UNIVERSITY OF SOUTHAMPTON

Faculty of Engineering, Science, and Mathematics
School of Electronics and Computer Science

# Multilinguality in Knowledge Graphs

*by*

**Lucie-Aimée Kaffee**

*A thesis for the degree of*
*Doctor of Philosophy*

October 2021

University of Southampton

Abstract

Faculty of Engineering, Science, and Mathematics
School of Electronics and Computer Science

Doctor of Philosophy

**Multilinguality in Knowledge Graphs**

by Lucie-Aimée Kaffee

Content on the web is predominantly in English, which makes it inaccessible to individuals who exclusively speak other languages. Knowledge graphs can store multilingual information, facilitate the creation of multilingual applications, and make these accessible to more language communities. In this thesis, we present studies to assess and improve the state of labels and languages in knowledge graphs and apply multilingual information. We propose ways to use multilingual knowledge graphs to reduce gaps in coverage between languages.

We explore the current state of language distribution in knowledge graphs by developing a framework - based on existing standards, frameworks, and guidelines - to measure label and language distribution in knowledge graphs. We apply this framework to a dataset representing the web of data, and to Wikidata. We find that there is a lack of labelling on the web of data, and a bias towards a small set of languages. Due to its multilingual editors, Wikidata has a better distribution of languages in labels. We explore how this knowledge about labels and languages can be used in the domain of question answering. We show that we can apply our framework to the task of ranking and selecting knowledge graphs for a set of user questions

A way of overcoming the lack of multilingual information in knowledge graphs is to transliterate and translate knowledge graph labels and aliases. We propose the automatic classification of labels into transliteration or translation in order to train a model for each task. Classification before generation improves results compared to using either a translation- or transliteration-based model on their own.

A use case of multilingual labels is the generation of article placeholders for Wikipedia using neural text generation in lower-resourced languages. On the basis of surveys and semi-structured interviews, we show that Wikipedia community members find the placeholder pages, and especially the generated summaries, helpful, and are highly likely to accept and reuse the generated text.

# Contents

**List of Figures** xi

**List of Tables** xv

**Listings** xvii

**Declaration of Authorship** xix

**Acknowledgements** xxi

**Definitions** xxiii

**1 Introduction** 1
- 1.1 Motivation . . . . . 4
  - 1.1.1 Question Answering . . . . . 5
  - 1.1.2 ArticlePlaceholder for Wikipedia . . . . . 6
- 1.2 Research Questions . . . . . 7
- 1.3 Context . . . . . 11
- 1.4 Structure of the Thesis . . . . . 13
- 1.5 Previous Publications by the Author . . . . . 14

**2 Background** 15
- 2.1 Web of Data . . . . . 15
  - 2.1.1 Datasets . . . . . 17
- 2.2 Knowledge Graphs . . . . . 18
- 2.3 Labels and Languages in Knowledge Graphs . . . . . 18
  - 2.3.1 Guidelines . . . . . 19
  - 2.3.2 Use Cases . . . . . 20
- 2.4 Wikidata . . . . . 20
  - 2.4.1 Labels and Languages . . . . . 22
- 2.5 Knowledge Graph Transliteration and Translation . . . . . 24
  - 2.5.1 Transliteration . . . . . 24
  - 2.5.2 Knowledge Graph Translation . . . . . 25
  - 2.5.3 Knowledge Graph Embeddings . . . . . 25
- 2.6 Background on the Motivating Scenarios . . . . . 26
  - 2.6.1 Multilingual Question Answering . . . . . 26
  - 2.6.2 Wikipedia . . . . . 27
  - 2.6.3 Natural Language Generation from Knowledge Graph Data . . . 28

2.6.3.1 Evaluating Text Generation Systems . . . . . . . . . . . 30

**3 Framework to measure labels and languages 33**
3.1 Dataset Description . . . . . . . . . . . . . . . . . . . . . . . . 35
3.2 Core Metrics . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37
3.3 Data Creator Metrics . . . . . . . . . . . . . . . . . . . . . . . 40
3.4 Application of the framework . . . . . . . . . . . . . . . . . . . 43

