August 2024

How to Embed Large But Incomplete Knowledge Graphs in the Culture Heritage Sector

Lessons Learned From Odeuropa

Pasquale LISENA [1], Thibault EHRHART and Raphaël TRONCY
EURECOM, Sophia Antipolis, France
ORCiD ID: Pasquale Lisena https://orcid.org/0000-0003-3094-5585, Thibault Ehrhart https://orcid.org/0000-0003-1377-8279, Raphaël Troncy https://orcid.org/0000-0003-0457-1436

Abstract. Classic machine learning tasks, such as clustering and link prediction, can be applied to Knowledge Graphs making used of the so-called graph embeddings, mathematical vector representations of the nodes present within the graph structure. Often, the data structure of Knowledge Graphs in Digital Humanities is at the same time versatile and complex, challenging the machine learning tasks. In this work, we compare algorithms on two different subgraphs extracted from a large knowledge graph developed in the cultural heritage domain: one is randomly selected while the other is built to maximise the triple density. Using the European Olfactory Knowledge Graph (EOKG) as use-case, we show that embedding dense subgraph can improve the performances of state-of-art algorithms.

Keywords. Knowledge Graphs, Embeddings, Semantic Web, Machine Learning, Clustering, Link Prediction

1. Introduction

Methods for machine learning have gained significant attention in recent years in the semantic web research community. In the context of Knowledge Graphs (KG), machine learning can either be used for refining the knowledge graph itself (which involves predicting new edges and/or identifying erroneous edges) or for downstream tasks: training models on the KG data are then used for classification, recommendation, regression, etc. in an application domain [1]. For feeding machine learning algorithms, the data in the graph must be encoded in a suitable vectorial format, commonly called *graph embedding* [2]. A number of techniques have been proposed: translational models that interpret edge labels as transformations from subject nodes to object nodes [3], tensor decomposition models [4], neural models and language models [5,6]. These techniques have shown good performances on several machine learning tasks, involving both general knowledge and domain-specific datasets.

[1]Corresponding Author: pasquale.lisena@eurecom.fr.

When applied to fields such Digital Humanities and Cultural Heritage, multiple challenges arise [7]. In particular, the domain intrinsically exhibits knowledge that can be structurally complex [8,9], imprecise and uncertain [10], interpretable and subjective [11]. These qualities may lead to hypothesis that, in the Digital Humanities field, the attention to the data should be higher. Notably, data sparsity plays a role in the generation of graph embeddings, with several works demonstrating an high correlation between the graph density and the embedding performances [12,13,14]. We aim to study if an optimal selection of a cultural knowledge graph exists for training successful graph-based machine learning tasks.

We will use the European Olfactory Knowledge Graph (EOKG) [15] as a use case for this research. The EOKG has been realised in the context of the Odeuropa project[2] for representing smells and olfactory experiences. Using the EOKG as source knowledge graph, we want to use graph-based AI technologies – specifically, graph embeddings – to infer and extract new knowledge from the existing graph. We target two specific tasks:

- We investigate strategies for grouping related odours, measuring the similarity between two intangible items. Machine learning approaches relying on graph embeddings are implemented to cluster items;
- We try to predict missing links in the graph. More precisely, we train the machine to predict the smell source given the description of the smell experience and its surrounding context, or vice-versa, to generate context-specific descriptions for a given odour.

Our experiments demonstrate that the use of a dense graph entail an improvement of the performances for both tasks, making it a useful strategy in real-world applications in the Cultural Heritage domain.

The remainder of this paper is organised as follows. We outline some related work in Section 2. We provide an overview of the EOKG we use in Section 3. We describe our method in Section 4, which is then used in the two tasks: clustering (Section 5) and link prediction (Section 6). Finally, we conclude and outline some future work in Section 7.

2. Related Work

2.1. Graph Embeddings

Research about graph embeddings is quite vast: several methods have been proposed, which have been grouped into different taxonomies, with some overlap [2,16,17].

Random-walk based methods are particularly suitable for approximating properties such as proximity, similarity, and various structural characteristics of nodes and subgraphs [18]. The core idea of this family of techniques consists in the simulation of the movement of a "walker" through the graph structure. At each step, the algorithm randomly chooses one of the outgoing links of the current node, and follows it to the next node. The full path (walk) consists on a series of nodes. These walks serve as analogous structures to sentences in a corpus. Next, *word2vec*) is often used on these walks yielding a number of methods such as *rdf2vec* [5], *node2vec* [19], or *entity2vec* [6]. Entities (or words in the word embedding scenario) that appear in the same random walks (or

[2] `https://odeuropa.eu/`

sentences) are considered semantically close, and so have higher possibilities to appear close also in the embedding space.

TransE [3] is the most representative method of the translation-based family. This algorithm assigns vectors to both entities (nodes) and predicates (edges), with the idea that for each triple (*subject*, *predicate*, *object*), the object vector O corresponds to the translation of the subject vector S along the predicate vector P, that is $O = S + P$. In the training phase, the algorithm tries to minimise the distance between O and the translation $T = S + P$. This algorithm is widely used in heterogeneous graphs [17], in particular when the number of different properties is important.

Tensor-factorisation based models include techniques such as ComplEx [20] and DistMult [21]. Similarly to the translation models, these methods embeds nodes and edges in the same real dimensional space. What is different is the translating function of the object, being the dot product of the subject and the predicate.

Finally, rotation-based models replace the translation of TransE with a rotation in the space. RotatE is a well known example [22] of this family.

2.2. Embeddings in Cultural Heritage

There are several application of KG embedding in the specific field of Digital Humanities and Cultural Heritage. In [23,24], embeddings are trained on cultural KGs, demonstrating their suitability for link prediction and similarity computation. In [25], missing *Place* nodes are inferred from the entities in the graphs in which the information was missing, using a learning function over KG embeddings.

The quality of clusters is used as a measure of the embeddings quality in [9,26], while the graph embedding clustering is used for other tasks in [27], where a similarity metric is used for entity alignment in the history domain.

3. Dataset Overview

3.1. The European Olfactory Knowledge Graph (EOKG)

Odeuropa is a European project (2021-2023) which integrates expertise in sensory mining, knowledge representation, computational linguistics, (art) history and heritage science with the objectives of contributing to the topic of olfactory heritage. One of the project goals is to develop novel methods to collect, model and publish and disseminate information about smells from digital text and image collections, while also working with the material presence of odours.

The European Olfactory Knowledge Graph (EOKG) that results from these efforts is one of the main outcomes of Odeuropa. It consist of a large dataset of olfactory experience information extracted from textual and figurative resources using domain-specific techniques [28,29] and organised according to the Odeuropa Ontology [15] which makes use of semantic web technologies. This ontology represents a Smell as an entity that exists at a precise time and space, connected to a Smell Emission and an Olfactory Experience. These are linked respectively to objective (smell source, odor carrier, place, etc.) and subjective characteristics (quality, emotion, gestures, etc.), as depicted in Figure 1. These characteristics are normally represented by terms or *Concepts* from controlled vocabularies.

