Using Knowledge Graphs to connect wildlife information sources and investigate impact caused by war

Thomas van Zwol[26555625] and Maarten van den Ende[2529117]

Knowledge Representation on the Web - Vrije Universiteit Amsterdam

Abstract. In this research we create a tool to connect the IUCN Redlist data with Wikidata and the WWF Biome information. This is the first and most complete and versatile tool to obtain information about wildlife from different sources. This tool should be able to be used by conservation agencies to discover where and what are the most effective actions to reduce biodiversity reduction.
We then use this as a case study by combining information on active war missions of the UN to see what animals are most threatened by these wars and to discover ending which war has most environmental impact.

Keywords: Wildlife Conservation · Knowledge Engineering · Linked Data · Visualisation

1 Introduction

The worldwide biodiversity is under pressure and many species of plants and animals are threatened with extinction [9]. This study reveals that the rate of extinction for species in the 20th century has been up to 100 times higher than would have been normal without human impact. There are many environmental and wildlife protective agencies around the world trying to combat the loss of biodiversity and preserving wildlife.

The main global ones are World Wildlife Foundation and the International Union for Conservation of Nature. Next to this, many local agencies try to have a positive impact on conserving biodiversity in more specific areas. As an example a quick google search shows over four different organizations specifically focused on preservation of chimpanzees, like Chimps Inc., Chimp Haven Inc., Mona Foundation and Save the Chimps. This in addition to more general foundations focused on primates like Primarily Primates Inc. and Ape Cognition and Conservation Project.

All these organizations work together as much as possible to work as efficiently as possibly to achieve their goals. IUCN and WWF for example worked together in creating the Wildlife Trade Monitoring Network TRAFFIC[3]. However, each of the organisations collect their own data and handle and store their data in different ways. Especially smaller local organizations do not have the ability to easily access the global knowledge of other organizations.

IUCN classifies which animals are threatened by what kinds of activities. Knowing what threat to investigate, can be an important part of discovering how to combat the decline of biodiversity. The hunting of mammals in West-Africa, for example, was mainly due to the need for proteins[8]. It turned out that the threat to the mammals was linked to fish supply, which in itself has its own threats. This shows that research into threats and their causes is an important part of wildlife conservation. In order to more easily identify research areas of interest, we aim to create a knowledge graph that incorporates different sources of open data.

This research has the goal of increasing efficiency of handling data for all agencies, and creating an easier, better accessible and broader knowledge base of wildlife and nature data. By linking the different datasets together, it will be easier for organisations to know where action is needed most.

This is done by combining data from WWF and IUCN as well as WikiData into a Knowledge Graph and creating a tool that makes obtaining information from these sources straightforward and easy. This technology is made to be easily extendable with more data sources using existing ontologies and a specially created ontology with the goal of giving smaller conservation initiatives the possibility to add their specialized knowledge.

As a case study to show the use of this tool we will add the UN Active Peacekeeping Missions data, which uses many features of the tool like the interactive map and flexible queries. We then showcase the capabilities of the tool by trying to answer the question on which war imposes most threats on biodiversity and thus which peace keeping mission UN WCMC could piggyback on to increase the conservation efforts in the area.

2 Related Work

To the best of our knowledge, there has not yet been research in constructing a knowledge graph for the domain of wildlife conservation. But for a number of other domains this has been done: combating human trafficking [11], illegally parked bicycles [10] and open city data [7].

All three papers cited above note that the fragmentation of data hinders the ability of the responsible people to effectively make use of the data. By establishing links between data, this problem is addressed. How to exactly do this linking is the problem they investigate.

One of the challenges [11] faced, was the fact that the different data sources didn't use the same schemas. This is something they handled by using a self learning tool that was used to map the data sources to the ontology they created.

In [7] the owl:sameAs relation is used to ensure that there is a one to one mapping between the same cities in different data sets. Basic entity recognition was used to identify the resources to be linked together on the basis of the city name.