**4 Multilinguality in the Web of Data 45**
4.1 Dataset . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 45
4.2 Size . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 45
4.3 Natural Language URIs . . . . . . . . . . . . . . . . . . . . . . 46
4.4 Labelling Properties . . . . . . . . . . . . . . . . . . . . . . . . 47
4.5 Entity Labelling . . . . . . . . . . . . . . . . . . . . . . . . . . 48
4.6 Class Labelling . . . . . . . . . . . . . . . . . . . . . . . . . . 48
4.7 Property Labelling . . . . . . . . . . . . . . . . . . . . . . . . . 48
4.8 Unambiguity . . . . . . . . . . . . . . . . . . . . . . . . . . . . 49
4.9 Multilinguality . . . . . . . . . . . . . . . . . . . . . . . . . . . 49
4.10 Monoglingual Islands . . . . . . . . . . . . . . . . . . . . . . . 50
4.11 Contextual Comparison . . . . . . . . . . . . . . . . . . . . . . 51
4.12 Discussion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51

**5 Multlinguality in Wikidata 53**
5.1 Editors in Wikidata . . . . . . . . . . . . . . . . . . . . . . . . 54
5.2 Datasets . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55
5.2.1 Wikidata's Entities . . . . . . . . . . . . . . . . . . . . . 55
5.2.2 Editor Data . . . . . . . . . . . . . . . . . . . . . . . . . 55
5.3 Size . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 58
5.4 Natural Language URI . . . . . . . . . . . . . . . . . . . . . . . 58
5.5 Labelling properties . . . . . . . . . . . . . . . . . . . . . . . . 59
5.6 Entity Labelling . . . . . . . . . . . . . . . . . . . . . . . . . . 59
5.7 Class Labelling . . . . . . . . . . . . . . . . . . . . . . . . . . 59
5.8 Property Labelling . . . . . . . . . . . . . . . . . . . . . . . . . 59
5.9 Unambiguity . . . . . . . . . . . . . . . . . . . . . . . . . . . . 59
5.10 Multilinguality . . . . . . . . . . . . . . . . . . . . . . . . . . . 60
5.11 Monolingual Islands . . . . . . . . . . . . . . . . . . . . . . . . 61
5.12 Contextual Comparison . . . . . . . . . . . . . . . . . . . . . . 62
5.13 User Language . . . . . . . . . . . . . . . . . . . . . . . . . . . 63
5.14 User Language Editing . . . . . . . . . . . . . . . . . . . . . . 65
5.15 User Activity . . . . . . . . . . . . . . . . . . . . . . . . . . . . 67
5.16 Edit Patterns . . . . . . . . . . . . . . . . . . . . . . . . . . . . 70
5.17 Language Overlap . . . . . . . . . . . . . . . . . . . . . . . . . 71
5.18 Activity and Multilinguality . . . . . . . . . . . . . . . . . . . . 71
5.19 Comparison by User Group . . . . . . . . . . . . . . . . . . . . 74
5.20 Comparison to the Web of Data . . . . . . . . . . . . . . . . . . 76
5.21 Discussion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 77

**6 Ranking Knowledge Graphs 79**
6.1 Problem statement . . . 80
6.2 Capturing Knowledge in LINGVO . . . 83
6.2.1 Representing Labelling and Multilinguality of Knowledge Graphs 83
6.2.2 Framework for Labelling and Multilinguality . . . 84
6.2.3 Ranking Knowledge Graphs by Queries and Languages . . . 84
6.3 Datasets . . . 85
6.4 Gold Standard . . . 86
6.5 Baselines . . . 87
6.6 Experiments . . . 88
6.6.1 Random Domains . . . 89
6.6.2 Gold-standard Domains . . . 89
6.7 Results . . . 90
6.7.1 Random Domains . . . 90
6.7.2 Gold-standard Domains . . . 91
6.8 Discussion . . . 92