One of the strong points of the EOKG is the harmonisation in a single dataset of 21 data sources covering several centuries from 1600 to 1920, involving both images and textual resources in 6 European languages, as reported in Table 1. Figure 2 shows a visual representation of an excerpt of the EOKG. The KG can be accessed in multiple ways, namely querying a SPARQL endpoint[3], calling the grlc-based API[4] [30] or accessing the dumps on the open-source repository[5].

Figure 1. The core of the Odeuropa ontology

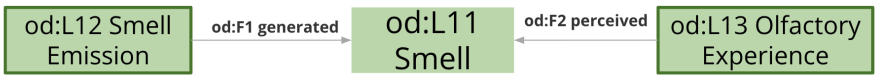
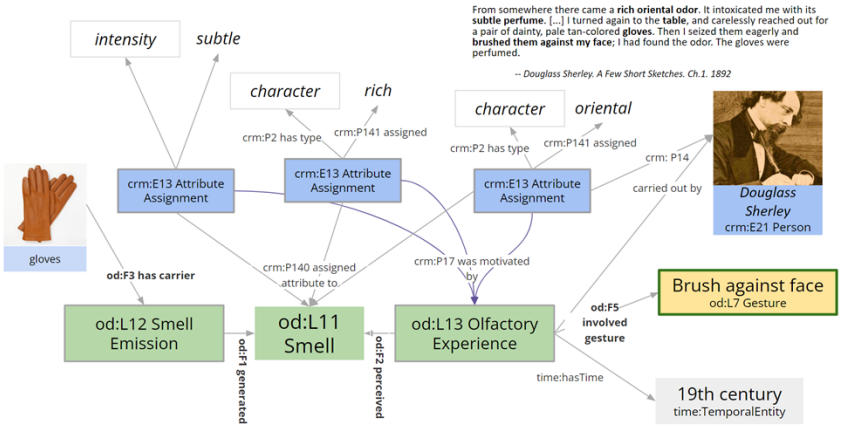


Figure 2. A visual representation of the EOKG. In this example, *Douglass Sherley* is the person who experiences a smell emitted by *gloves* and assigned characteristics such as being *intense*, *rich* and *oriental* in the 19th century.



The EOKG contains over 400 million statements (or triples), referring to over 2 million unique smells. For each of these smells, the dataset may contain a set of characteristics which are interlinked, whenever possible, with controlled vocabularies. The data population involved also some post-processing including parsing of time, parsing and interlinking of places to GeoNames, the interlinking of several entities with the controlled vocabularies.

3.2. The Role of Controlled Vocabularies

Together with a set of multi-language alternative labels which allow to disambiguate terms towards the same identifier (URI), the Odeuropa controlled vocabularies are organized as a hierarchical structure formalized in SKOS [32]. In addition, some links between vocabularies are instantiated for:

[3] `http://data.odeuropa.eu`
[4] `https://grlc.eurecom.fr/api/Odeuropa/kg-api/`
[5] `https://github.com/Odeuropa/knowledge-graph`

Table 1. Count of smell instances per data source

Graph	N. of smell instances	Type	Language
Odeuropa Benchmark [31]	7,125	Text	All
British Library	328,828	Text	EN
Medical Heritage	898,458	Text	EN
Gallica	811,633	Text	FR
Gutenberg	36,766	Text	EN, IT
Digitale Bibliotheek voor de Nederlandse Letteren	39,642	Text	NL
Digitalna knjižnica Slovenije (Dlib)	22,270	Text	SL
LiberLiber	30,228	Text	IT
Deutsches Textarchiv (DTA)	14,776	Text	DE
Wikisource	110,917	Text	EN, IT
Eighteenth Century Collections Online (ECCO)	8,435	Text	EN
Early English Books Online (EEBO)	81,038	Text	EN
London's Pulse: Medical Officer of Health reports	6,562	Text	EN
Royal Society Corpus	3,627	Text	EN
Old Bayley Corpus	1,170	Text	EN
Bibliothèque Bleue de Troyes	495	Text	FR
Grimm's Correspondance littéraire	64	Text	FR
ODOR dataset [29]	24,351	Image	-
Rijksmuseum	46	Image	-
Europeana	17,530	Image	-
NUK	1,752	Image	-

- connecting elements which refer to the same thing, linked using `skos:exactMatch`;
- connecting elements which are somehow related, linked using `skos:related`. For example, *tobacco* is related to *pipe*, *stable* is related to *cow*, etc.

To better appreciate the represented hierarchy, for the *olfactory objects* vocabulary, we present in Table 2 the top-level concepts with the relative number of narrower concepts. The full list of olfactory objects can be extracted from the EOKG using the following SPARQL query:[6]

```
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
SELECT DISTINCT ?top ?top_label ?concept ?concept_label
WHERE {
    ?top skos:prefLabel ?top_label;
       skos:topConceptOf
          <http://data.odeuropa.eu/vocabulary/olfactory-objects> ;
       skos:narrower* ?concept.
    ?concept skos:prefLabel ?concept_label .
    FILTER (lang(?concept_label) = "en")
    FILTER (lang(?top_label) = "en")
}
ORDER BY ?top ?top_label
```

[6]Quickly accessible at `https://tinyurl.com/bdft538n`

Table 2. The top concepts of the olfactory objects vocabulary, with the number of relative narrower concepts (at any depth). The sum of each individual count of narrower concepts may differ from the total dimension of the vocabulary, since some concepts belonging to two different categories are counted twice. The namespace `olfactory-objects:` is equivalent to `http://data.odeuropa.eu/vocabulary/olfactory-objects/`.

URI	label	# narrower concepts
olfactory-objects:560	Flora	178
olfactory-objects:405	Food	176
olfactory-objects:554	Artefact	148
olfactory-objects:555	Being	84
olfactory-objects:558	Fragrance / Cosmetic	56
olfactory-objects:533	Body	38
olfactory-objects:562	Element	32
olfactory-objects:559	Nature	27
olfactory-objects:563	Abstract	19
olfactory-objects:616	Matter	8
olfactory-objects:564	Religion	6
olfactory-objects:631	Product	4
olfactory-objects:615	Fumes	2

These vocabularies can be easily browsed at `https://vocab.odeuropa.eu/`. In Table 3, we report on the number of terms successfully interlinked with the controlled vocabularies. This gives an idea of the size, complexity and density of the graph.

