Above the Surface

Organizing the digitally processed archaeological finds of the north/southline

Abhilash M. 14642859
Camilla Santos Andersen 11909099
Chris Atherton 14293285
Radu David 15241637
Danny de Vries 14495643
University of Amsterdam
Amsterdam, The Netherlands

ABSTRACT

Over 700,000 objects were found during the excavations of the North/Southline and digitally processed during the "Below the Surface" project. This study focuses on the organizing and visualization of these objects by creating a website that re-organizes the collection, an ontology, a data model, and the implementation of a machine learning model to enrich the dataset. Our results share the successful realization of these goals, making the collection more accessible for various user personas. Future work involves further enhancing the website with additional visualizations and improved data quality and completeness.

KEYWORDS

information organisation, data organisation, data visualization, archaeology, machine learning, ontology, data model, website

1 INTRODUCTION

From 2003 to 2012 excavations took place for the creation of the North/South metro line in Amsterdam. At Damrak and Rokin, which are unlikely archaeological sites due to being in the city center, archaeologists had a chance to physically access the riverbed. During these excavations in the Amstel over 700,000 objects were preserved which resulted in the archaeological collection called 'Below the Surface' [1] commissioned by the Municipality of Amsterdam.

The collection has a great variety of objects, from tools over centuries old to credit cards recently lost which makes the collection a rare source of urban history. All objects are digitally processed (e.g. photographed, labeled, metadata added) and displayed on a front-end website at *belowthesurface.amsterdam*. This website shows an overview of all the objects and a detail page with metadata of a particular object but no further categorization or classifications. This research aims to further organize this collection of objects with a focus on grouping the items by *functional properties*, determining *cultural relevance*, and researching *object relationships*.

2 RELATED WORK

2.1 Below the Surface

The starting point for re-organizing the collection, creating aggregated datasets, and exploring overview data is the dataset downloadable in *.csv* format on the Below the Surface project website. During the excavations around 700,000 objects were found, the dataset is a subset and contains around 20,000 objects that are digitally processed and rudimentary labeled. A separate data field description [2] file can be downloaded which further explains the controlled vocabulary used for the dataset.

2.2 User Personas

Defining the users is an important first step in developing a user-focused data system [**?**]. As our team did not have direct access to potential users, we developed three user personas to help guide our decision-making process for the website design.

2.2.1 Museum Curators. Stephanie is a female professional museum curator, aged between 35 and 45. She has a master's education in fine art and spends her days curating exhibitions for a local history museum in Amsterdam, the Netherlands. She speaks English and Dutch. Stephanie is good at networking and discovering new and interesting local history subjects. She desires to find new stories, themes, and artifacts to be displayed in the local history museum. Stephanie typically develops concepts by visiting other museums and exhibitions and building connections with other museum curators. Stephanie has limited time available to research what artifacts to include in an exhibit. As a result, she values quicker methods of finding exhibit items. Stephanie typically uses a desktop-based computer to access search engines, museum websites, and news websites. She also arranges face-to-face meetings.

2.2.2 Antique collectors. Gerald is a male amateur antique collector, aged between 45 and 60. He has a PhD in IT and is based in Europe. He speaks English and has a growing collection of antiques from the 17th Century. He has a desire to collect items that have some historic value, are from the 17th century, and have a unique visual appearance. He collects his antiques in his spare time, so time spent searching for antiques is at a premium. Gerald finds joy in hunting for antiques, especially using new ways to find antiques. He sees's optimisation of searching for antiques as a benefit. His main channels for searching are desktop research through search engines, auction houses, eBay and antique shops.