**7 Transliteration and Translation of Knowledge Graph Labels 95**
7.1 Problem Statement . . . 97
7.2 Approach . . . 97
7.3 Crowdsourcing Task . . . 98
7.4 Automatic Transliteration/Translation Classification . . . 99
7.4.1 Rule Mining . . . 100
7.4.2 Knowledge Graph Embeddings . . . 100
7.5 Out-of-domain Training Data . . . 102
7.5.1 Transliteration Data . . . 102
7.5.2 Translation Data . . . 102
7.5.3 Pinyin Data . . . 102
7.6 Evaluation and Models . . . 103
7.7 Transliteration and Translation Model Setup . . . 104
7.7.1 Simplified Chinese . . . 104
7.7.2 Transliteration Model . . . 105
7.7.3 Translation Model . . . 105
7.8 Transliteration and Translation Based on Gold Standard . . . 106
7.9 Transliteration and Translation Based on Automated Classification . . . 107
7.10 Discussion . . . 107

**8 ArticlePlaceholder from Wikidata for Wikipedia 111**
8.1 Bootstrapping empty Wikipedia articles . . . 113
8.1.1 ArticlePlaceholder . . . 114
8.1.2 Text generation . . . 114
8.2 Methods for Answering the Research Questions . . . 115
8.3 RQ4.1 - Judgement-based evaluation . . . 116
8.3.1 Recruitment . . . 116
8.3.2 Participants . . . 117
8.3.2.1 Ethics . . . 117
8.3.3 Data . . . 118

8.3.4 Metrics . . . 118
8.3.5 Results: RQ4.1 - Judgement-based evaluation . . . 118
8.3.5.1 Fluency . . . 118
8.3.5.2 Appropriateness . . . 120
8.4 RQ4.2 and RQ4.3 - Task-based evaluation . . . 120
8.4.1 Recruitment . . . 121
8.4.2 Participants . . . 122
8.4.3 Ethics . . . 123
8.4.4 Data . . . 123
8.4.5 Task . . . 123
8.4.6 Analysing the interviews . . . 124
8.4.7 Editors' reuse metric . . . 124
8.4.8 Results: RQ4.2 - Task-based evaluation . . . 125
8.4.9 Results: RQ4.3 - Task-based evaluation . . . 128
8.5 Discussion . . . 130
8.6 Limitations . . . 132

**9 Conclusions 135**
9.1 Contributions and Results . . . 135
9.1.1 RQ1 What is the state of knowledge graphs with regard to labels and multilinguality? . . . 136
9.1.2 RQ2 How can knowledge about languages in a knowledge graph be applied to the task of ranking them for question answering? . 137
9.1.3 RQ3 How does differentiating between translation and transliteration impact the generation of new knowledge graph labels and aliases? . . . 137
9.1.4 RQ4 How do Wikipedia editors perceive automatically generated Wikipedia summaries? . . . 138
9.2 Future Work . . . 138
9.2.1 Increasing knowledge graph language coverage . . . 139
9.2.2 Leveraging multilingual knowledge graph information . . . 140
9.3 Final remarks . . . 141

**Appendix A Labelling Properties in LOD Laundromat 143**

**Appendix B Crowdsouring Guidelines for Annotation of the Knowledge Graph Ranking 145**
Appendix B.1 Overview . . . 145
Appendix B.2 Steps . . . 145
Appendix B.3 Rules & Tips . . . 145
Appendix B.4 Examples . . . 146
Appendix B.4.1 Example (1): Too many ambiguous answers . . . 146
Appendix B.4.2 Example (2): Repetition of the same answer . . . 146
Appendix B.4.3 Example (3): List of answers . . . 146

**Appendix C Guidelines for the semi-structured interview 149**

**Appendix D Sentences editors created in the interviews and English translations. 151**

**References 153**

# List of Figures

1.1 Example of a triple (*Ada Lovelace - occupation - computer scientist*) with the respective label for each entity and the property in English (en) and Arabic (ar). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

1.2 Representation of Figure 1.1 in triples format. The `rdfs:label` property connects opaque URIs of the entity to their natural language representations in English and Arabic. . . . . . . . . . . . . . . . . . . . . . . . . . . 3

1.3 Motivating Example. (1) User queries, processed by a question answering system, in three different languages. (2) Three different knowledge graphs that are used by the QA systems independently to find an answer to the user question. (3) The collected answers. If there is a lack of data in the language to disambiguate and answer the question, the answer is empty (“”), if there is more than one label for the returned entity, the answer contains multiple, independent words. The second knowledge graph (KG2) is therefore most appropriate for English and German, the third knowledge graph (K3) is most appropriate for Spanish. . . . . . . . 4