Table 3. Statistics of the interlinking of controlled vocabularies for the most important properties in the EOKG

Graph element	**Nb. of disambiguated mentions**	**Relevant vocabularies**	**Nb. of concepts**
Gestures	3,066	Olfactory Gestures	36
Agents	8,868	Noses	14
Places	79,106	Fragrant Spaces	137
Evoked odorants	64,671	Olfactory Objects	682
Carriers	86,707		
Odor Sources	678,249		
Qualities	509,886	Smell Classifications, Hedonic, Intensity	1,824
Emotions	158,005	Plutchik + Odeuropa extension	32

4. Methodology

4.1. Pre-processing steps

To generate embeddings from the EOKG, we collected a list of relevant properties, consisting of all the connections between the smell objects and the vocabularies in Table 3. We collected all pairs of instances involving these properties locally, in the edgelist format.

We decided to include only the data extracted by textual resources and exclude those coming from the annotation of images. The reason for this choice stems from the nature

of those images: for example, still life paintings are widely represented and depict together a series of objects – from pipes to fruits, from fishes to wine – without having a clear meaning of their olfactory similarity or relatedness.

Given the size of the graph, any embedding strategy involving the totality of the data would be expensive and would require a computation time of several weeks or months to be completed. This effort may be repeated if the graph evolves over the time. For this reason, we decided to use a subset of the graph for training our machine learning models. However, following the research question mentioned in Section 1, we aim to investigate for an optimal subgraph, rather than randomly sampling a number N of entities. Considering the considerable effect of data sparsity in graph-based methods [33,34], we aim to to identify a dense subgraph for our purposes.

Given a directed graph $G(V,E)$ consisting of the vertex set V and the edge set E, we apply the *density* definition for directed graph from [35]:

$$d_G = \frac{|E|}{|V|(|V|-1)} \tag{1}$$

In addition to the density, we include a more *specific completeness* metric, which is measuring how much the studied entities (in our case, the smells) are interconnected with the rest of the properties in the graph. Getting freely inspired by the definition of completeness from graph theory[7], we consider our graph complete for the domain if all studied entities are the subject of the object for all relevant properties. For a specific entity type T, the relevant properties set P_T consists of all properties which have that specific entity type as property range or property domain. We define the relevant statement S_T a triple (s,p,o) in which $p \in P_T$. This *specific completeness* is measured as:

$$c_T = \frac{|S_T|}{|V| * |P_T|} \tag{2}$$

Please note that if the *density* measure is agnostic with respect to the schema, for computing the *completeness* one should have the knowledge of the relation between classes and properties present in the data model.

Given the particular structure of the schema, the relevant information is not directly linked to the Smell entity, but to its Smell Emission and Olfactory Experience. This ensures high flexibility in representation, allowing to have multiple Olfactory Experiences for the same Smell. However, looking at the real data of the EOKG, each Smell is linked to a single Experience, creating an unnecessary hop in the connection between the entity and the relevant informative values. Because of this, we made a series of selection of the graph, with the progression of changes tracked in Table 4, together with the main metrics. We created a new version of the KG called *Cleaned Graph* in which the core composed of Smell, Smell Emission, and Olfactory Experience is represented as a single atomic entity, which we will call *Smell* for simplicity. In addition, all edges not linked to a smell are removed from the graph, with the exception of the vocabulary hierarchy which is always kept in all subgraphs. As a consequence, the number of involved predicates slightly decrease.

Starting from this *Cleaned Graph*, we realise the *Dense Graph* by selecting only the smell entities having at least 4 links (ingoing or outgoing). Respect to the original graph,

[7] *"A simple graph in which every pair of two vertices is adjacent is called a complete graph"* [36]

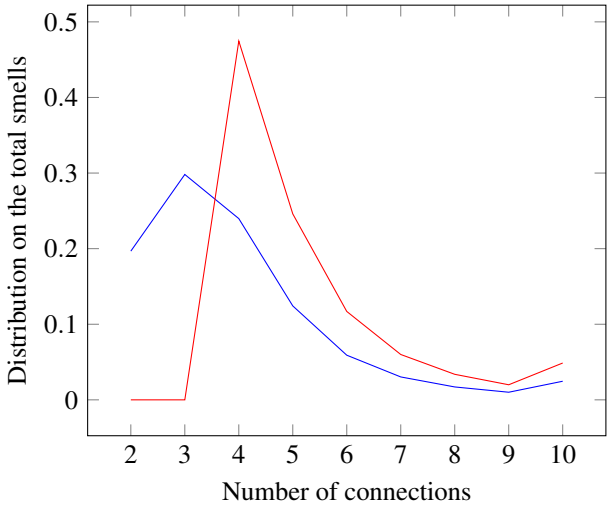
this operation multiplied the density of the graph by over 17 times, and increased the specific completeness of 0.14.

From both the *Original Graph* and the *Dense Graph*, we extract a selection of 10,000 smells to reduce the computational complexity of the link prediction task. These two selections have density metrics quite similar and very different completeness, as it is possible to see in Table 4. The distribution of smells with different numbers of connections changes between graph, with the dense graph having the 47% of smells with 4 connections. The distribution is reported in Figure 3.

Table 4. The composition of each version of the dataset following the different preprocessing steps

Graph Version	Nb entities	Nb predicates	Nb edges	Density	Completeness
Original Graph	5,182,119	23	8,434,414	$0.31e^{-6}$	0.24
Original Graph 10k	9,071	22	10,795	$131.21e^{-6}$	0.048
Cleaned Graph	2,232,870	15	8,179,561	$1.64e^{-6}$	0.27
Dense Graph	823,265	15	3,752,435	$5.54e^{-6}$	0.37
Dense Graph 10k	18,208	13	61,343	$185.04e^{-6}$	0.50

Figure 3. Distribution of connections (incoming or outgoing) for the cleaned (blue) and the dense (red) graphs, relatively to the total number of smells. The reduced version of 10k smells follows similar distributions.



4.2. Embedding Strategies

We experiment with three embedding algorithms:

- RDF2vec [5], that applies Random Walks [18] and Word2Vec [37] to the graph;
- TransE [3], as a representative of a geometric embedding strategy [38];
- DistMult [21], as a representative of a semantic embedding strategy (it quantifies the likelihood of a triple to belong to the KG through a multiplicative score function).

For RDF2vec, we use the PyRDF2vec implementation [39], while for TransE and DistMult, we use the PyKEEN library [40]. The resulting embeddings include in the same embedding space both the concepts coming from the vocabularies and the smell entities, even if the experiments reported later in this paper will focus on particular subset of them. The generated embeddings have been tested in two different tasks, clustering and link prediction, that we detail in the next sections.

5. Clustering

In this section, we investigate strategies for clustering related odours. The clustering targets two kinds of entities, namely, either the over 600 olfactory object concepts, or the smell instances. These targets have been studied separately, isolating the relevant embeddings from the rest.

We assess the clustering performances using three common metrics in the literature [41]:

- *Homogeneity*: a clustering output is considered homogeneous if all elements assigned to each cluster belong to the same ground-truth label;
- *Completeness*: a clustering output is considered complete if all elements from one ground-truth label fall into the same cluster;
- *V-Measure*: the harmonic mean of Homogeneity and Completeness. The V-Measure of 1.0 corresponds to a perfect alignment between clusters and ground truth labels.

