BMKG Technical Report

Brewing Connections

A Knowledge Graph of Belgian Beer and Breweries

Data Science for Decision Making
2025

Author:
Leo J. Paggen

Student ID:
i6236337

Course Coordinator:
Prof. Dr. Michel Dumontier

Teacher Assistants:
Rena Yang
Shervin Mehryar

13th March 2025

Contents

1 Abstract

Belgium is widely known for its deep tradition of brewing beer, with hundreds of breweries producing a wide variety of beers. This project introduces a knowledge graph to map relations between beers, breweries, user ratings, and many more fields. By integrating data from multiple providers, the graph presents a novel representation of Belgian beers, while providing users with the power to gain insights regarding the influence of breweries, and the popularity of beers, as well as geographical information about the breweries. This graph opens the door to future improvements, such as a recommender system, or a retrieval augmented generation system.

2 Introduction

Belgium is known for its beer. From sophisticated Trappist beers to less elaborate supermarket beer, Belgian beer is deeply rooted in local culture, and is appreciated by an ever-growing international public. A downside to the vast array of beers brewed in Belgium is that extracting useful insights on any beer is a challenge, mainly due to the fact that key information such as user ratings or ingredients, is scattered across many different sources. In 2023, the number of active breweries in Belgium was estimated to be more than 400 [7]. Extracting data from individual breweries is therefore a tedious process, and the existing platforms which compile beer data are plenty.

The aim of this project is to construct a knowledge graph that compiles information about beers, breweries, user ratings, brewery location, and influences in a single source. By organizing as much data as possible into such a data structure, we can provide easier and better insights as well as better analysis of Belgian beers. As such, two main research questions will be answered in this report: ”How does a knowledge graph improves upon the data extraction capabilties of the current platforms?” and ”What are the technical challenges associated with maintaining this knowledge graph”.

2.1 Primary Objectives

The primary objectives of this project are: gathering data from different sources and compiling it into a single graph, build an intuitive ontology to capture the domain of beer as well as possible, and query the resulting graph to demonstrate the power of such a structure compared to other platforms.

2.2 Problem Description

Currently, the data related to Belgian beer is scattered around many different sources, most of which do not provide us with a structured basis for proper analysis. These sources do not allow for complex queries, and often fail to represent the complex relations between beers in Belgium. This knowledge graph aims to provide a solution to this problem by connecting beer-related entities and information in more appropriate ways, enabling users to obtain richer insights, and more information about the beers they enjoy.

This knowledge graph contributes to the archiving of information on Belgian beers on the

internet, and serves as a proof of concept for future researchers and enthusiasts alike.

3 Related Work

Food-related knowledge graphs have been an increasingly popular subject of research. To build such knowledge graphs, researchers use ontologies. One popular ontology related to food is *FoodOn*[4], an ontology which provides over 9,000 unique food product categories. The scope of this project is smaller, but in a similar fashion, this graph uses a custom ontology tailored to the domain of Belgian beer. An example of a knowledge graph built around food is *FoodKG*. Built by Hausmann et al. (2019), it focuses on recipes and ingredients, and provides an extensive knowledge base for users to query, and researchers to use. FoodKG is the most similar to this project, as it specifically focuses on food, which is a large part of this project.

Regarding the compilation of beer-related data in one single place, platforms such as Untappd[6] and BeerAdvocate[1] have already accomplished much of this task. One area where they do differ a lot from the

Knowledge graphs focusing on the preservation of one region or country's culture have also been developed. One such example is *ArCo*[2], the Italian Cultural Heritage Graph. While ArCo does not focus on food or beverages, it does focus on features from a single country, and also provides a rich set of triples for users to query. Similarly, Dou et al. (2018) propose a knowledge graph focused on preserving Chinese culture[5]. Again, this knowledge graph does not specifically focus on food, but it is easy to argue that beer is an important aspect of Belgian culture.

4 Methodology

4.1 Collecting and Processing the Raw Data

Given the goal of this project to incorporate different sources into a single knowledge graph, web scraping was used extensively to gather data from Wikipedia and Untappd. This section provides an overview of how the data was gathered for the project.