1.4 Wikipedia language versions and size in articles, based on `https://en.wikipedia.org/wiki/List_of_Wikipedias`, retrieved 28 April 2021 . . 6

1.5 Research questions posed in this thesis, contributions, methods, and datasets for each research question. . . . . . . . . . . . . . . . . . . . . . . . . . . 7

2.1 Linked Open Data cloud visualising the interconnections between different knowledge graphs. CC-BY `https://lod-cloud.net/`, retrieved 9. October 2020. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16

2.2 Wikidata’s data model. CC-0 Charlie Kritschmar, `https://commons.wikimedia.org/wiki/File:Datamodel_in_Wikidata.svg` . . . . . . . . . . 21

2.3 Example of labelling in Wikidata. Each entity can be labelled in multiple languages using the labelling property `rdfs:label`. . . . . . . . . . . . . 22

2.4 Representation of Wikidata statements and their inclusion in a Wikipedia infobox. Wikidata statements in French (middle, English translation to their left) are used to fill out the fields of the infobox in articles using the *fromage* infobox on the French Wikipedia. . . . . . . . . . . . . . . . . . . 28

3.1 Metrics and the respective chapters they are applied in. Metrics are used and applied based on the need of the respective chapter. . . . . . . . . 44

4.1 Language distribution in the LOD laundromat dataset, including labels that do not have a language code (no language). English is the most prominent language on the web of data, followed by German (de), French (fr), and Italian (it). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 49

4.2 Comparison of language distribution of the LOD Laundromat and native speakers in the world. . . . . . . . . . . . . . . . . . . . . . . . . . 51

5.1 Example for a BabelBox of User:Frimelle . . . . . . . . . . . . . . . . . . . 56
5.2 Percentage of all labels per language in Wikidata . . . . . . . . . . . . . . 60
5.3 Distribution of languages for properties in Wikidata . . . . . . . . . . . . 60
5.4 Comparison of distribution of languages in Wikidata and first language speakers in the world . . . . . . . . . . . . . . . . . . . . . . . . . . . 62
5.5 Language Distribution in User Language Setting, excluding English . . . 64
5.6 Language Distribution in BabelBox, Native Speakers . . . . . . . . . . . 64
5.7 Language Distribution in BabelBox, excluding language levels 0 and 1 . 66
5.8 Users plotted according to their label edits in languages they are familiar with (known) and unknown languages . . . . . . . . . . . . . . . . . . . 67
5.9 Users plotted according to their label edits in languages they are familiar with (known) and unknown languages, excluding most extreme outliers 67
5.10 Measuring the distribution of multilingual editors: each editor type is represented by one bar, and split by the number of languages they edit. The majority of editors edit in one language. . . . . . . . . . . . . . . . . 69
5.11 Timeline of number of edits (log) of the three different editor groups from January 2013 to March 2019. Edits are aggregated by month. The highest number of edits for registered users is in October 2016, for bots October 2014, and for anonymous users in September 2018. . . . . . . . . . . . . . 69
5.12 Language distribution over the three different editor groups, sorted by number of edits, including language ordering by number of editor in that language . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 70
5.13 Boxplot comparing the number of edits in languages of the same and different language families. . . . . . . . . . . . . . . . . . . . . . . . . 72
5.14 Displaying the connections between languages, where the number of connections is greater than the average. Nodes are colored by language family. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 73
5.15 Scatter plot of the number of languages and the number of edits, testing correlation for all users. . . . . . . . . . . . . . . . . . . . . . . . . . . 75
5.16 Comparison of language coverage between native speakers in the world, Wikidata, and the web of data (LOD Laundromat, removing labels without language tag). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 76

6.1 LINGVO approach. For each knowledge graph, classes for the required SPARQL query are extracted. Then, Class-based Label Captures (CLCs) are matched to each extracted class with respect to the language provided in the required SPARQL queries. The respective scores of the CLCs are aggregated to finally rank the knowledge graphs. In this example, DBpedia is ranked highest and therefore recommended. . . . . . . . . . 80
6.2 Boxplots of classes used for QALD. Scores are between 0 and 1; a higher variety in the scores between different classes can be observed with a higher number of classes overall. . . . . . . . . . . . . . . . . . . . . . . 90