For both targets, we compared the performances of the 3 embedding algorithms (Section 4.2) using the same K-means clustering algorithm, which is an iterative algorithm that tries to partition the dataset into K pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group. Table 5 provides our evaluation for the 3 algorithms, with respectively 10, 20 and 100 clusters of olfactory objects. Each algorithm has been trained on both the original graph and the dense graph.[8] For each concept, we consider as ground truth class the `skos:broader` term which is also `skos:topConceptOf` the SKOS schema. In other words, we use as classes the 13 top categories of the vocabulary which are reported in Table 2. Given that an olfactory object may belong to multiple top categories – e.g. a *herb* belongs to both *Flora* (being a plant) and *Food* (being a seasoning) – we consider in the computation of the metrics the most popular class in the clustering itself, in order to not penalise meaningful aggregations.

From the results, the benefit of the dense graph is quite obvious: the embeddings trained on the dense graph have better scores regardless of the considered metric, incrementing in some cases of over 0.10 points.

Table 6 presents the same scores computed on smells clustering. In this case, we used the smell sources as the ground truth classes. Again, when a smell has multiple sources, we select the most popular source in the cluster. In addition, when computing the metrics, we also use the top categories of these smells as classes, e.g. *food* for *meat*, *seafood* or *fruit*.

[8] Please note that for DistMult and TransE, which have a longer training time, the 10k version of the graphs has been employed for training.

Table 5. Homogeneity, **C**ompleteness and **V**-measure computed on 10, 20 and 100 clusters of **olfactory objects**. In **bold**, the best results for each number of cluster, while underlined the best values in absolute

	# clusters →	10			20			100		
	trained on ↓	H	C	V	H	C	V	H	C	V
RDF2vec	original	0.80	0.69	0.74	0.87	0.59	0.70	0.95	0.44	0.59
	dense	**0.89**	**0.78**	**0.83**	0.94	0.61	0.74	**0.98**	**0.44**	**0.61**
DistMult	original	0.70	0.62	0.66	0.80	0.55	0.65	0.89	0.42	0.57
	dense	0.88	0.76	0.81	**0.96**	**0.66**	**0.78**	0.97	0.44	0.61
TransE	original	0.10	0.09	0.10	0.18	0.12	0.14	0.36	0.17	0.23
	dense	0.83	0.73	0.77	0.87	0.58	0.70	0.95	0.42	0.58

Table 6. Homogeneity, **C**ompleteness and **V**-measure computed on 10, 20 and 100 clusters of **smells**. In **bold**, the best results for each number of cluster, while underlined the best values in absolute

	# clusters →	10			20			100		
		H	C	V	H	C	V	H	C	V
	trained on ↓	**Smell Source**								
RDF2vec	original	0.04	0.12	0.06	0.05	0.12	0.07	0.08	0.15	0.10
	dense	0.24	0.51	0.33	0.57	0.59	0.58	0.55	0.57	0.56
DistMult	original	0.06	0.14	0.08	0.08	0.14	0.10	0.14	0.15	0.14
	dense	**0.25**	**0.51**	**0.34**	**0.59**	**0.60**	**0.60**	**0.59**	**0.59**	**0.59**
TransE	original	0.05	0.13	0.07	0.06	0.11	0.08	0.22	0.11	0.11
	dense	0.24	0.48	0.32	0.58	0.58	0.58	0.58	0.58	0.58
	trained on ↓	**Top Category**								
RDF2vec	original	0.03	0.03	0.03	0.04	0.03	0.03	0.06	0.04	0.04
	dense	**0.62**	**0.52**	**0.57**	**0.83**	**0.33**	**0.47**	**0.84**	**0.33**	**0.48**
DistMult	original	0.04	0.03	0.04	0.06	0.03	0.04	0.08	0.03	0.05
	dense	0.34	0.27	0.30	0.59	0.23	0.33	0.57	0.23	0.32
TransE	original	0.04	0.03	0.03	0.05	0.03	0.04	0.06	0.02	0.03
	dense	0.22	0.16	0.19	0.43	0.17	0.24	0.45	0.17	0.24

In this scenario, the embeddings trained on the original graph gave low scores, never surpassing 0.22. Moreover, the values are unexpectedly lower for the top category because the metrics are also taking into account the ground truth entropy. Hence, even if the clusters may result more homogeneous in absolute terms, their scores are lower when comparing with the original ground truth homogeneity. These factors make even more evident the increment of scores for the embeddings trained on the dense graph, for which we obtained scores almost always above 0.50, with the scores for top categories definitely better than those referring to fine-grained smell sources.

To provide a qualitative assessment, we computed the most similar entities to the centroid using the cosine similarity, for each cluster obtained by K-means using one of our embedding strategy (RDF2vec, TransE, DistMult), in the 20 clusters scenario. We present the results in Tables 9 to 14, to which we manually added – where possible – a meaningful label for each cluster. Apart from some miscellaneous clusters, the top concepts are quite homogeneous, making easy to manually assign a label to the cluster. Please note that there are some repeated entries (e.g. *sachet* in clusters 2 and 9) because it can happen that two centroids are quite close. Even if DistMult has better scores, qualitatively, the clusters generated on RDF2vec embeddings are easier to manually classify.

It is possible that the two algorithms capture different kind of knowledge. We observe that the clusters are relatively coherent also in the embeddings computed on the original graph.

Overall, we observe that both RDF2Vec and DistMult provide good embeddings of the EOKG for the clustering task, both according the quantitative and qualitative assessment. TransE embeddings yield, in contrast, hard to understand clusters, with lower performance scores.

6. Link Prediction

In this section, we investigate which method among our embedding strategies enables to best enrich the European Olfactory Knowledge Graph. We cast this problem as a link prediction task. More precisely, we aim to predict a possible smell source given a description of a smell experience. For example, looking at the example depicted in Figure 2, the task amounts to predict that the *gloves* are the olfactory object responsible for this smell experience, given the other attributes that describe it such as its characteristics (*subtle*, *rich*, *oriental*) and other circumstances (*19th century*, *Douglas Sherley*, etc.).

We employ two different ground truths: the 565 precise smell sources (a very fine-grained classification task), and their 13 top categories (a coarse-grained classification task) as shown in Table 2. The precise smell source represents the specific class responsible to a particular smell, while its top category represents the broader class to which the smell source class belongs. In the example depicted in Figure 2, the precise smell source is *gloves* while the broader class is *body*.

We take the two subgraph of 10,000 smells and randomly split them into 8,000 training samples and 2,000 test samples. We train TransE and DistMult on the training set. Table 7 provides the uneven distribution of the smell sources in the training dataset while the test dataset follows a similar distribution since the split was stratified.