The data collected from Wikipedia concerned both a list of Belgian beers[9] (and other characteristics) and a list of Belgian breweries[10]. To scrape the data from the relevant tables on both pages, Python's BeautifulSoup[14] library was used. Scraping data from Wikipedia is a straightforward process, as long as one respects the regulations of the website regarding scraping. The information extracted from Wikipedia is: beer names, alcohol percentage, breweries, type of beer, brewery province, and brewery municipality, all of which were stored in a Pandas[11] DataFrame. Both aforementioned tables contain the "Brewey" column, but many of the names in this column are not exact matches across both tables, therefore, Lehvenstein[8] distance (80% threshold) was used to rename all entries from table A breweries to their most similar match in table B. This operation allowed for the combination of both tables into a single one. A novel feature was added to the table, namely the coordinates of each brewery, obtained by using Python's Geopy[3] library. Some further processing and cleaning of the data was done, mostly manually using excel, although the work that could be done in Pandas was automated. These operations resulted in a clean dataset, which was still missing data such as user ratings,

and beer descriptions, as well as popularity, all of which were obtained from Untappd.

Scraping data from Untappd is more complex than scraping from Wikipedia: the site blocks proxies (rotating residential too!), and is quite fast to ban users from scraping their publicly available data. Given they have locked public access to their API, Selenium[17] was used to get the data from Untappd. This task was done in two parts: first, search URLs were constructed using the names of the beers from the dataset. The search URLs were used to find a page for each specific beer on Untappd. Again, Lehvenstein distance (threshold of 70%) was used to find the link with the closest match to each beer. Secondly, once on the relevant page, obtaining ratings, popularity, and descriptions was straightforward. Unfortunately, for a lot of the beers in my table, the Lehvenstein distance was not enough to find a valid URL even if the beer was in the database of Untappd. Perhaps future work should use more elaborate tools such as LLMs should be used for this process. Another issue encountered at this stage of the data collection was that some beers do not appear in Untappd's database. For those beers that did not appear on Untappd, a value of "None" was assigned to *description, rating* and *popularity*. This lack of information for many of the beers in the dataset has a negative impact on the graph's power to recommend similar beers to users, for example.

Below, the algorithm used to find the closest match for any given string using the Lehvenstein distance is given:

```
# find the best match for a string using Lehvenstein distance
let s1, s2 = String, String
m, n = len(s1), len(s2)
dp = [[0] * (n + 1) for i in range(m + 1)]

for i in range(m + 1):
    for j in range(n + 1):
        if i == 0:
            dp[i][j] = j  # cost of inserting j characters
        elif j == 0:
            dp[i][j] = i  # cost of deleting i characters
        else:
            cost = 0 if s1[i - 1] == s2[j - 1] else 1
            dp[i][j] = min(dp[i - 1][j] + 1,         # deletion
                           dp[i][j - 1] + 1,         # insertion
                           dp[i - 1][j - 1] + cost)  # substitution

dist = dp[m][n]
max_len = max(m, n)
similarity = (1 - dist / max_len) * 100 if max_len > 0 else 100
```

4.2 Designing and Building the Knowledge Graph

An OWL ontology was designed to represent beer. The ontology aims to capture characteristics of each of the features of the dataset and furthermore capture complex relationships between these features. To this effect, the ontology is especially focused on following the *schema.org* where possible. This ensures any further integration of the project with web sources is feasible,

and simplified to some extent. Before this could be done, however, it was necessary to translate a lot of the data from the table from French to English. For this task, OpenAI's GPT-4-Turbo was used to automate the process. The LLM was queried and asked to translate columns "Type" and normalize it. Additionally, the "Description" column was processed using the LLM to extract precise keywords, which helps with the simplicity of the graph.

The follwing classes were defined: *brewedBy, locatedInProvince, locatedInMunicipality, ownedBy, hasBeerType, hasLocation, containsMunicipality* and *isContainedInProvince*. The data properties which serve as attributes of classes include: *beerName, alcoholPercentage, beerRating, beerDescription, beerPopularity, breweryName, provinceName, municipalityName, parentCompanyName, beerTypeName, latitude* and *longitude*.