6.3 Random domains ($D1 - D5$). Results for correlation between the gold standard ranking to computed ranking. Each domain consists of 20 randomly selected queries from the QALD dataset. Three baseline ranking approaches, **CosN**, **Cos**, and **NoCLC** and the LINGVO approach, i.e., **CLC**. The addition of CLCs leads to higher scores in the ranking over all domains. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 91

6.4 Selected domains. Results for correlation between the gold standard ranking to three baseline ranking approaches, **CosN**, **Cos**, and **NoCLC** and the LINGVO approach, i.e., **CLC**. The domains are contextual. The addition of CLCs leads to higher scores in the ranking over all domains. 92

7.1 Overview of the contributions in this chapter. (1) Conversion from traditional to simplified Chinese (zh). (2) Crowdsourcing to differentiate between translation and transliteration in the entities. (3) Training two models, one for transliteration and one for translation, on out of domain data. (4) Training the classifier on the entities classified as transliteration or translation by the crowd. (4) Overview of the final pipeline. . . . . . . 96

7.2 Approach: given an entity $s$, its English label, and its respective triples ($G_s$), we classify it as either translation or transliteration. With this knowledge, we can run it through a model which is specialised in either translation or transliteration, and trained specifically for that task. The resulting Chinese-language alias will be checked for correctness. . . . . . . . . 97

7.3 Crowdsourcing results. Number of (English label, Chinese alias) pairs categorised as Translation, Transliteration, Partial or None, and their percentage. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 99

7.4 TSNE scatter plot of entities used for the classification; blue for translation, orange for transliteration as annotated by the crowd. . . . . . . . . 101

8.1 Example page of the ArticlePlaceholder as deployed now on 14 Wikipedias. This example contains information from Wikidata on Triceratops in Haitian-Creole. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 113

8.2 Representation of the neural network architecture. The triple encoder computes a vector representation for each one of the three input triples from the ArticlePlaceholder, $h_{f_1}$, $h_{f_2}$ and $h_{f_3}$. Subsequently, the decoder is initialised using the concatenation of the three vectors, $[h_{f_1}; h_{f_2}; h_{f_3}]$. The purple boxes represent the tokens of the generated text. Each snippet starts and ends with special tokens: start-of-summary `<start>` and end-of-summary `<end>`. Example is in Esperanto. . . . . . . . . . . . . . . . 115

8.3 Example of a question to readers about quality and appropriateness for Wikipedia of the generated summaries in Arabic. They see this page after the instructions are displayed. First, the user is asked to evaluate the quality from 0 to 6 (Question 69), then they are asked whether the sentence could be part of Wikipedia (Question 70). The sentence to evaluate has a grey background. English translation in Figure 8.4 . . . . . . . . . . 116

8.4 English translation of Figure 8.3 . . . . . . . . . . . . . . . . . . . . . . . 117

8.5 Screenshot of the reading task. The page is stored as a subpage of the author's user page on Wikidata, therefore the layout copies the original layout of any Wikipedia. The layout of the information displayed mirrors the ArticlePlaceholder setup. The participants sees the sentence to evaluate alongside information included from the Wikidata triples (such as the image and statements) in their native language (Arabic in this example). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 119

8.6 Screenshot of the reading task as in Figure 8.5, translated to English. . . 119

8.7 Screenshot of the editing task, with annotations. The page is stored on a subpage of the author's user page on Wikidata, so the layout is equivalent to the current MediaWiki installations on Wikipedia. The participants see the sentence (in green) that they saw before in the reading task, and the triples from Wikidata in their native language (Arabic in this example, in purple). The triples are manually added to the page by the researchers for easier interaction with the data by the editor. The data is the same data as in the reading task (Figure 8.5). . . . . . . . . . . . . . 121

Appendix D.1 Sentences editors created in the interviews and English translations. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 152

# List of Tables

3.1 Core metrics in the framework and literature the metrics are based on. . 35
3.2 Overview of all metrics in the framework. . . . . . . . . . . . . . . . . . 36