Table 7. Distribution of the smell sources with their number of occurrences in the datasets. We present only the top 10 smell sources, to which a long tail distribution follows with more than 200 sources having less than 5 occurrences.

Original Graph 10k			**Dense Graph 10k**		
URI	**label**	**count**	**URI**	**label**	**count**
olfactory-objects:72	Flower	1,449	olfactory-objects:72	Flower	1,839
olfactory-objects:138	Rose	387	olfactory-objects:138	Rose	453
olfactory-objects:269	Incense	330	olfactory-objects:78	Plant	288
olfactory-objects:227	Tobacco	199	olfactory-objects:75	Fruit	246
olfactory-objects:270	Musk	187	olfactory-objects:193	Sheep	223
olfactory-objects:258	Jasmine	142	olfactory-objects:126	Water	218
olfactory-objects:223	Violet	111	olfactory-objects:531	Leaf	175
olfactory-objects:108	Odor of sanctity	101	olfactory-objects:460	Carcass	156
olfactory-objects:23	Cadaver	99	olfactory-objects:109	Oil	148
olfactory-objects:17	Blood	99	olfactory-objects:169	Grape	146

The model was trained for 200 epochs, with a batch size of 32. For each triple in the dataset, we attempted to predict the correct smell source by applying the `predict_triples` function from PyKEEN, which calculates a score for each given

triple. We computed the accuracy by extracting the top-scoring entity from the predicted scores and comparing it to the actual smell source from the ground truth.

The results reported in Table 8 demonstrate that even if the task is complex, choosing the right sub-graph can have beneficial results: under any configuration, the accuracy of the prediction is almost doubled for the fine-grained smell source and more than doubled for the top category. Among the two studied methods, DistMult exhibited significantly better performances, achieving an average accuracy of 73% on the top category. However, the accuracy of 42% suggests that the current approach is not yet sufficient to precisely infer the fine-grained smell sources.

Table 8. Accuracy of link prediction for guessing the smell source using graph embeddings

Dataset →	**Original Graph 10k**		**Dense Graph 10k**	
Ground Truth	**Smell source**	**Top Category**	**Smell source**	**Top Category**
TransE	0.18	0.25	0.38	0.69
DistMult	0.28	0.31	**0.42**	**0.73**

7. Conclusions

In this paper, we applied state-of-art graph embedding techniques to the cultural heritage domain, namely on the EOKG, which represents olfactory heritage information. We used them on two different tasks: clustering and link prediction. Given the complexity of the involved Knowledge Graph, we applied a strategy for selecting smaller but curated subsets that are more homogeneous to train our embedding models. This effectively improved the performance of the predictions. In particular, we discovered that a more dense subset improved the accuracy of the models in all metrics. The obtained results can be directly used into the EOKG and support the research on it in the Cultural Heritage domain. The code for computing the embeddings, clustering and link predicting is available at `https://github.com/Odeuropa/kg-embeddings`.

Our future work will focus on using the clustering algorithm to select part of the graph which are more homogeneous in content, in order to refine the link prediction step to a specific graph area. In other words, we would create link prediction models that are specialised alternatively in floral smells, malodors, etc. As alternative, we may substitute the clustering with community detection approaches [42,43]. In addition, we aim to experiment with other graph embeddings methods such as RotatE [22] or GCN to better understand their capabilities in these scenarios where graphs can be sparse but where nodes and properties have well-defined semantics.

Acknowledgements

The authors would like to thank Inger Leemans, Marieke van Erp, Cecilia Bembibre and William Tullett for supporting the interpretation of the results. This work has been partially supported by European Union's Horizon 2020 research and innovation programme within the Odeuropa project (grant agreement No. 101004469).

References

[1] Hogan A, Blomqvist E, Cochez M, d'Amato C, Melo GD, Gutierrez C, et al. Knowledge Graphs. ACM Computing Surveys. 2021;54(4).

[2] Cai H, Zheng VW, Chang KCC. A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. IEEE Transactions on Knowledge & Data Engineering. 2018;30(09):1616-37.

[3] Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O. Translating embeddings for modeling multi-relational data. Advances in neural information processing systems. 2013;26.

[4] Nickel M, Murphy K, Tresp V, Gabrilovich E. A Review of Relational Machine Learning for Knowledge Graphs. Proceedings of the IEEE. 2016;104(1):11-33.

[5] Ristoski P, Rosati J, Di Noia T, De Leone R, Paulheim H. RDF2Vec: RDF graph embeddings and their applications. Semantic Web. 2019;10(4):721-52.

[6] Palumbo E, Monti D, Rizzo G, Troncy R, Baralis E. entity2rec: Property-specific knowledge graph embeddings for item recommendation. Expert Systems with Applications. 2020;151.

[7] Fekete M, Bjerva J, Beinborn L. Typological Challenges for the Application of Multilingual Language Models in the Digital Humanities. In: Multilingual Digital Humanities. Digital Research in the Arts and Humanities. Routledge; 2023. .

[8] Lisena P, Troncy R. Representing Complex Knowledge for Exploration and Recommendation: The Case of Classical Music Information. In: Cota G, Daquino M, Pozzato GL, editors. Applications and Practices in Ontology Design, Extraction, and Reasoning. vol. 49 of Studies on the Semantic Web Series (SSWS). IOSPress; 2020. p. 107-23.

[9] El-Hajj H, Valleriani M. CIDOC2VEC: Extracting Information from Atomized CIDOC-CRM Humanities Knowledge Graphs. Information. 2021;12(12).

[10] Martin-Rodilla P, Gonzalez-Perez C. Representing Imprecise and Uncertain Knowledge in Digital Humanities: A Theoretical Framework and ConML Implementation with a Real Case Study. In: Proceedings of the Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality. TEEM'18. Association for Computing Machinery; 2018. p. 863–871.

[11] Lisena P, van Erp M, Bembibre C, Leemans I. Data Mining and Knowledge Graphs as a Backbone for Advanced Olfactory Experiences. In: STT21: Smell, Taste, and Temperature Interfaces workshop. Yokohama, Japan; 2021. .

[12] Pujara J, Augustine E, Getoor L. Sparsity and Noise: Where Knowledge Graph Embeddings Fall Short. In: Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark: Association for Computational Linguistics; 2017. p. 1751-6.

[13] Portisch J, Hladik M, Paulheim H. RDF2Vec Light – A Lightweight Approach for Knowledge Graph Embeddings. In: International Semantic Web Conference, Posters and Demos; 2020. .

[14] Fanourakis N, Efthymiou V, Kotzinos D, Christophides V. Knowledge graph embedding methods for entity alignment: experimental review. Data Mining and Knowledge Discovery. 2023;37(5):2070-137.

[15] Lisena P, Schwabe D, van Erp M, Troncy R, Tullett W, Leemans I, et al. Capturing the Semantics of Smell: The Odeuropa Data Model for Olfactory Heritage Information. In: 19th European Conference on the Semantic Web (ESWC); 2022. p. 387-405.

