Fortaleza, Foz do Iguaçu, Geneva, Genoa, Ghent, Glasgow, Goa, Gothenburg, Granada, Graz, Guangzhou, Guilin, Hamburg, Hangzhou, Havana, Heidelberg, Helsinki, Hong Kong, Honolulu, Houston, Innsbruck, Inverness, Istanbul, Jerusalem, Krakow, Kuala Lumpur, Kunming, Las Vegas (Nevada), Leeds, Linz, Lisbon, Liverpool, London, Los Angeles, Luxembourg, Lyon, Macau, Madrid, Malmö, Manchester, Marrakech, Marseille, Mecca, Melbourne, Mexico City, Miami, Milan, Monaco, Montreal, Moscow, Mumbai, Munich, Nanjing, Naples, New Delhi, New York City, Newcastle upon Tyne, Nice, Nottingham, Nuremberg, Orlando (Florida), Oslo, Oxford, Paris, Prague, Qingdao, Reading (Berkshire), Reading (Pennsylvania), Reykjavík, Rheims, Rio de Janeiro, Rome, Saint Petersburg, Salvador (Bahia), Salzburg, San Diego, San Francisco, San Jose (California), São Paulo, Seattle, Seoul, Seville, Shanghai, Shenzhen, Singapore, Stockholm, Suzhou, Sydney, Taipei, Tallinn, Tarragona, Tianjin, Tokyo, Toronto, Turku, Valencia, Spain, Varadero, Venice, Vienna, Warsaw, Washington D.C., Wuxi, Xiamen, Xi'an, York, Zaragoza, Zhuhai, Zürich.

2. A set of 12 heterogeneous attributes that describe the city with respect to different characteristics that may be useful for the tourist to find the most appropriate destination for his/her holidays. We can divide the attributes in two blocks:
 - a) Contextual information:
 - Population (Numerical): to choose small villages or large metropolis
 - Elevation (Numerical): geographical reference of the place (near the sea, in the mountain ...)
 - Continent code (Categorical): to restrict the trip to some part of the world
 - Climate (Categorical): general indicator about temperature, humidity
 - b) Leisure activities and touristic places:
 - Aquatic and Nature sports
 - Other sports (Motor, martial, street sports, field sports, aerial, dance, competition)
 - Religious buildings that can be visited
 - Cultural buildings
 - Other interesting buildings (Residential, skyscraper, industrial, military, commercial or sport related)
 - Museums
 - Landmarks related to Water and Geographical items
 - Other Landmarks (such as parks, commemorative monuments, memorial monuments, or tombs).

The data matrix is available at <http://deim.urv.cat/~itaka/CMS2/index.php>

3 References

- Bremner, C. (2007). Top 150 City Destinations: London Leads the Way. *Euromonitor International*. Retrieved from <http://blog.euromonitor.com/2007/10/top-150-city-destinations-london-leads-the-way.html>
- Vicient, C. (2009). Extracció basada en ontologies d' informació de destinacions turístiques a partir de la Wikipedia. Graduate Thesis on Computer Engineering, Universitat Rovira i Virgili.
- Vicient, C., Sánchez, D., & Moreno, A. (2011). Ontology-Based Feature Extraction. *2011 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)* (Vol. 3, pp. 189-192). doi:10.1109/WI-IAT.2011.199