4.1 The 20 most frequently used labelling properties used in the LOD Laundromat dataset, manually selected. The most used labelling property is rdfs:label. Full list in Appendix A. . . . . . . . . . . . . . . . . . . . . . . . 47
4.2 Results for the entity, class and property labelling. . . . . . . . . . . . . 48
4.3 Share of entities having labels in multiple (1, 2, 2-5, 5-10, over 10) languages over 127, 244, 182 entities. . . . . . . . . . . . . . . . . . . . . . . 50

5.1 Share of entities having labels in one or multiple (2, 2-5, 5-10, over 10) languages. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 61
5.2 Ranking of number of Wikipedia articles by language, all labels in Wikidata, and labels for properties in Wikidata . . . . . . . . . . . . . . . . . 63
5.3 Distribution of multilingual users . . . . . . . . . . . . . . . . . . . . . . 65
5.4 Results of the analyses of Wikidata's editing history of 2019 for the user activity metric. The total number of editors is highest for anonymous editors; their average edit per editor is lowest however. Bots are the smallest group of editors, but have the highest number of average edit per editor. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 68
5.5 Bots with the highest numbers of languages edited . . . . . . . . . . . . 68
5.6 Average number of *jumps* between languages and edits for all three user groups. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 71
5.7 Comparison of values for each metric between the web of data (represented by the LOD Laundromat dataset) and Wikidata. . . . . . . . . . . 76

6.1 Metrics defined in Chapter 3. Metrics that are applied over the whole dataset are denoted with *dataset*, all other metrics are applied per CLC. . 84
6.2 English example of answers displayed to annotators for the question *Who is the president of the United States of America?*. The annotator has to select between the four knowledge graphs the one that fits the question best. In this example, an annotator should select knowledge graph (1). For more details, see Appendix B. . . . . . . . . . . . . . . . . . . . . 86
6.3 CLCs for each KG covered by QALD questions and number of QALD questions translated to the respective ontology. DBpedia has the largest numbers of classes and average number of classes per QALD questions, as well as the largest maximum number of classes for one QALD question. 88

7.1 Classification of the entities as translation or transliteration, using knowledge graph embeddings-based classifiers. . . . . . . . . . . . . . . . . . . 99

7.2 Size of the datasets used in the transliteration experiments (transliteration data and pinyin data). Duplicates are removed in the total transliteration dataset. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 102
7.3 Experiment comparing training on the original people names dataset, the dataset with only simplified Chinese and the dataset converted to simplified Chinese, reported with BLEU-4. Overall the model converted to simplified Chinese performs best on both the dataset converted to simplified Chinese and the dataset only simplified Chinese. . . . . . . . . . . 104
7.4 Experiment comparing running over the dataset including pinyin data and the person names dataset only. Training and validation on the original datasets, test on company names, evaluated with BLEU score. . . . . 105
7.5 Results for out-of-domain data, i.e., train on people names/WMT and test on the companies annotated by the crowd, BLEU-1/BLEU-4 (top), CER (middle), WER (bottom) scores. . . . . . . . . . . . . . . . . . . . . . 106
7.6 Results for aliases generated after automated classification as transliteration or translation, BLEU-1/BLEU-4 (top), CER (middle), WER (bottom) scores. The translation and transliteration models perform better on the entities classified as translation and transliteration respectively, while both models perform better with the classification than the companies overall. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 107

8.1 Page statistics and number of unique words (vocabulary size) of Esperanto, Arabic and English Wikipedias in comparison with Wikidata. Retrieved 27 September 2017. Active users are registered users that have performed an action in the last 30 days. . . . . . . . . . . . . . . . . . . . 114
8.2 Evaluation methodology . . . . . . . . . . . . . . . . . . . . . . . . . . . 115
8.3 Judgement-based evaluation: total number of participants ($P$), total number of sentences ($S$), number of participants who evaluated at least 50% of the sentences, average and mean number of sentences evaluated per participant, and number of total annotations . . . . . . . . . . . . . . . . 117
8.4 Number of Wikipedia articles; active editors on Wikipedia (editors that performed at least one edit in the last 30 days); and number of native speakers in Million for each language. . . . . . . . . . . . . . . . . . . . . 120
8.5 Results for fluency and appropriateness. . . . . . . . . . . . . . . . . . . . 120
8.6 Overview of sentences and number of participants in each language . . 122
8.7 Number of snippets in each category of reuse. A generated snippet (top) and its edited version (bottom). Solid lines represent reused tiles, while dashed lines represent overlapping sub-sequences not contributing to the *gstscore*. The first two examples are created in the studies, the last one (ND) is from a previous experiment. A list of all created sentences and their translations to English can be found in Appendix D. . . . . . . 128