2.2.3 Archaeologists. Mădălina is a female archaeologist from Romania aged between 25 and 35. She is currently in a PostDoc position researching building materials of the 18th century in European nations that border the North Sea. She speaks English and German. Kate is interested in the functional use of certain artifacts from different time periods within the 18th century, specifically building materials and techniques. Kate wants to find materials that have a specific function related to buildings. Kate prefers to physically visit materials, archives and museums. She is involved in excavations located around the North Sea coast. She likes to spend time researching materials on her computer. However, if she can make the search and

[1] https://belowthesurface.amsterdam/en

[2] https://belowthesurface.amsterdam/en/pagina/publicaties-en-datasets

selection of objects for viewing more efficient, she can spend more time analyzing the findings that she does visit. Kate's preferred channels for discovery are museum archives, search engines, and excavation project websites.

2.3 Other websites

Due to the user personas preferring websites similar to museum or archive websites, we searched for several others in the field to give us an idea. We found 5 collections and museum archive websites that guided our design choices. We found that two sites had comparable layouts and search features found on the Below the surface website [3] [4]. The other three websites had additional features which enriched the search capabilities for users. Such as the ability to scroll down a collection based on the year [5], providing clear and simple visualizations corresponding to related data points [6], and providing information on the numbers of tags and albums which have been highlighted by other users [7].

As a result of the user personas, we decided that the website needed to be designed in a similar way to other websites from the field, give the ability to filter data based on object information, be a desktop-based website, and be available in English.

3 METHODOLOGY

3.1 Glushko's Questions on Information Organisation

3.1.1 What is the main purpose of the website? The Above the surface website focuses on providing an interactive platform for users to explore the archaeological finds from the excavation of Amsterdam's North-South metro line.

3.1.2 Who is it for? The website is for anyone interested in archaeology, history, and Amsterdam's heritage. The website has collections of items from different periods so can act as a gateway for collectors of ancient antique artifacts.

3.1.3 Why it is being organized? Users can use the website to learn about the archaeological finds from the excavation of Amsterdam's North-South metro line, view images of the artifacts, and read about their historical significance.

3.1.4 Where it is being organized? The website provides information about the project, a timeline of the excavation, and a gallery of images of the artifacts found during the excavation

3.1.5 How it is being organized? The website is organized into four main sections: Home, Project, Timeline, and Gallery. Each section has sub-sections that provide more detailed information.

3.2 Data Dictionary and Metadata

The Below the surface website provides users with a data dictionary as part of the available data downloads [**?**]. However, the information provided was not sufficient for the needs of the project. For example, the location of the sites related to the project code provided for each object. However, there was no description provided of which project code related to which site. It was only through research on the website that we were able to determine which site related to which project code. The metadata provided was left unchanged, as the enriched description in the data dictionary met the needs of the project.

3.3 Ontology

Development of the ontology consisted of 4 steps: Define scope, Enumerate important entities, Build the class hierarchy, and Specify the relationships between entities.

Five questions were posed to define and limit the scope of the ontology. These questions were based on the needs identified by the user personas.

- Where are the objects from?
- Where are the objects found?
- What materials are the objects made of?
- What functional use does the object have?
- What weight do the objects have?

The ontology was developed in WebProtégé [8]. As the ontology was developed the scope questions were used to ensure that additional aspects and classes were not investigated. The metadata fields in the data dictionary were used as a starting point, specifically those referring to the physical characteristics of the objects. The other key metadata field was the 'vondstnummer' or find number. This is a unique identifier for each of the objects in the database. It is created with a project code suffix. This project code also relates to the location where the object was found during excavations. The class names were also selected based on the existing schema defined on the web [9]. Where a pre-defined schema was not available, for example with metadata headers in the database, rdfs:label was used.

Once the key classes were listed, a hierarchy was created. For example, trench, level, feature, and selection were all hierarchically sorted under each respective project code, as they related to the differing levels of location accuracy within each site. Once the classes and hierarchies had been listed, relationships between each were defined. This was also at the point that defined schemas were linked to metadata headings found in the database. Relationships between Dutch metadata names in the database with their English descriptions were made. For example, the triple "put, rdfs:sameAs, trench. With the established hierarchy and relationships as a foundation, the ontology was created over a number of iterations. To test the ontology, the WebWOWL[10] web tool was used.