[16] Goyal P, Ferrara E. Graph embedding techniques, applications, and performance: A survey. Knowledge-Based Systems. 2018;151:78-94.

[17] Wang X, Bo D, Shi C, Fan S, Ye Y, Yu PS. A Survey on Heterogeneous Graph Embedding: Methods, Techniques, Applications and Sources. IEEE Transactions on Big Data. 2023;9(2):415-36.

[18] Spitzer F. Principles of random walk. vol. 34. Springer Science & Business Media; 2001.

[19] Grover A, Leskovec J. node2vec: Scalable Feature Learning for Networks. In: 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery; 2016. p. 855–864.

[20] Trouillon T, Welbl J, Riedel S, Gaussier E, Bouchard G. Complex Embeddings for Simple Link Prediction. In: 33rd International Conference on Machine Learning. vol. 48 of Proceedings of Machine Learning Research. PMLR; 2016. p. 2071-80.

[21] Yang B, Yih W, He X, Gao J, Deng L. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In: International Conference on Learning Representations ICLR; 2015. .

[22] Sun Z, Deng ZH, Nie JY, Tang J. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In: International Conference on Learning Representations; 2019. .

[23] Hou W, Bai B, Cai C. CR-TransR: A Knowledge Graph Embedding Model for Cultural Domain. Journal on Computing and Cultural Heritage. 2024;17(1).

[24] Han M, Wang Q, Chen H, Chen W, Zhang J, Wang G. Representing the Intangible Cultural Heritage Knowledge Graph with Vector Embedding. In: 2023 IEEE International Conference on High Performance Computing & Communications, Data Science & Systems, Smart City & Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys); 2023. p. 718-25.

[25] Mohamed HA, Vascon S, Hibraj F, James S, Pilutti D, Del Bue A, et al. Geolocation of Cultural Heritage Using Multi-view Knowledge Graph Embedding. In: Rousseau JJ, Kapralos B, editors. Pattern Recognition, Computer Vision, and Image Processing. ICPR 2022 International Workshops and Challenges. Springer Nature Switzerland; 2023. p. 142-54.

[26] El-Hajj H, Zamani M, Büttner J, Martinetz J, Eberle O, Shlomi N, et al. An Ever-Expanding Humanities Knowledge Graph: The Sphaera Corpus at the Intersection of Humanities, Data Management, and Machine Learning. Datenbank-Spektrum. 2022;22(2):153-62.

[27] Baas J, Dastani M, Feelders A. Tailored Graph Embeddings for Entity Alignment on Historical Data. In: 22nd International Conference on Information Integration and Web-Based Applications & Services. iiWAS '20. Association for Computing Machinery; 2021. p. 125–133.

[28] Menini S. Semantic Frame Extraction in Multilingual Olfactory Events. In: Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024). Torino, Italia: ELRA and ICCL; 2024. p. 14622-7. Available from: `https://aclanthology.org/2024.lrec-main.1273`.

[29] Zinnen M, Madhu P, Leemans I, Bell P, Hussian A, Tran H, et al. Smelly, dense, and spreaded: The Object Detection for Olfactory References (ODOR) dataset. Expert Systems with Applications. 2024;255:124576.

[30] Lisena P, Meroño-Peñuela A, Kuhn T, Troncy R. Easy Web API Development with SPARQL Transformer. In: 18^{th} International Semantic Web Conference (ISWC), In-Use Track. Auckland, New Zealand; 2019. .

[31] Menini S, Paccosi T, Tonelli S, Van Erp M, Leemans I, Lisena P, et al. A Multilingual Benchmark to Capture Olfactory Situations over Time. In: 3rd Workshop on Computational Approaches to Historical Language Change. Dublin, Ireland: Association for Computational Linguistics; 2022. p. 1-10.

[32] Miles A, Pérez-Agüera JR. SKOS: Simple knowledge organisation for the web. Cataloging & Classification Quarterly. 2007;43(3-4):69-83.

[33] Chen Y, Sanghavi S, Xu H. Clustering Sparse Graphs. In: Advances in Neural Information Processing Systems. vol. 25. Curran Associates, Inc.; 2012. Available from: `https://proceedings.neurips.cc/paper_files/paper/2012/file/1e6e0a04d20f50967c64dac2d639a577-Paper.pdf`.

[34] Wang S, Zhu W. Sparse Graph Embedding Unsupervised Feature Selection. IEEE Transactions on Systems, Man, and Cybernetics: Systems. 2018;48(3):329-41.

[35] Chen J, Saad Y. Finding dense subgraphs for sparse undirected, directed, and bipartite graphs. Minnesota: University of Minnesota Twin Cities. 2009.

[36] Attenborough M. 19 - Graph theory. In: Attenborough M, editor. Mathematics for Electrical Engineering and Computing. Oxford: Newnes; 2003. p. 461-78.

[37] Mikolov T, Chen K, Corrado G, Dean J. Efficient Estimation of Word Representations in Vector Space. In: International Conference on Learning Representations (ICLR); 2013. .

[38] Chandrahas, Sharma A, Talukdar P. Towards Understanding the Geometry of Knowledge Graph Embeddings. In: 56th Annual Meeting of the Association for Computational Linguistics (ACL). Melbourne, Australia: Association for Computational Linguistics; 2018. p. 122-31.

[39] Steenwinckel B, Vandewiele G, Agozzino T, Ongenae F. pyRDF2Vec: A Python Implementation and Extension of RDF2Vec. In: European Semantic Web Conference (ESWC); 2023. p. 471-83.

[40] Ali M, Berrendorf M, Hoyt CT, Vermue L, Sharifzadeh S, Tresp V, et al. PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings. Journal of Machine Learning Research. 2021;22(82):1-6. Available from: `http://jmlr.org/papers/v22/20-825.html`.

[41] Rosenberg A, Hirschberg J. V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure. In: Eisner J, editor. Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL). Prague, Czech Republic: Association for Computational Linguistics; 2007. p. 410-20. Available from: `https://aclanthology.org/D07-1043`.

[42] Fortunato S. Community detection in graphs. Physics Reports. 2010;486(3):75-174.

[43] Chen J, Saad Y. Dense Subgraph Extraction with Application to Community Detection. IEEE Transactions on Knowledge and Data Engineering. 2012;24(7):1216-30.