Appendix A.1 Properties used across the LOD cloud for labelling, manually selected. The most used labelling property is rdfs:label. . . . . . . . . . . 144

Appendix B.1 Example 1 knowledge graph answers to select from . . . . . . 146
Appendix B.2 Example 2 knowledge graph answers to select from . . . . . . 146
Appendix B.3 Example 3 knowledge graph answers to select from . . . . . . 147

# Listings

# Declaration of Authorship

I declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:
   Lucie-Aimée Kaffee and Elena Simperl. The Human Face of the Web of Data: A Cross-sectional Study of Labels. In *Proceedings of the 14th International Conference on Semantic Systems, SEMANTICS 2018, Vienna, Austria, September 10-13, 2018*, pages 66–77, 2018a
   Lucie-Aimée Kaffee, Alessandro Piscopo, Pavlos Vougiouklis, Elena Simperl, Leslie Carr, and Lydia Pintscher. A Glimpse into Babel: An Analysis of Multilinguality in Wikidata. In *Proceedings of the 13th International Symposium on Open Collaboration*, page 14. ACM, 2017
   Lucie-Aimée Kaffee, Kemele M. Endris, and Elena Simperl. When Humans and Machines Collaborate: Cross-lingual Label Editing in Wikidata. In *Proceedings of the 15th International Symposium on Open Collaboration, OpenSym 2019, Skövde, Sweden, August 20-22, 2019*, pages 16:1–16:9, 2019a. . URL https://doi.org/10.1145/3306446.3340826

Lucie-Aimée Kaffee, Kemele M. Endris, Elena Simperl, and Maria-Esther Vidal. Ranking Knowledge Graphs by Capturing Knowledge about Languages and Labels. In Mayank Kejriwal, Pedro A. Szekely, and Raphaël Troncy, editors, *Proceedings of the 10th International Conference on Knowledge Capture, K-CAP 2019, Marina Del Rey, CA, USA, November 19-21, 2019*, pages 21–28. ACM, 2019b. . URL https://doi.org/10.1145/3360901.3364443

Lucie-Aimée Kaffee, Hady Elsahar, Pavlos Vougiouklis, Christophe Gravier, Frédérique Laforest, Jonathon S. Hare, and Elena Simperl. Learning to Generate Wikipedia Summaries for Underserved Languages from Wikidata. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 640–645, 2018b. URL https://aclanthology.info/papers/N18-2101/n18-2101

Lucie-Aimée Kaffee, Hady Elsahar, Pavlos Vougiouklis, Christophe Gravier, Frédérique Laforest, Jonathon S. Hare, and Elena Simperl. Mind the (Language) Gap: Generation of Multilingual Wikipedia Summaries from Wikidata for ArticlePlaceholders. In Aldo Gangemi, Roberto Navigli, Maria-Esther Vidal, Pascal Hitzler, Raphaël Troncy, Laura Hollink, Anna Tordai, and Mehwish Alam, editors, *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings*, volume 10843 of *Lecture Notes in Computer Science*, pages 319–334. Springer, 2018a. . URL https://doi.org/10.1007/978-3-319-93417-4_21

Lucie-Aimée Kaffee, Pavlos Vougiouklis, and Elena Simperl. Using Natural Language Generation to Bootstrap Missing Wikipedia Articles: A Human-centric Perspective. *Semantic Web*, 2021

Signed

Date:..................

# Acknowledgements

The pronoun *we* is used throughout this thesis as a matter of academic practice. It is also a fit acknowledgement of the collaborative nature of all good research. This thesis has nine chapters that cross at least three different research fields. Such breadth of approach would not have been possible without the collaboration of peers. I would like to express my gratitude to some of those who accompanied me on this journey to a PhD. My thanks, like the web I would like to see going forward, are expressed in multiple languages.

I would especially like to thank:

Hady Elsahar – الرسالة دي مكانتش هتبقى ممكنة من غير دعمك ومساعدتك

The WDAqua network – For supplying the framework for my academic work, but also for creating long-lasting friendships for which I am very grateful.