3.4 Data Model

The data model for the website is designed to manage the archaeological finds efficiently. The primary goals of this data model are:

- Normalize the data to minimize redundancy and maintain data integrity.
- Provide a unified structure for different material categories.
- Enable the storage of common attributes shared by all archaeological finds.

[3] https://www.vangoghmuseum.nl/nl/collectie?q=
[4] https://www.guggenheim.org/collection-online
[5] https://www.moma.org/calendar/exhibitions
[6] https://ourworldindata.org/how-many-animals-get-slaughtered-every-day
[7] https://www.fotozoektfamilie.nl
[8] https://webprotege.stanford.edu
[9] https://schema.org/
[10] https://service.tib.eu/webvowl/

The data model as shown in Figure 4 is structured into the following main parts:

3.4.1 Object. This table is central to the data organization for common attributes of all archaeological finds. It includes information such as the ObjectID, project code, category, subcategory, weight, object name, object part, number of fragments, and temporal data (start date, end date). The ObjectID field acts as the identifier for all objects and is the primary key.

3.4.2 Location. This table contains fields for location information (trench number, level number, feature number, and section number), as well as keywords and classification levels. A binary field indicates whether the find is listed on the website. ObjectID relates the location with Object as a foreign key.

3.4.3 Functional classification. This table contains the main functional usage of the object. It is divided into categories and levels of functional classification between Level 1 to Level 4.

3.4.4 Related Objects. The "Related Objects" table is a critical component of the data model, as it enables the establishment of relationships between different archaeological finds. This table serves as a bridge to link findings that may have connections between each other. In the context of the model, it enables the website to visualize associations between artifacts and materials within and across categories.

3.4.5 Material-Specific Tables. Each material category (e.g., Ceramics, Glass, Building Materials, etc.) is represented by a dedicated table linked to the Object Table through a foreign key relationship. These material-specific tables store attributes that are unique to each category, including dimensions, production details, and any other specialized data relevant to the category.

3.5 Technical Implementation of the website

The website is a custom front-end mostly using web standards and open-source software and libraries. The assumption is that it will mainly be used in a desktop environment by the users to explore on larger screens so the website is not fully responsive and thus not mobile-optimized.

3.5.1 Front-end frameworks. The web application is created with the open-source front-end framework Svelte [11] and UI framework SvelteKit which allows the application to be built-in interface components, each chart is rendered separately making it more efficient to add functionality (e.g. add datasets, render different chart types) in the future but also makes the website performant when more data and charts are added. Svelte can be downloaded as a module (package) from NPM [12] and uses the JavaScript back-end run-time Node.js [13]. For the charts the JavaScript charting library Chart.js [14] is integrated into the components which allows charts to be rendered in HTML5 Canvas without much configuration.

[11] https://svelte.dev
[12] https://www.npmjs.com
[13] https://nodejs.org/
[14] https://www.chartjs.org

3.5.2 Dataloading. The processed and transformed dataset was used as a primary data source of which subsets of the dataset in *.csv* format, roughly one per year chart and section, are converted to *.json*. These data files are loaded on-page-load of the browser. For this prototype, no back-end was set up and no database queries are being made. Any filter options and updating of the charts are more custom, it uses JavaScript utility functions to allow the data to be pre-processed and have only the data change.

3.6 Machine Learning Model

3.6.1 Data-set presentation. Bellow the surface provides a data-set[**?**] of all the objects resulting from the excavations. The data is provided in the form of a .csv file, with 139190 rows and 163 columns. Each row corresponds to an object. Describing each object is well outside of the scope of the purposes of this section, however, an explanation of the relevant columns is necessary.
The following columns are relevant for the purposes of the ML model:

- vondstnummer - represents a unique inventory number, in the form of a string. Every object has a vondstnummer. Example: "NZC1.00001MTL001".
- object - a description of the contents of the object. Example: "sieve residue"
- subcategorie - a categorization of the object material. Example: "metal: copper alloy"
- objectdeel - describes the object type morphologically (if it is part of a bigger object, a set, etc). Example: "fragment"
- vlak_min - Describes the minimum depth at which the object might have been found. Example: "-22.0"
- vlak_max - Describes the maximum depth at which the object might have been found. Example: "-22.01"
- begin_dat - The beginning of the interval of the estimated year of the object. Example: "1675.0"
- eind_dat - End of the interval of the estimated year of the object. Example: "1725.0"
- niveau1 - The category in which the object is placed. Example: "Communication & Exchange"

For the columns *object, subcategorie, objectdeel, vlak_min, vlak_max, begin_dat, eind_dat, niveau1* there are rows in the dataset in which one or more of these columns are blank.
The column *niveau1* can take the value of one of 12 pre-determined categories, as well as the value "Not classified". As previously mentioned, there are rows where this column is blank.

3.6.2 Objectives. Our objective is to create a machine-learning model that will complete the missing data for the "niveau1" column. This means that our model will predict a value in the niveau1 column, for the rows where currently that column is blank or has the value "Not classified". The prediction will be based on the values in the *object, subcategorie, objectdeel, vlak_min, vlak_max, begin_dat, eind_dat, niveau1* columns, which will act like input to the machine-learning model.

3.6.3 Deliverables. In order to achieve our objectives, the following files are delivered:

- process_dataset.py - a Python script that takes the original 163 column .csv files and consolidates them into another .csv

file that only contains the columns of interest. The name of this .csv file is "selected_dataset.csv"
- machine_learning.py - this Python script is the backbone of the machine-learning process. It is a script that does the following steps:
  - Loads the "selected_dataset.csv" dataset
  - Preprocess the data (completes the values with 0 or placeholders here they are blank", etc)
  - Converts text stings to vectors
  - Splits the data into unlabeled and labeled data based on the values in the "niveau1" column
  - Splits the labeled data using a training, testing, and validation split
  - Builds the ML model
  - Compiles the model
  - Trains the model
  - Tests the model
  - Predicts the values of niveau1 for the unlabelled data
  - Saves the updated dataset into a file named predicted_dataset.csv
- predicted_dataset.csv - a file containing the dataset in selected_dataset.csv, but with the column completed.

3.6.4 Model description. Given the problem, a relatively simple neural network model was chosen. It consists of 1 input layer, 2 hidden layers and one output layer. The input layer has 7 input neurons, corresponding to the following variables:

- subcategorie
- object
- objectdeel
- vlak_min
- vlak_max
- begin_dat
- eind_dat

For the text fields, a Text-to-Vector conversion was necessary. We used the word2vec_model for that. The data of the other columns was normalized. The hidden layers were composed of 10 neurons (dense layers) with a Relu activation function. The outputted has 12 neurons, with a softmax activation function, each neuron corresponding to one of the 12 category values niveau1 can take. The neural network was trained on the labeled data, using an 80 - 10 - 10 training - validation - testing split.

4 RESULTS

4.1 Website

The original website has a lot of individual objects and detailed metadata about physical properties. The aim of the website was mainly to *summarise* the collection to allow the before-mentioned users to explore the broad dataset and find interesting patterns. Then the user would be able to *create a subset* of the dataset based on physical properties and categories by further filtering the dataset. The design is based on the branding of the original Below the surface project and follows standard information architecture (e.g. primary navigation, form filters) and visual design principles.

4.1.1 Overview summary page. On loading of the website, the user is greeted with a 'summary' overview page (shown in Figure 1) with introductory text and three visualizations that display; *time of origin* in horizontally stacked bar charts, *functional properties* in a polar chart and *material usage* in a doughnut chart. The page has a primary top navigation to 'smooth-scroll' to each section and a 'create a subset' floating action button to navigate to the collection detail page.

4.1.2 Collection detail subset. Based on the categories of the summary page a user can create a subset of the collection on the 'create a subset' detail page (shown in Figure 2). Above the fold, the user has the option to use the same *category filters* from the summary page. Then a user additionally has the option to click on the filter dropdown to expose more form filter options using a checkbox with more granular options such as location found, size and dimensions and material technique. A 'download subset' button allows the user to download the generated subset in .csv format.