Table 9. The most similar concept for each cluster centroid using the **RDF2vec** embeddings (20 clusters), clustered on the **original** graph

CLUSTER 1	**CLUSTER 2**	**CLUSTER 3**	**CLUSTER 4**	**CLUSTER 5**
Fruit	*Cosmetics and artefacts*	*Fruit*	*Flowers*	*Bodily fluids*
Melon	Talc	Lemon	Petunia	Semen
Mandarin	Scent box	Pineapple	Raflessia	Slurry
Durian	Earring	Raspberry	Delphinium	Breath
Lemongrass	Pomander ship	Currant	Chamomile	Liquid manure
Watermelon	Sachet	Coconut	Passion flower	Sweat
CLUSTER 6	**CLUSTER 7**	**CLUSTER 8**	**CLUSTER 9**	**CLUSTER 10**
Malodours	*Spices*	*Fuel*	*Artefacts*	*Balms*
Air pollution	Saffron	Kerosene	Powder Box	Opoponax
Filth	Allspice	Petroleum	Scent box	Tolu balm
Pollution	Vanilla	Sea	Pomander ship	Peru balm
Malodor	Anise	Peat	Feather	Spikenard
Fumigation	Cinnamon cassia	Diesel fuel	Sachet	Balm
CLUSTER 11	**CLUSTER 12**	**CLUSTER 13**	**CLUSTER 14**	**CLUSTER 15**
Beings	*Tobacco*	*Vegetables*	*Carriers*	*Animals*
Being	Cigar-box	Cucumber	Incense ship	Monkey
Woman	Snuff Box	Corn	Tobacco grater	Tiger
Person	Match	Cress	Beaver hat	Rabbit
Man	Tobacco packag.	Asparagus	Beer stein	Guinea pig
Bird	Plague mask	Aubergine	Jan Steen jug	Camel
CLUSTER 16	**CLUSTER 17**	**CLUSTER 18**	**CLUSTER 19**	**CLUSTER 20**
Interior	*Plants*	*Perfumes*	*Miscellaneous*	*Food*
Bath	Furze	Floral water	Bonfire	Curry
Head cone	Stapelia	Geosmin	Pigeon manure	Yeast
Fireplace	Sassafras	Lemon balm	Candle	Chili
Ciborium	Organic waste	Eau de Luce	Alchemy Equip.	Cream
Distillation eq.	Rapeseed	Chypre	Chimney	Liquorice

Table 10. The most similar concept for each cluster centroid using the **RDF2vec** embeddings (20 clusters), clustered on the **dense** graph

CLUSTER 1	**CLUSTER 2**	**CLUSTER 3**	**CLUSTER 4**	**CLUSTER 5**
Vessels	*Spices*	*Artefacts*	*Mammals*	*Flowers*
Barrel	Cinnamon cassia	Hand	Lion	Daffodil
Milk Jug	Allspice	Posy	Wolf	Jasmine
Beer stein	Pepper	Scent	Guinea pig	Violet
Chatelaine flask	Fennel	Perfume	Donkey	Raflessia
Betrothal Ring	Anise	Rosary	Camel	Acacia
CLUSTER 6	**CLUSTER 7**	**CLUSTER 8**	**CLUSTER 9**	**CLUSTER 10**
Resins	*Malodors*	*Fruits*	*Breakfast*	*Secretions*
Styrax	Air pollution	Apple	Butter	Breath
Opoponax	Filth	Rhubarb	Sausage	Semen
Benzoin	Halitosis	Pineapple	Yoghurt	Cat urine
Labdanum	Fumigation	Mandarin	Pie	Sweat
Tolu balm	Malodor	Blackcurrant	Ham	Vomit
CLUSTER 11	**CLUSTER 12**	**CLUSTER 13**	**CLUSTER 14**	**CLUSTER 15**
Vegetable	*Flowers*	*Liquids*	*Fresh flowers*	*Holders*
Horse radish	Garland	Mildew	Bouquet	Scent box
Celery	Violet	Lake	Heliotropium	Pomander ship
Cabbage	Neroli	Diesel fuel	Furze	Pomander Watch
Bellpepper	Hyacinth	Sea	Acacia farnesiana	Tobacco packaging
Pea	Lily-of-the-valley	Wet earth	Water mint	Medicine Jar
CLUSTER 16	**CLUSTER 17**	**CLUSTER 18**	**CLUSTER 19**	**CLUSTER 20**
Tobacco	*Animal-based perfumery*	*Equipments*	*Oils*	*Invertebrates*
Cigarette	Musk Rat	Distillation eq.	Geosmin	Butterfly
Cigar-holder	Musk Deer	Candle	Essential oil	Ant
Cigar-box	Civet (mammal)	Cassolette	Eau de Luce	Prawn
Cigar-case	Skunk	Alchemy equip.	Eau de Hungary	Onycha
Pipe	Beaver	Apothecary eq.	Ointment	Dragonfly

Table 11. The most similar concept for each cluster centroid using the **DistMult** embeddings (20 clusters), clustered on the **original** graph

CLUSTER 1	**CLUSTER 2**	**CLUSTER 3**	**CLUSTER 4**	**CLUSTER 5**
Perfumes	*Trees*	*Malodour*	*Carriers*	*Miscellaneous*
Eau de Lice	Flora	Abstract	Artefact	Nature
Boldoot	Tree	Malodor	Pomander watch	Miasma
Perfume	Pine needle	Halitosis	Pomander ship	Iso-butyl-quinoline
Acqua della Regina	Conifer	Theriaca	Cashmere	Artefact
Peau d'Espagne	Raflessia	Pollution	Posy holder	Cashmere
CLUSTER 6	**CLUSTER 7**	**CLUSTER 8**	**CLUSTER 9**	**CLUSTER 10**
Vegetable/Fruits	*Bodily Fluids*	*Jewelry*	*Flowers*	*Seasonings*
Strawberry	Body	Flacon ring	Neroli	Food
Salad	Grangreen	Bracelet vinaigrette	Columbine	Lemongrass
Cauliflower	Cat urine	Artefact	Heliotropium	Sage
Vegetable	Liquid manure	Jewelry	Carnation	Besamin
Fruit	Slurry	Berothal Ring	Cornflower	Coriander leaf
CLUSTER 11	**CLUSTER 12**	**CLUSTER 13**	**CLUSTER 14**	**CLUSTER 15**
Flora	*Mammals*	*Balms*	*Flowers*	*Vessels*
Flora	Pig	Styrax	Flower	Glass without stem
Pine needle	Tiger	Resin	Matthiola	Vessel
Coriander leaf	Mammal	Animal raw material	Orchid	Cup
Raflessia	Deer	Tolu balm	Raflessia	Ashtray
Beetroot	Guinea pig	Galbanum	Rosehip	Glass with stem
CLUSTER 16	**CLUSTER 17**	**CLUSTER 18**	**CLUSTER 19**	**CLUSTER 20**
Chemical	*Tobacco*	*Animal*	*Artefact*	*Animal*
Chemical Element	Match	Mammal	Jewelry	Invertebrate
Iso-butyl-quinoline	Musical snuff box	Being	Artefact	Reptile
Element	Smoking equipment	Deer	Posy holder	Vertebrate
Vinegar	Blueberry	Sperm whale	Dairy	Being
Ozone	Snuff box	Wolf	Earring	Butterfly