As mentioned previously, to align with web standards and facilitate future work on the project, the ontology this graph implements relies on *schema.org*[15]. As such, *Beer* is a subclass of *schema:BreweryProduct*[1], *Brewery* is a subclass of *schema:Brewery*, *Province* is a subclass of *schema:AdministrativeArea*, *Municipality* is a subclass of *schema:city*, *ParentCompany* is a subclass of *schema:Organization*, and *Location* is a subclass of *schema:GeoCoordinates*.

The knowledge graph was built using Python's RDFlib[13] alongside Pandas[11]. A Python script was written to parse the CSV file containing the data and populate the graph following the ontology defined above. All strings and floats were converted to *xsd:string* and *xsd:float*. The resulting graph was serialized into a Turtle file.

5 Results

In total, 1855 unique beers were added to the graph. Of these beers, many are missing data (around 1111 are missing a description, for example). The missing data results from errors during the scraping phase: sometimes, no match was found for a particular beer, and other times information was missing from the websites. Some example queries are provided in a separate file submitted with this document. To demonstrate the power of this graph, some outputs to those queries are presented in figures 1 to 4. Figure 1 shows that this graph makes it possible for users to return N results (based on any metric, in this case alcohol content) easily. Figure 2 displays the ten largest breweries; it highlights the graph's power to provide novel insights on Belgian beer. Figure 3 demonstrates it is possible to get insights on entire provinces, but this can also be done on individual municipalities too. Finally, figure 4 displays the most popular beer type (as in most commonly brewed) per province, an interesting statistic.

Concerning metrics more specific to the graph itself, the work of Smith et al. (2017)[16] was used for inspiration. As such, *average node degree* and *sparsity* were computed for this graph. This graph has an average node degree of 4.88, and a sparsity of 0,99. These numbers suggest the graph does not have many edges, therefore more analysis should be done in this regard. In total, the graph has 21880 triples, and 8959 unique nodes. The graph is missing some data: 15 entries for *Alcohol Content*, 629 for *Rating*, 561 for *Check-ins* and 1 coordinate. For other columns, counts are not reported because they are logical (like the absence of a parent company).

[1] *schema:Food* was not used here, because this graph's focus isn't on nutrition

Beer Name	Alcohol Percentage
Black Damnation V	26.0
Cuvée du Flo Ambrée	15.0
Black Albert	13.0
Black Damnation I	13.0
Black Damnation III	13.0
Black Damnation IV	13.0
Bush Prestige	13.0
Bush 7	13.0
La Cambre	13.0
Cuvée du Château	13.0

Figure 1: Top 10 Beers by Alcohol Percentage

Brewery Name	Number of Beers
De Proefbrouwerij	111
Brasserie De Hoorn	57
Brasserie Den Herberg	54
Brasserie du Bocq	54
Brasserie Huyghe	50
Brasserie Strubbe	41
Brasserie Alvinne	35
Brasserie De Graal	31
Brasserie Haacht	30
Alken-Maes	28

Figure 2: Top 10 Breweries with the Most Beers

Province	Number of Breweries
Hainaut	33
West Flanders	30
Flemish Brabant	28
Luxembourg	24
East Flanders	24
Liege	17
Antwerp	17
Limburg	14
Walloon Brabant	11
Namur	11
Brussels	5

Figure 3: Number of Breweries per Province

Province	Most Popular Beer Type
Hainaut	Blonde Ale
West Flanders	Tripel
Flemish Brabant	High Fermentation
Luxembourg	High Fermentation
East Flanders	Blonde Ale
Liege	Blonde Ale
Antwerp	Blonde Ale
Limburg	Tripel
Walloon Brabant	Blonde Ale
Namur	Blonde Ale
Brussels	Geuze

Figure 4: Most Popular Beer Type per Province

6 Discussion

6.1 Answering the Research Questions

The main objective of this project was to improve on the existing platforms (Untappd, BeerAdvocate etc.). By looking at the example queries of section 4, it is apparent that this knowledge graph allows for the extraction of more interesting insights than the other platforms. The limitations of the current platforms restrict users to very basic searches; they have no function to get beers per region, nor do they provide a way to obtain ordered (top N) lists of results, for example. This knowledge graph provides better insights than those platforms: It is possible to compute average ratings per brewery or beer type, return results from multiple breweries, or even analyze beers and breweries per individual (or multiple at once!) regions and provinces. Information on parent companies is also provided in this graph. One area in which this graph does not improve upon the aforementioned platforms is that it does not incorporate any user reviews, nor does it provide users with pictures of the beers they wish to learn about. Additionally, using SPARQL to query the graph alienates a large portion of the public which does not possess the technical skills to do so.