4.2 Machine Learning Model

Following training the neural network over 50 epochs, the accuracy of the neural network was tested. An accuracy of approximately 75% was obtained over the test data, which the neural network has never seen before. The accuracy is considered satisfactory for the given application. Furthermore, the neural network was used to classify the remaining unlabelled data, the results being deemed satisfactory.

4.3 Dataset

4.3.1 Initial Dataset. The initial dataset as provided in Below the surface website was structured as a flat model with each row representing an individual archaeological object. This flat structure included many columns, each corresponding to specific attributes associated with the objects. These attributes are stored in a wide range of material categories, such as ceramics, building materials, fauna, glass, and more. While the flat model provided an accessible format for data entry, it led to redundant storage of certain information, challenging in visualization, particularly common attributes that applied to all finds.

4.3.2 Data Transformation and Normalization. The dataset was provided in a flat model, which presented certain challenges in terms of data redundancy and efficient visualization. To address these issues and enhance the integrity of the data, a normalization process was done to fit the data into the relational model as described in the Data Model section.

4.4 Ontology and Data Dictionary

The initial data dictionary provided by the Below the surface website contained limited descriptions of the metadata. This posed a challenge to understanding the information in the flat-file dataset. Further work was performed to enrich the information so that it would be useful. For example, adding information about which geographic locations referred to which project code.

The ontology was created from scratch as none was supplied with the original website. Over 35 classes were created which covered aspects of the find-number, location, and various physical characteristics of the objects. It is now possible to determine which project code refers to which physical location. All of the finds were imported as individuals and assigned as a type of 'vondstnummer'

or 'find'. To import the find numbers into a turtle format correctly, the full stops needed to be removed from the find numbers. Having the full stop included resulted in the turtle file becoming unreadable. The code to assign the find number to the correct class in the ontology is provided.

5 CONCLUSION

In conclusion, the structuring of the Below the Surface collection has been a comprehensive effort. It incorporates data organization, user personas, development of an ontology, data modeling and the integration of a machine learning model. Collectively this has resulted in the implementation of a website that can visualize the goals of the project: grouping objects by functional properties, exploring cultural relevance and research object relationships of parts. The accessibility of the website is improved by the ability to filter through object subsets and user-centric design principles. The machine learning model that was used to complete missing and unclassified data, achieved a satisfactory accuracy of approximately 75%. These results indicate that the insights are a valuable contribution to the classification of the archaeological collection.

6 DISCUSSION

The main findings of the report show the significance of the comprehensive approach that was employed for the restructuring of the collection. The implementation of the website makes the data accessible to various user groups and highlights the relevance of archaeological findings. The machine learning model shows satisfactory results and therefore ensures completeness of the data. The initial dataset being a flat dump of data posed challenges for ontology, machine learning and visualization which we tackled by normalizing the data sets. The information like related objects was stored as plain text in the initial dataset which made it challenging to ensure the objects are related based on identifiers.

7 FUTURE WORK

Additional visualization features and filter options need to be added to the website in the next iteration of the website. Currently, the website only visualizes broad categories but there are more subcategories in the dataset that the user might be able to click on to get more detailed views and overlays. From a user experience point of view, further usability testing needs to be done to validate the User Experience (UX) and User Interface (UI) of the prototype website. Direct feedback from the users would further validate the workings of the website and uncover hidden interface and interaction problems. From a technical perspective, the website currently relies on exported data that is then loaded into the web visualization. A further enhancement for the website, and to make it more dynamic, is to have the dataset exposed through a Query-like API (e.g. GraphQL [15]) with an underlying 'headless' database (e.g. MongoDB [16]) in which the website is able to fetch up-to-date collection real-time data. This also allows for more performant and dynamic filter options on the 'create a subset page' through the use of a back-end.