For [10] a complete data set was created by combining data from different APIs. There was no need for linking between existing data sets, but the data

returned from the different APIs were mapped to the schema that they had designed. This is done in real time, ensuring the data set stays up to date.

3 Methods

The methodology of this research in broad terms exists of obtaining data from different sources. This data comes in different formats, with some already being in triple format and other data still had to be converted. To do this we create a custom ontology next to the already existing RDFS, OWL and SKOS. Linking everything together was the biggest challenge in this research. Finally an interactive web app to make accessing and using the data easily obtainable is created.

3.1 Data sources

The data sources we used are the IUCN Redlist, WikiData, iNaturalist and a list of active UN Missions.

The main data source used is the IUCN Redlist [2]. IUCN provides an API which one can query for taxon name, country, region and more. However, the API is heavily focused on species and taxons and as our goal is to create a versatile tool we want to be able to select on much more like type of threat and conservation measures. To be able to do this we needed to have all data and connect all features using a Knowledge Graph. Therefor we scraped the IUCN data of all mammals. Later this could be extended to include all species. The obtained dataset in the end exists of the complete data with classification schemes defined by the IUCN Classification Scheme [1].

The WikiData Knowledge Base is used to provide the tool with additional general information of the animals. Once the user enters a query the tool requests a SPARQL Query to WikiData using the taxon of the animal. The images in our web app come from WikiData and prove that the linking is successful. Other data about the species can be incorporated in the web app as they are deemed necessary.

New methods for collecting data for creating Linked Open Data include crowd sourcing [10]. Since there is, to the best of our knowledge, no data set with observations of animals tied to locations, we used observations from iNaturalist to enrich the IUCN data with observations. iNaturalist is an app and website where users can submit their observations of flora and fauna. It has as one of its goals generating data for scientific research. Since observations change over time, we used real time integration similar to [10] to ensure our data set stays up to date. This also saved us time when constructing the data set, since the data is only collected when needed and consequently stored for reuse.

The United Nation has a CSV dataset of active missions [4]. It is a small dataset consisting of less than twenty active missions. It gives the name, the location of the UN Headquarters, which can be a village, city or just a general area. It also gives the country code it is located in in multiple formats. This CSV

was converted to RDF triple format. For a more detailed description of how the data was obtained and converted, see Appendix A.

3.2 Linking the data

In contrast with [11], the number of data sources was relatively small for us. Only a small number of data sources is unstructured, allowing us to determine the mapping between them by hand. Linking the instances was done with the owl:sameAs link, similar to [7]. We didn't need to use entity recognition to achieve this: each species has a unique taxon that is the same across all data sets. Establishing an owl:sameAs relation between the properties identifying the taxon allowed for easy instance linking.

Most of the actual linking and aggregating the data happened real time in the web app. The obtained data can then be stored in our triple store, eliminating the need for repeated queries for the same species.

3.3 Ontology and mapping

The IUCN Classification system [1] provides a well defined set of classes and subclasses which could be taken over in the Knowledge Graph. For example, the main class 'Threats' consists of multiple different subclasses like agricultural threats, energy production and pollution. These subclasses are more specific in their respective subclasses. For pollution these can be urban waste water or industrial and military effluents. The second one is then classified finally in Oil spills, Seepage from mining and Unknown. The specific threat to a species is an instance of these classes, since the specific threat also has other information that is specific to the species. The same goes for the habitats and measures.

3.4 Interactive Web App

The Web App is built as a tool to make the data more comprehensible. It consists of the user interface, with easy search queries for users as well as clickable links and clickable biome and warzone map. A more comprehensive description of the web app and how it functions can be found in Appendix D.

4 Results

In this project we have successfully created an interactive tool making it easy for nature and wildlife conservation organizations to access data from IUCN. The tool works relatively quickly, allows exploration of the IUCN data in a linked fashion while also incorporating other linked data from WikiData and non structured data from iNaturalist. The system allows for addition of other data sources relatively easy, which in the future can be used by other organizations to add their specific knowledge and further complete this overview of species and their conservation status.