6.2 Challenges Faced and Recommendations for Future Work

This report successfully improved upon the work of the popular platforms such as Untappd and BeerAdvocate by providing users with more powerful queries, enabling the extraction of more detailed and relevant insights than the aforementioned platforms. What it did not do, however, was add more information to the existing base of knowledge. Attributes such as ingredients, price, or even type of drinking container (for those situations where the beer is purchased at a store) would all be relevant and useful additions to the knowledge graph, and could help this project to become a more scientific one. By incorporating a detailed list of ingredients to each beer in the graph, individual consumers could check for eventual allergens when searching for a beer, as well as indicators such as calories of the beverage, for example. Parties interested in nutritional analysis could derive insights based on this list of ingredients and nutritional values. Finally, breweries themselves could benefit the most from such a graph, as they could analyze the different ingredients and their impact on the consumer ratings of their beers, as well as the impact of using certain ingredients on their production costs. Such data must be gathered either by compiling the relevant information as a store which sells beer (for each individual one, quite a lengthy and tedious task), or from the breweries themselves. While the potential impact of adding such attributes to the knowledge graph might not be useful for most consumers, it would both enable the extraction or more detailed data from the graph, and it would mark the first step towards a digitization of a Belgian beer archive.

Future contributors should also be mindful of the completion of the knowledge graph. Currently, the graph has data regarding only a little more than 1800 beers. The number of beers brewed in Belgium is well above 1800, and the number of beers brewed across the world is likely to be much higher than that, which highlights a need for future contributors to gather more data. Additionally, the graph currently lacks a lot of the data regarding consumer reviews. As those are hard to find, this project only incorporated the data which Untappd made available on their platform, but many statistics are still missing.

The addition of multiple user reviews to the graph would enable researchers to use LLMs (or other tools) to derive flavor profiles for each beer, a welcome addition to the graph for any enthusiast. Many other attributes could be derived from user reviews using LLMs, and would help enrich the graph in such a way humans could not do reliably.

7 Conclusion

This project presents a successful proof-of-concept knowledge graph on Belgian beers, highlighting the possibilities of compiling data from different sources to provide interesting information about beer. The graph enables users to write more complex queries than the current popular platforms allow. Although many issues were encountered, such as lack of data and difficulties in collecting data, the outcome of the project serves as a first step in the digitization of Belgian beer archives. Future work should focus on incorporating nutritional data associated with beer, and enriching the graph with keywords derived from user reviews using LLMs. The above would enhance the graph's utility for consumers, enthusiasts, researchers and breweries. Embeddings can also be explored to turn this project into a true recommender system.

References

[1] *BeerAdvocate*. English. URL: https://www.beeradvocate.com/beer/.

[2] Valentina Anita Carriero et al. 'ArCo: The Italian Cultural Heritage Knowledge Graph'. en. In: *The Semantic Web – ISWC 2019*. Ed. by Chiara Ghidini et al. Cham: Springer International Publishing, 2019, pp. 36–52. ISBN: 9783030307967. DOI: 10.1007/978-3-030-30796-7_3.

[3] GeoPy Contributors. 'GeoPy Documentation'. In: (2014).

[4] Damion M. Dooley et al. 'FoodOn: a harmonized food ontology to increase global food traceability, quality control and data integration'. en. In: *npj Science of Food* 2.1 (Dec. 2018), p. 23. ISSN: 2396-8370. DOI: 10.1038/s41538-018-0032-6. URL: https://www.nature.com/articles/s41538-018-0032-6 (visited on 10/03/2025).