[15] https://graphql.org/
[16] https://www.mongodb.com/

8 CONTRIBUTION OF AUTHORS

- *Abhilash:* developed the normalized data model from a flat data dump for the datasets and did further data analysis.
- *Camilla:* created aggregated datasets for the data visualization on the website.
- *Chris:* desk researched the user personas and created the data dictionary and ontology.
- *Radu:* built the neural network model, programmed it using Python and TensorFlow, prepared the data for being used by the neural network, trained validated and tested the neural network and predicted the remaining values for the missing niveau1 values in the dataset.
- *Danny:* programmed the website and visualizations (front-end development) as well as conceptualized the interface interactions (interaction design) and look and feel (visual design).

9 ACKNOWLEDGEMENTS

We thank lecturer Dr. V.O. Degeler (University of Amsterdam) for providing guidance and assistance during the project and MsC. A. Fleck (University of Amsterdam) who provided valuable feedback and answered our questions during the seminars which helped us further expand our research.

10 APPENDIX

10.1 Hosted source code

The source code of the web-based visualization, Python notebooks of the machine learning model and datasets used are hosted on GitHub using the MIT License. Under the *uvaio* username we have several code repositories:

(1) Notebooks: Source Code for the Jupyter Notebooks for data processing and machine learning. https://github.com/uvaio/notebooks
(2) Website: Source Code for the custom front-end website and interface. https://github.com/uvaio/website
(3) Datasets: The processed and modeled datasets, the machine learning completed categories, and the ontology. https://github.com/uvaio/datasets

10.2 Live version

A live demo version of the front-end website and visualization (desktop only) is hosted on Netlify and can be viewed using the following link https://uvaio.netlify.app

10.3 Website screenshots

Shown in Figures 1 and 2.

REFERENCES

[2]]belowdata [n. d.]. Bellow the surface dataset. https://statics.belowthesurface.amsterdam/downloadbare-datasets/Downloadtabel_EN.csv Accessed: 2023-10-09.
[2]]personas [n. d.]. Just-Right Personas: How to Choose the Scope of Your Personas. https://www.nngroup.com/articles/persona-scope/ Accessed: 2023-14-09.