We have added a simple dataset of Active Peacekeeping Missions of the United Nations with the goal of easily displaying which animals that IUCN classifies as threatened by war, are found near UN Peacekeeping Missions. Determining which animals are actually threatened by the conflicts is hard to do, since the geographical extend of the conflict is not apparent from the dataset. Since finding which animals are near the headquarters requires a radius, the fact that this information is missing makes it hard to determine which war is most detrimental to bioconservation.

5 Conclusion

This has been a reasonably successful research project, both in proving a concept and idea as well as giving some useful information that could not be obtained otherwise. We have created an intuitive tool that combines information from multiple sources and with this it could be one of the biggest known datasets on wildlife and its conservation.

The tool can be used to explore the web of data that we have obtained from IUCN and enriched with WikiData information. The headquarters of the UN missions are plotted on the map and are clickable, resulting in a query for animals that have been observed in a certain radius around the headquarter.

If the tool is extended and used by multiple organizations it could prove to be very useful in aiding nature conservation by making data easier accessible and obtainable and thus streamlining the information process.

6 Discussion

The main focus point for improving our tool would be to expand the number and types of animals included and incorporate the data WWF has but has not provided on their website. The first part can be achieved by scraping more data from the IUCN API. The second part needs cooperation from WWF to provide us with the data.

A limitation to extending our dataset of animals from IUCN, is scraping time. As the API can only be queried per species, the scraper took about 10 hours to run just for mammals. While this is still very viable, it needs to be taken into account when extending for more species, as scraping time might be non-viably long.

The tool as it is now is mainly a proof of concept, with promising abilities. Its use can be increased significantly with added information and utilities.

The map is already a special addition to the tool. More spatially distributed information can be added to the map, like logging work, wildfires or urbanization, to improve the visualisation of the spacial aspect of wildlife conservation. The interactivity of the map could be increased as well, making the wars or other data clickable to show more in-depth information.

The WikiData connection is used now to provide additional information. However, this information is readily available and now mainly adds convenience.

Connecting the tool with other Knowledge Graphs is a promising proof of concept that could make it even more simple to add additional information owned by local conservation organizations, if they would store their information in a Knowledge Graph.

One nice addition is that the data that is queried from WikiData and iNaturalist.org, can be saved in our triple store as the requests are made. This means that our tool overtime becomes less dependent on the other data sources and our data becomes enriched further.

References

1. IUCN Classification. https://www.iucnredlist.org/resources/classification-schemes, accessed: 2019-06-03
2. IUCN Red List API. http://apiv3.iucnredlist.org/, accessed: 2019-04-26
3. TRAFFIC. www.iucn.org/news/secretariat/201710/iucn-welcomes-wwf-international-iucn-conservation-centre, accessed: 2019-04-26
4. United Nations Peacekeeping Missions. https://peacekeeping.un.org/en/peacekeeping-master-open-datasets , accessed: 2019-06-03
5. WWF Terrestrial Ecoregions. https://www.worldwildlife.org/publications/terrestrial-ecoregions-of-the-world, accessed: 2019-06-03
6. WWF WildFinder. https://www.worldwildlife.org/publications/wildfinder-database, accessed: 2019-04-26
7. Bischof, S., Martin, C., Polleres, A., Schneider, P.: Collecting, integrating, enriching and republishing open city data as linked data. In: International Semantic Web Conference. pp. 57–75. Springer (2015)
8. Brashares, J.S., Arcese, P., Sam, M.K., Coppolillo, P.B., Sinclair, A.R., Balmford, A.: Bushmeat hunting, wildlife declines, and fish supply in west africa. Science **306**(5699), 1180–1183 (2004)
9. Ceballos, G., Ehrlich, P.R., Dirzo, R.: Biological annihilation via the ongoing sixth mass extinction signaled by vertebrate population losses and declines. Proceedings of the National Academy of Sciences **114**(30), E6089–E6096 (2017). https://doi.org/10.1073/pnas.1704949114, https://www.pnas.org/content/114/30/E6089
10. Egami, S., Kawamura, T., Ohsuga, A.: Building urban lod for solving illegally parked bicycles in tokyo. In: International Semantic Web Conference. pp. 291–307. Springer (2016)
11. Szekely, P., Knoblock, C.A., Slepicka, J., Philpot, A., Singh, A., Yin, C., Kapoor, D., Natarajan, P., Marcu, D., Knight, K., et al.: Building and using a knowledge graph to combat human trafficking. In: International Semantic Web Conference. pp. 205–221. Springer (2015)

