Project report

Project title: KAGGLE - Improving the quality of museums' data
Team members: Hanna Saskia Ambre & Pihla Järv
GitHub repository: https://github.com/hanna-ambre/ItDS_museum_data
Kaggle competition: https://www.kaggle.com/t/720d66660aef45a2af496c279664e382

Task 2. Business understanding (1 point)

Identifying business goals

Background
The Estonian museums' information system MuIS contains the collections of many museums in Estonia. The site allows everybody to see the objects from any institutions that have joined the system. There are almost 4 million objects but a lot of them aren't fully labeled and manually improving such a state would take years.

Business goals
Our project won't help any specific business financially but it may make the work of museums' workers and other people who work with museums' data easier.

The goal is to improve Estonian museums information system and therefore help to preserve Estonia's cultural heritage.

Business success criteria
Unlabelled objects are given correct type labels, thus improving the MuIS data's usability. We believe that around a 95% accuracy should be sufficient.

Assessing situation

Inventory of resources
In the Kaggle competition, there are 3 datasets given:

- a training dataset with descriptive variables and object types as columns
- a test dataset, which doesn't have the object types given, because we will be predicting them
- a dataset that contains a list of all possible types of museum objects

Software:

- jupyter notebook to write the code
- Kaggle to check the accuracy of our model on the test dataset

Requirements, assumptions, and constraints
The project will have to be completed before the Kaggle deadline, which is the 9th of December. The deadline for the project's poster is the 12th of December. The project's

GitHub repository has to be updated by adding there all the code we wrote and used for the final model.

Risks and contingencies
If a team member becomes sick and cannot attend the team meeting, then we can divide up the tasks and do them separately or host the meeting online.

If we become stuck with a problem we cannot solve on our own, then we will either ask for help on Campuswire or during the project consultation on the 6th of December.

Terminology
- Collection - a set of museum objects, tied together by a category assigned by the museum, e.g. ‘fotokogu’ (photo collection)
- Cross-validation - creating different validation sets out of the training set to predict accuracy
- Normalizing the string values - creating 1-hot vectors (creating new columns named after all possible string values and for every cell the value is either 1 or 0)

Costs and benefits
This does not apply to our project.

Defining data-mining goals

Data-mining goals
Our project's goal is to create a model to predict the type of the museum object based on its descriptive variables. We will submit the model’s predictions for the test dataset to the Kaggle competition. The model will be added to the GitHub repository. We will also submit a poster about our project and present it in the poster session at the end of the course.

Data-mining success criteria
In the Kaggle competition, our model’s accuracy will be compared to others who also chose this topic. Our code and presentation will be graded by the course teachers based on technical and presentational quality respectively.

Task 3. Data understanding (2 points)

Gathering data

Outline data requirements
We would need descriptive information for the museum objects as well as what type of objects they are. We will also need a list of possible object types.

We would need this data until the 12th of December.

Having the data in a csv-file would be nice but the data format is not that important as long as it is possible to read it into a DataFrame object.

Verify data availability
We have access to the data on the Kaggle competition and we can download the data from there. (https://www.kaggle.com/competitions/caps-in-museums-data-part-2/data).
We can load all the files into pandas DataFrame objects in Jupyter notebook.

Define selection criteria
We will use all the data given in the Kaggle competition. There are 3 datasets total: the training dataset, the test dataset and the list of possible types (see last task's subtask "Inventory of resources" for descriptions of these datasets). The extra dataset, sample_submission.csv, will be used to determine how the prediction result should be formatted.

Describing data

We have 3 datasets which are described in detail below. Our requirements for the data are fulfilled: train.csv has many objects with different descriptive fields as well as the type of object it is and we have a list of all possible object types. We also have test.csv for testing purposes. We believe that we have a sufficient amount of cases for training and testing.

unique_types.csv
- **Columns:** one column - "type"
- **Rows**: 55 rows (this means that there 55 different possible types of museum objects)

train.csv
- **Columns:** 38 columns. The names of the columns are: id, full_nr, name, ks, material, commentary, event_type, location, start, end, before_Christ, country_and_unit, participants_role, participant, parish, text, class, technique, parameter, unit, value, museum_abbr, musealia_mark, musealia_seria_nr, musealia_queue_nr, musealia_additional_nr, collection_mark, collection_queue_nr, collection_additional_nr, element_count, legend, is_original, initial_info, damages, state, color, additional_text, type.
- **Rows:** 14 000 rows of museum objects

test.csv
- **Columns:** 37 columns. The columns are the same as in train.csv except for the last column "type" which is missing because this is the one we have to predict.
- **Rows:** 6000 rows of museum objects