Table 12. The most similar concept for each cluster centroid using the **DistMult** embeddings (20 clusters), clustered on the **dense** graph

CLUSTER 1	**CLUSTER 2**	**CLUSTER 3**	**CLUSTER 4**	**CLUSTER 5**
	Flowers	*Vegetables*	*Fossils*	*Mammals*
Fireplace	Dahlia	Avocado	Peat	Civet (mammal)
Amulet	Woodruff	Parsnip	Mould	Guinea pig
Ink	Dandelion	Passion fruit	Petrichor	Bear
Flacon	Garland	Chives	Mildew	Donkey
Cigar	Mimosa	Radish	Diesel fuel	Beaver
CLUSTER 6	**CLUSTER 7**	**CLUSTER 8**	**CLUSTER 9**	**CLUSTER 10**
Resins			*Herbs*	*Chemicals*
Myrrh	Anatomical Lessons	Avocado	Thyme	Aceltene
Wax	Burnt offering	Curry	Rosemary	Carbolic acid
Rubber	Blossom	Cress	Musk mallow	Ozone
Resin	Soot	Passion fruit	Mint	Coumarin
Balm	various	Schalg	Basilic	Acetic acid
CLUSTER 11	**CLUSTER 12**	**CLUSTER 13**	**CLUSTER 14**	**CLUSTER 15**
Fruits	*Fishy*	*House products*	*Animal categories*	*Holders*
Peach	Slurry	Cashmere	Vertebrate	Cashmere
Plum	Cod liver oil	Bouquet Holder	Being	Posy holder
Mandarin	Liquid manure	Ammonia soap	Reptile/Amphibia	Chatelaine flask
Grape	Whale oil	Smelling Box	Amphibia	Cigar-case
Pear	Spermaceti	Cigar-case	Insect	Rope tobacco
CLUSTER 16	**CLUSTER 17**	**CLUSTER 18**	**CLUSTER 19**	**CLUSTER 20**
Trees	*Scented waters*	*Artefacts*	*Flowers*	*Malodors*
Tagetes	Boldoot	Jewelry	Petunia	Halitosis
Fir needle	Aqua mirabilis	Ring	Dandelion	Filth
Cedar (Lebanon)	Florida Water	Bracelet vinaigrette	Tulip	Sillage
Rapeseed	Reukwerk	Wine Bottle	Dahlia	Malodor
Oud	Peau d'Espagne	Oil lamp	Carnation	Fumigation

Table 13. The most similar concept for each cluster centroid using the *TransE* embeddings (20 clusters), clustered on the **original** graph

CLUSTER 1	**CLUSTER 2**	**CLUSTER 3**	**CLUSTER 4**	**CLUSTER 5**
Lether	Sulphur	Fumigation	Allspice	Worm
Perfume	Asa foetida	Pidgeon manure	Geranium	Geranium
Vegetable	Man	Powdered wig	Turmeric	Ambergris
Lily of the valley	Lime blossom tree	Asa foetida	Currant	Turpentine
Fruit	Geranium	Geranium	Tulip	Wet earth
CLUSTER 6	**CLUSTER 7**	**CLUSTER 8**	**CLUSTER 9**	**CLUSTER 10**
		Artefacts		
Watermelon	Fish	Drinking Glass	Lavender	Ointment
Pineapple	Pipe	Bracelet	Geranium	Pipe
Physalis	Oil	Ashtray	Hyacinth	Smoking equipment
Sausage	Fruit	Smelling Bottle	Water	Bread
Geranium	Geranium	Wine bottle	Saffron	Tobacco packaging
CLUSTER 11	**CLUSTER 12**	**CLUSTER 13**	**CLUSTER 14**	**CLUSTER 15**
Geranium	Acacia	Fur	Vial	Smoke
Jonquil	Geranium	Man	Butterfly	Religion
Passion flower	Hazelnut	Iris	Asa Foetida	Vegetable
Frog	Asa foetida	Herb	Geranium	Wine
Butterfly	Mustard	Geranium	Calamus	Garlic
CLUSTER 16	**CLUSTER 17**	**CLUSTER 18**	**CLUSTER 19**	**CLUSTER 20**
Matthiola	Egg	Ink	Being	Patchouli
Musk Deer	Lignum aquilae	Raspberry	Food	Gunpowder
Inro	Tallow	Vinegar	Flora	Geranium
Dairy	Geranium	Geranium	Element	Egg
Pineapple	Smelling box	Myrtle	Animal raw material	Sulphuric acid

Table 14. The most similar concept for each cluster centroid using the *TransE* embeddings (20 clusters), clustered on the **dense** graph

CLUSTER 1	**CLUSTER 2**	**CLUSTER 3**	**CLUSTER 4**	**CLUSTER 5**
Fruits	*Artefacts*	*Food*	*Malodors*	
Pineapple	Head cone	Oil	Malodor	Lodereindoos
Watermelon	Pomander	Coffee	Odor of sanctity	Leather Tapestry
Halitosis	Chamber pot	Blueberry	Fumigation	Cigar-box
Peanut	Perfume flacon	Spice	Abstract	Diaper
Apricot	Lodereindoos	Vegetable	Pollution	Perfume box
CLUSTER 6	**CLUSTER 7**	**CLUSTER 8**	**CLUSTER 9**	**CLUSTER 10**
	Trees	*Plants*	*Plants*	
Snuff	Sassafras	Rosehip	Elderflower	Spirit (alcohol)
Myrrh	Cedar (Lebanon)	Elm tree	Rapeseed	Carrot
Sulphur	Cedar (Virginia)	Passion flower	Quince	Grapefruit
Grass	Stapelia	Heliotropium	Acacia farnesiana	Broth
Blackcurrant leaf	Acacia farnesiana	Quince	Dahlia	Schalg
CLUSTER 11	**CLUSTER 12**	**CLUSTER 13**	**CLUSTER 14**	**CLUSTER 15**
Mammals	*Vegetables*	*Animal products*	*Scented waters*	*Flowers*
Wolf	Pea	Body	Florida Water	Dahlia
Beaver	Radish	Tallow	Eau de cologne	Geranium
Lion	Carrot	Liquid manure	Floral water	Furze
Bull	Cress	Whale oil	Fougère	Columbine
Man	Spring onion	Ambergris	Boldoot	Anemone
CLUSTER 16	**CLUSTER 17**	**CLUSTER 18**	**CLUSTER 19**	**CLUSTER 20**
Vessels	*Nature*	*Summer Fruits*	*Artefacts*	*Chemicals*
Vessel	Nature	Fig	Powdered Wig	Carbolic acid
Drinking glass	Peat	Spring onion	Necklace	Aceltene
Teapot	Wet earth	Plum	Vial	Chemical Element
Bottle	Wind	Peach	Smelling Bottle	Element
Pot	Lake	Coconut	Lodereindoos	Iso-butyl-quinoline