A Datasets and methods for conversion

A.1 IUCN Redlist Scrape

The scraper uses the API to obtain all information per species by going through the list of mammals as provided by the IUCN API. It then takes the information

given and puts it into triple format using existing ontology as well as a custom written one which will be discussed in a following section. As the API is queried per species, scraping just the mammals takes around 10 hours.

A.2 WikiData

The WikiData was obtained real time through a SPARQL query. It was linked to the IUCN data through the taxon of the species.

A.3 iNaturalist

The iNaturalist dataset contains a large amount of data about species, their observations and about the observations themselves (who made them, when, how many people agree on the species etc.). We used the part of the API that allowed us to query for observations of mammals within a certain radius of specific coordinates to determine which animals live in regions where UN Peacekeeping missions are underway. These observations were integrated in the web app in real time.

A.4 UN Active Missions

The United Nation has a CSV dataset of active missions [4]. It is a small dataset consisting of less than twenty active missions. It gives the name, the location of the UN Headquarters, which can be a village, city or just a general area. It also gives the country code it is located in in multiple formats. The CSV was converted with COW.

The main challenge in this dataset is connecting the locality of the war, as the country codes are unusable, both because of some wars lacking them as well as the lack of data on connecting them with the countries name. This gave us two options: using the GeoNames Database to automatically obtain the countries name or doing it manually. While the automated version would have more scalability we decided that for this few instances the manual solution of taking the longitude and latitude was quicker.

A.5 WWF Data

Initially we wanted to use data from WWF as well. However, it turned out that the data we wanted to use was incomplete and specifically missing for the mammals.

There are two datasets from the World Wildlife Foundation that we wanted to use. They are the Terrestrial Ecoregions Dataset [5] and the Wildfinder Dataset [6]. The Ecoregions dataset, which consists of shapefiles of all ecoregions in the world, we wanted to use in our webapp as a map. This would be done by converting it to a GeoJson and shown using Dash and React. The Wildfinder dataset is used to connect the taxon of species with the ecoregions they occur in. However the ecoregions for the mammals were all ‘None’.

We did extract and process the the Wildfinder dataset, only to find out we could not use it. It is a Microsoft Access Database file, consisting of multiple linked data tables. For example one table gives the name of ecoregions for each ecoregion code, while another gives the WWF Species ID with the ecoregions this species occurs in. Then another table connects the WWF ID with the taxon of the animal. Using a Python script these tables are linked to each other to finally be able to connect the taxon directly with the Ecoregion name they live in.

A WWF ontology has been created using the class Species and hasEcoregion and hasGenusSpeciesName have been created to incorporate the ecoregions species live in. The instance hasGenusSpeciesName is then connected using owl:sameAs with the scientific name of the IUCN species. Since the data turned out to be incomplete, we didn't incorporate this link and the data in our final dataset.

B Ontology design and reuse

We didn't find an ontology that was suitable for what we wanted to achieve (mapping IUCN data to RDF), so we created an ontology ourselves. For every piece of information that IUCN had on the animals, we turned that into a property.

We used SKOS broader and narrower to define relations between different levels of taxa. The classifications of threats, measures and habitats was derived from the classification scheme from IUCN. How it was applied can be found in the python file names iucn_classification_to_rdf.py.