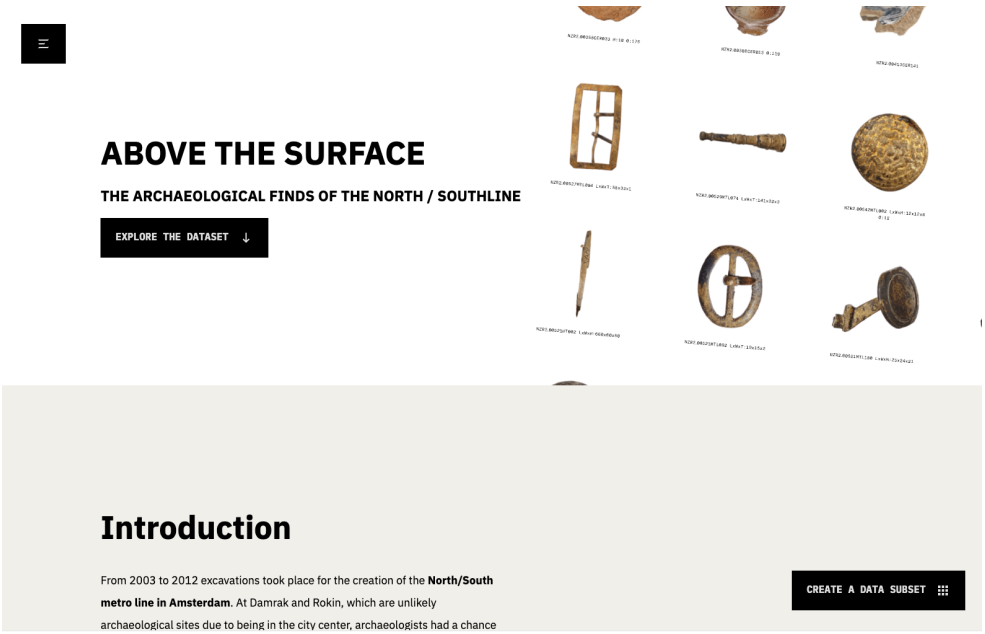

Figure 1: Screenshot of the 'overview landing page' of the website

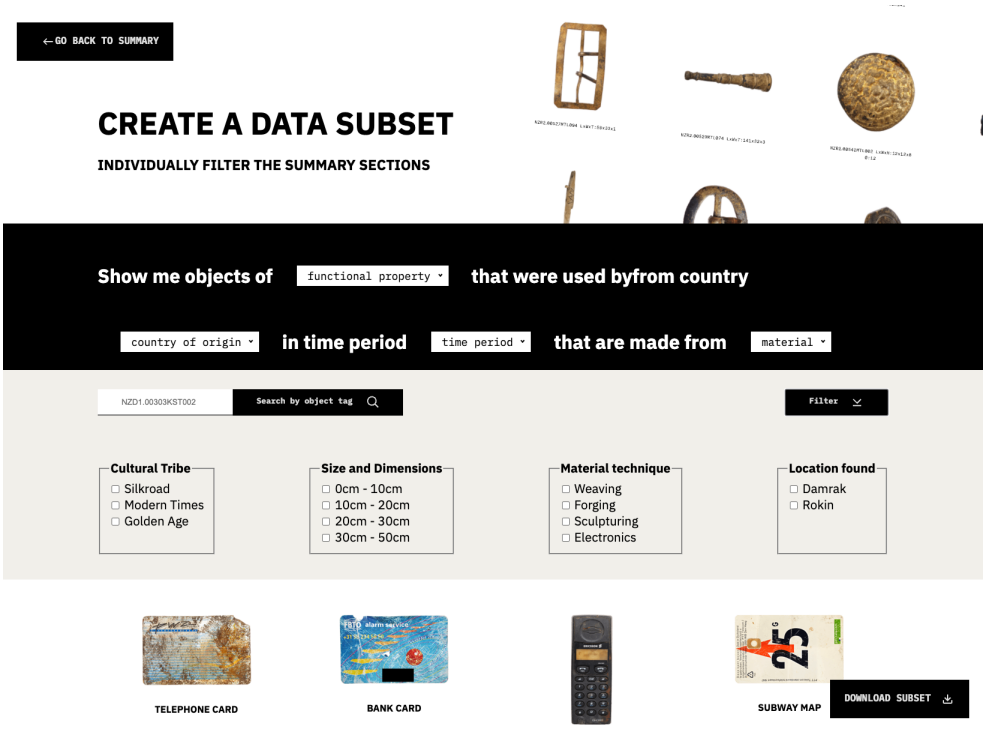

Figure 2: Screenshot of the detail 'create a subset' page of the website