Exploring data

Train dataset columns:

Column	Description	Amount of unique values (excluding null)	Amount of total non-null values	Object data type
id	Unique id of museum object in the table	14000	14000	int64

full_nr	Unique id that also shows which museum and collection the object is from. Because the information is already separated in other columns, we will use those instead of this one.	9012	9012	object
name	Short description of the object (may contain type, author, year, etc.)	7595	9012	object
ks	A part of the full_nr value	2828	8816	float64
material	Material of the object	108	9012	object
commentary	Additional comment about the object (might be one word or a more detailed description)	197	440	object
event_type	Event/activity that the object is related to	477	9302	object
location	City, street, address, etc.	333	1239	object
start	Year or date (might contain the name of the month or might contain only numbers and a dot)	1653	5920	object
end	Year or date (might contain the name of the month or might contain only numbers and a dot)	301	1247	object
before_Christ	Only non-null value is ‘ei’ (‘no’)	1	5435	object
country_and_unit	Country and sometimes city as well	158	9302	object
participants_role	Shows what role the participant had	45	6234	object
participant	Person’s or organization’s name	1848	6234	object
parish	Municipalities or towns in Estonia. Because of low non-null occurrences, probably will not use this column	11	14	object

text	Commentary about the object or who made it	184	229	object
class	Shows if someone's name is on the object	2	229	object
technique	Related to making the object or what type it is	96	5691	object
parameter	Sets up what context the unit and value are used in	23	6571	object
unit	A unit of measurement or size of the object (e.g. '24 x 36 mm' for a photo)	21	6571	object
value	Shows how many units there are in the object	760	6571	object
museum_abbr	Museum that the object belongs to	2	13998	object
musealia_mark	'_' or null depending on if the full_nr has a '_' after the museum_abbr part or not	1	10356	object
musealia_seria_nr	Part of full_nr, after the musealia_mark bit	2243	10010	float64
musealia_queue_nr	Part of full_nr, after the musealia_seria_nr bit	733	2209	float64
musealia_additional_nr	Part of full_nr, before the collection_mark bit, might be a letter, date or comment	96	483	object
collection_mark	Collection that the object belongs to, museum-specific	33	13998	object
collection_queue_nr	Part of full_nr, after the ks bit	1204	10440	float64
collection_additional_nr	Part of full_nr, last part, might be a letter, date or comment	3463	7506	object
element_count	Usually a '1', more if the object is made up of many parts. Because of low amount of unique	6	14000	float64

	occurrences, probably will not use this column			
legend	Where or how the object was acquired. Might contain a small story.	569	3098	object
is_original	1.0 or 0.0 (respectively meaning yes or no)	2	11779	float64
initial_info	Some code (for example TB200222) or a free text description	1062	1613	object
damages	Description of damages of the object as free text	594	1160	object
state	“Hea”, “rahuldav”, “halb”, “väga halb” or “määramata”	5	14000	object
color	Color name (or “black-and-white” or just “colored” in estonian)	43	451	object
additional_text	Additional notes about the object	2248	3062	object
type	Type of the object. This is the value we need to predict in testset.	55	14000	object

The train.csv dataset has the same columns, excluding the ‘type’ column, and with different unique and non-null values.

Verifying data quality

We can access the data. The only problem might be that some of the objects have very few descriptive fields labeled as non-null values. We can find out if it is a relevant problem only after we have tried to create and test the model.

Task 4. Planning your project (0.5 points)

List of tasks

Task	**Hours (Hanna and Pihla)**
Analyze the data and decide which columns to use	H: 6 P: 6
Normalize the values in columns chosen for creating the model	H: 5 P: 5

Write the code to do cross-validation	H: 0 P: 2
Find the best model for the task using cross-validation	H: 3 P: 1
Tune the hyperparameters using cross-validation	H: 1 P: 2
Train the model on all training data	H: 0.5 P: 0.5
Predict types for the test data and submit the answers to Kaggle	H: 0.5 P: 0.5
Check if github repo is up-to-date	H: 0.2 P: 0
Make a poster	H: 6 P: 6

Methods, tools and comments

- Comment: The tasks should initially be done in the order they are in the table. We will repeat the tasks if necessary until we get a result (in the Kaggle competition) that we are satisfied with. Since we don't know how many times we will have to repeat the steps, we are not sure how accurate the estimated hours are.
- Methods: normalizing values, cross-validation, hyper-parameter tuning
- Tools: Jupyter notebook, GitHub, Kaggle (for accuracy on test set)