The following mapping was used (copied and adjusted for latex from the python file iucn_json_to_rdf.py that did the mapping):

iucnToRdfMap:
"taxonid": 'type': XSD.int, 'propertyName': 'hasTaxonId'
"scientific name": 'type': XSD.string, 'propertyName': 'hasScientificName'
"kingdom": 'type': XSD.string, 'propertyName': 'inKingdom', 'className': 'Kingdom', other': [((nc['Kingdom'], SKOS.narrower, nc['Phylum']))]
"phylum": 'type': XSD.string, 'propertyName': 'inPhylum', 'className': 'Phylum', 'other': [((nc['Phylum'], SKOS.narrower, nc['Class'])), ((nc['Phylum'], SKOS.broader, nc['Kingdom']))]
"class": 'type': XSD.string, 'propertyName': 'inClass', 'className': 'Class', 'other': [((nc['Class'], SKOS.narrower, nc['Order'])), ((nc['Class'], SKOS.broader, nc['Phylum']))]
"order": 'type': XSD.string, 'propertyName': 'inOrder', 'className': 'Order', 'other': [((nc['Order'], SKOS.narrower, nc['Family'])), ((nc['Order'], SKOS.broader, nc['Class']))]
"family": 'type': XSD.string, 'propertyName': 'inFamily', 'className': 'Family', 'other': [((nc['Family'], SKOS.narrower, nc['Genus'])), ((nc['Family'], SKOS.broader, nc['Order']))]
"genus": 'type': XSD.string, 'propertyName': 'inGenus', 'className': 'Genus',

'other': [((nc['Genus'], SKOS.broader, nc['Family']))]
"main common name": 'type': XSD.string, 'propertyName': 'hasCommonName'
"subspecies": 'type': XSD.string, 'propertyName': 'hasSubspeciesLabel'
"rank": 'type': XSD.string, 'propertyName': 'hasRankLabel'
"subpopulation": 'type': XSD.string, 'propertyName': 'hasSubpopulation'
"category": 'type': XSD.string, 'propertyName': 'hasCategory', 'className': 'Category'
"authority": 'type': XSD.string, 'propertyName': 'hasAuthority'
"published year": 'type': XSD.gYear, 'propertyName': 'hasPublicationYear'
"assessment date": 'type': XSD.date, 'propertyName': 'hasAssesmentDate'
"criteria": 'type': XSD.string, 'propertyName': 'meetsCriteria'
"population trend": 'type': XSD.string, 'propertyName': 'hasPopulationTrend'
"marine system": 'type': XSD.boolean, 'propertyName': 'livesInMarineSystem'
"freshwater system": 'type': XSD.boolean, 'propertyName': 'livesInFreshwaterSystem'
"terrestrial system": 'type': XSD.boolean, 'propertyName': 'livesInTerrestialSystem'
"assessor": 'type': XSD.string, 'propertyName': 'hasAssesor', 'className': 'Assesor'
"reviewer": 'type': XSD.string, 'propertyName': 'hasReviewer', 'className': 'Reviewer'
"aoo km2": 'type': XSD.string, 'propertyName': 'hasAreaOfOccupation'
"eoo km2": 'type': XSD.string, 'propertyName': 'hasExtendOfOccurence'
"elevation upper": 'type': XSD.int, 'propertyName': 'hasElevationUpperLimit'
"elevation lower": 'type': XSD.int, 'propertyName': 'hasElevationLowerLimit'
"depth upper": 'type': XSD.int, 'propertyName': 'hasDepthUpperLimit'
"depth lower": 'type': XSD.int, 'propertyName': 'hasDepthLowerLimit'
"errata flag": 'type': XSD.boolean, 'propertyName': 'hasErrataFlag'
"errata reason": 'type': XSD.string, 'propertyName': 'hasErrataReason'
"amended flag": 'type': XSD.boolean, 'propertyName': 'hasAmendFlag'
"amended reason": 'type': XSD.string, 'propertyName': 'hasAmendReason'
start of threaths, habitats and measures **"code"**: 'type': XSD.string, 'propertyName': 'hasCode'
"title": 'type': XSD.string, 'propertyName': 'hasTitle'
"timing": 'type': XSD.string, 'propertyName': 'hasTiming'
"scope": 'type': XSD.string, 'propertyName': 'hasScope'
"severity": 'type': XSD.string, 'propertyName': 'hasSeverity'
"score": 'type': XSD.string, 'propertyName': 'hasScore'
"invasive": 'type': XSD.boolean, 'propertyName': 'isInvasive'
"suitability": 'type': XSD.string, 'propertyName': 'hasSuitability'
"season": 'type': XSD.string, 'propertyName': 'hasSeason'
"majorimportance": 'type': XSD.string, 'propertyName': 'hasMajorImportance'