[5] Jinhua Dou et al. 'Knowledge graph based on domain ontology and natural language processing technology for Chinese intangible cultural heritage'. In: *Journal of Visual Languages & Computing* 48 (Oct. 2018), pp. 19–28. ISSN: 1045-926X. DOI: 10.1016/j.jvlc.2018.06.005. URL: https://www.sciencedirect.com/science/article/pii/S1045926X18300041 (visited on 10/03/2025).

[6] Untappd Inc and The Untappd Team. *Untappd*. en. URL: https://untappd.com/ (visited on 10/03/2025).

[7] *Le macroblues des microbrasseries ? Pour la première fois en vingt ans, leur nombre diminue en Belgique*. fr. URL: https://www.rtbf.be/article/le-macroblues-des-microbrasseries-pour-la-premiere-fois-en-vingt-ans-leur-nombre-diminue-en-belgique-11311371 (visited on 10/03/2025).

[8] Vladimir I Levenshtein et al. 'Binary codes capable of correcting deletions, insertions, and reversals'. In: *Soviet physics doklady*. Vol. 10. 8. Soviet Union. 1966, pp. 707–710.

[9] *Liste de bières belges*. fr. Page Version ID: 220932519. Dec. 2024. URL: https://fr.wikipedia.org/w/index.php?title=Liste_de_bi%C3%A8res_belges&oldid=220932519 (visited on 10/03/2025).

[10] *Liste de brasseries belges*. fr. Page Version ID: 223749666. Mar. 2025. URL: https://fr.wikipedia.org/w/index.php?title=Liste_de_brasseries_belges&oldid=223749666 (visited on 10/03/2025).

[11] Wes McKinney et al. 'pandas: a foundational Python library for data analysis and statistics'. In: *Python for high performance and scientific computing* 14.9 (2011), pp. 1–9.

[12] Mark A. Musen. 'The protégé project: a look back and a look forward'. en. In: *AI Matters* 1.4 (June 2015), pp. 4–12. ISSN: 2372-3483. DOI: 10.1145/2757001.2757003. URL: https://dl.acm.org/doi/10.1145/2757001.2757003 (visited on 12/03/2025).

[13] *rdflib 7.1.3 — rdflib 7.1.3 documentation*. URL: https://rdflib.readthedocs.io/en/stable/ (visited on 11/03/2025).

[14] Leonard Richardson. *Beautiful soup documentation*. 2007.

[15] *Schema.org - Schema.org*. URL: https://schema.org/ (visited on 11/03/2025).

[16] Jeffrey A Smith, James Moody and Jonathan H Morgan. 'Network sampling coverage II: The effect of non-random missing data on network measurement'. In: *Social networks* 48 (2017), pp. 78–99.

[17] *The Selenium Browser Automation Project*. en. URL: `https://www.selenium.dev/documentation/` (visited on 10/03/2025).

8 Reflection on the Usage of LLMs in this Project

Throughout the web scraping part of this project, I used an LLM (Gemini 2.0 Flash) to assist me with some code trouble shooting. As web scraping is not my expertise, I struggled to gather the data I needed from Untappd, which explains the sudden switch to Selenium instead of BeautifulSoup. After reading the documentation of Selenium, I was able to scrape some data on my own, but some elements of Untappd such as its consent form were very problematic for me. Elements of the code, particularly "read_more_button" in the "get_data" function, were out of my skillset, and given I needed to obtain this data for the completion of this project, I ended up asking Gemini for a way to bypass this consent form, which it gave me, as well as a general structure for this function (e.g., how to find certain elements). Once I had that, everything else was quite straightforward. Overall, this is not my preferred approach, as I would usually read the documents fully in such a scenario. In this case however, the time constraint and my lack of knowledge about web scraping in general made me resort to using such a tool. Using an LLM to help with parts of my code does come with responsibilities for the user: there are ways in which LLMs can make users rely on them too much and not learn anything from the course. This however is not what I did, as I only used it to fix some mistakes in my code.

I also used Gemini for recommendations on tools to use when building my ontology and for recommendations regarding data sources at the start of this project. One such tool recommendation was Protégé[12], which I used to visualize and design the ontology used in this graph. This tool was quite useful to model the hierarchy and necessary attributes for each class I created.