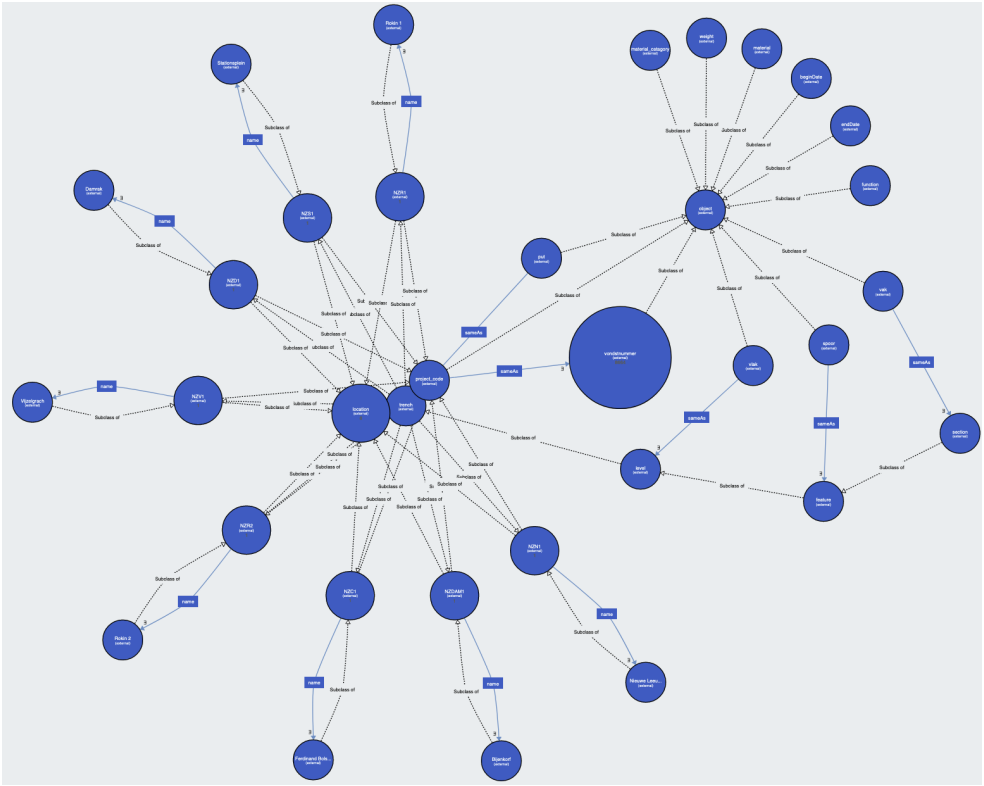

Figure 3: Graphical overview of the linked data within the ontology

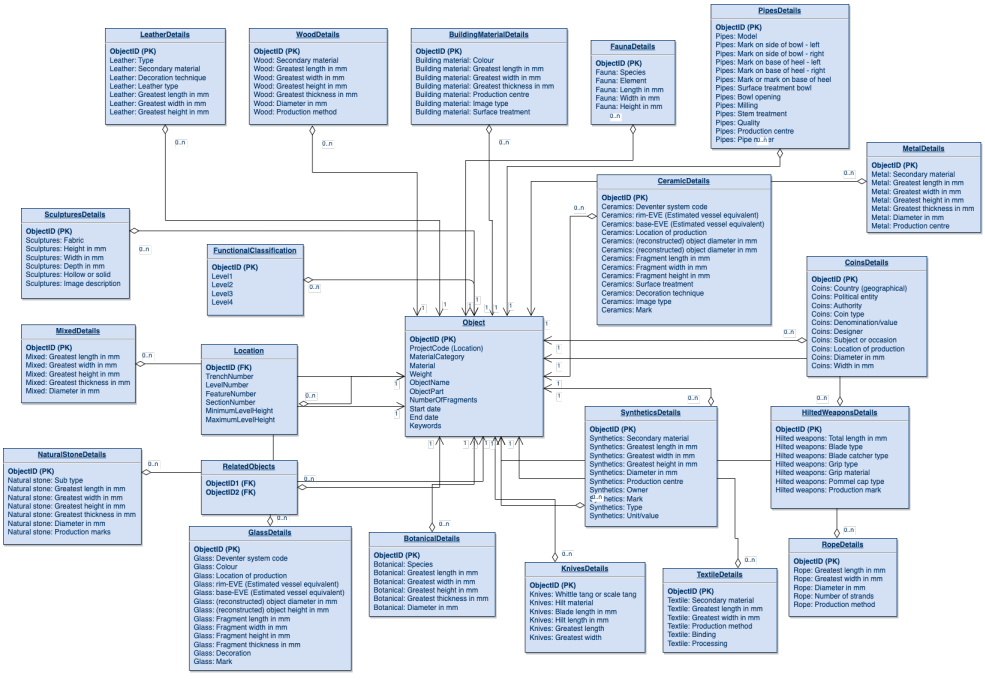

Figure 4: Graphical overview of the data model of the dataset