nc['Mammal'], RDF.type, RDFS.Class nc['Assesor'], RDF.type, FOAF.Person nc['Reviewer'], RDF.type, FOAF.Person nc['Threat'], RDF.type, FOAF.Class nc['Habitat'], RDF.type, FOAF.Class nc['Country'], RDF.type, FOAF.Class nc['Measure'], RDF.type, FOAF.Class

C Instance linking

As described in the methods section, instances were linked with owl:sameAs. The owl:sameAs link was established between the properties in each ontology that indicated the animal taxon, since this is a functional and inverse functional property: each species has one taxon and a taxon only belongs to one species.

Concrete this meant linking hasScientificName from our ontology to the WikiData Property:P225. Since the instances of the species in iNaturalist were queried from the API and only then created in RDF, linking was a matter of adding the information obtained to our already created instances.

D Querying / Visualization and Interface

The web app is built in React (Javascript) and uses http requests to connect to both the GraphDB database that acts as our triple store and the WikiData SPARQL endpoint. The SPARL queries are dynamic, meaning that the search terms a user enters become part of the query.

Currently not all the information we have on an animal is displayed, due to the tool becoming too cluttered. The most important information for navigating through the web of data we have created is there though.

D.1 Triplestore

We use GraphDB as our triple store. We chose this triple store because it was recommended to us and has an easy to access SPARQL endpoint to query. Furthermore it has a built in reasoner that allowed us to easily link the WWF data with the IUCN data through owl:sameAs links.

Linking with data sources Connecting both SPARQL endpoints proved a challenge, due to the fact that federated queries to GraphDB didn't work as expected and because sending federated queries to the WikiData endpoint wouldn't make sense since the GraphDB endpoint was on a local machine.

The data from WikiData that is incorporated at this point in time is a proof of concept that linking between the IUCN dataset and WikiData is possible. Other types of data can be incorporated relatively easily in the web app.

Linking with the API from iNaturalist was relatively easy once we had connected the two SPARQL endpoints. Converting this data to triple format was a little harder and was done in real time.

User interface The user interface consists of three main components: the search and select on the left, the information panel in the top right and in the bottom right the map. In the search bar one can enter a search term and select what the search term refers to: species name, habitat, threat, or measure. Then the possible matches show up as cards with the name of the animal as well as an image. After selecting one, the available data is shown top right, with clickable links initiating a search for that specific item wherever possible. This makes it so that for example you can explore other animals in the same habitat without querying the habitat manually, but instead by clicking the habitat link.

Next there is a map that shows active war missions on top of google maps. They are shown as red dots that when clicked show the animals that are threatened by that war.

E Code and instructions

The code can be found in the GitHub repository. There is a read me file as well for how to run the code to be able to get the web app working.

F Division of labour

The rapport has mostly been written by Maarten van den Ende, with a few additions here and there from Thomas van Zwol.

The web app has been built by Thomas van Zwol. Maarten van den Ende made an interactive map with the ecoregions from WWF, but since the WWF data was not included due to missing data, this map was not included in the web app. Instead a google maps instance was used. The ecoregions map can be found in the GitHub repository.

The data from WWF and UN Peacekeeping Missions was converted by Maarten van den Ende. The data from IUCN was scraped and converted by Thomas van Zwol. Combining all the converted data in the GraphDB triplestore was done by Thomas van Zwol. The links with WikiData and iNaturalist were established by Thomas van Zwol.