Elena Simperl – For her support and academic guidance as my supervisor.

Pavlos Vougiouklis – Σε ευχαριστώ που έκανες να νιώσω σπίτι ακόμα και στο Southampton. Πάντα πίστευες σε μένα και τη δουλειά μου. Σε ευχαριστώ που με άφησες να σε τραβήξω στον κόσμο της πολυγλωσσίας.

Alessandro Piscopo – Sono sempre ispirata dal tuo lavoro e dalle ostre conversazioni. È stato un grande piacere condividere parte di questo folle viaggio.

Lydia Pintscher and the Wikidata team – I would like to thank for their support throughout the years. You introduced me to the wonderful world of linked data, and here we are, many many years later. I cannot tell you how excited I still am to make Wikipedia more accessible to speakers of every language. Thanks for helping me keep this excitement alive.

Thomas Pellisier-Tanon – Tu étais l'une des premières personnes à discuter avec moi sur ma thèse. J'apprécie vraiment toutes les conversations que nous avions sur Wikidata.

The research team at TIB Hannover, and in particular Maria-Esther Vidal and Kemele M. Endris – Your kindness and good spirits made working with you a great experience. Gracias y አመሰግናለሁ

David Chaves Fraga – No puedo esperar a compartir mas momentos contigo, tal vez en un balcón tranquilo de una ciudad a la que nunca llegamos a imaginar que vendríamos.

Bloomberg London AI team, in particular Oana Tifrea-Marciuska, Edgar Meij, and Sameer Bansal – I want thank for the collaboration and insight into the world of research in industry. You trusted me with your time and my idea to transliterate everything I possibly could!
John Cummings – Diolch for all the fun chats about Wikipedia, communities, and this thesis. Those very last weeks of the PhD would have been a lot less fun without your encouragement.
Katrina Zaat – Having you proofread this thesis was a great pleasure. Besides learning a lot about English grammar, I finally got a sense of what semicolons are for; thanks for all your work and dedication!

Ich danke meiner Familie für die Unterstützung, Liebe und Aufmerksamkeit für meine Arbeit. Meine Eltern, Markus Naimer und Karina Kaffee, haben den Grundstein gelegt für alles, was ich geschafft habe. Es gibt keine Worte, um meine Dankbarkeit auszudrücken. Meine Geschwister, Amaru, Laura und Guillem – ihr seid mir immer ein Vorbild. Gràcies Conxa, per el teu carinyo. Meinen Großeltern danke ich für die Nachsicht, dass ich sie in der langen Zeit meiner Promotion kaum besuchen kam. Ich kann nicht versprechen, dass ich jetzt öfter da bin, aber ich denke immer an euch.

I'd like to thank the many people I met and was lucky enough to call friends in all the places I lived throughout these PhD years - Berlin, Southampton, Bristol, Hannover, and London. You made the frequent moving bearable and kept life exciting.
Johanna, Vanessa, Jabir, and Timo – Danke, dass ihr immer mein Zuhause seid.
Vinayak, നീയാണ് കൊടുങ്കാറ്റിൽ എന്റെ ശാന്തത ഡാ

Last but certainly not least, I would like to express my gratitude to the Wikidata and Wikipedia communities, who not only collaborated on my work but also supported my ideas and were eager to test them with me. This, among others, made the ArticlePlaceholder and Scribe projects possible. You make the world a better place, one edit at a time.

# Definitions

**Alias** An alias is an alternative name for a concept. While each entity should only have one label per language, it can have multiple aliases. Properties to indicate aliases include `skos:altLabel` and `schema:alternateName`.

**Class** Classes are a type of resource to sort entities into groups, i.e., to describe the type of an entity. The property used to declare an entity as a type of a class is `rdf:type`.

**Description** A description is a short text describing the content of an entity, helping users to differentiate between entities with similar labels.[1] Descriptions are in natural language, and can therefore be covered in different languages. Properties to indicate descriptions include `schema:description`.

**Entity** An entity is one data point in the web of data, describing a concept, expressed in a linked data format, and identified by a URI. In the following, we use *entity* and *concept* interchangeably.

**Knowledge graph** A knowledge graph is a database in a graph format, containing linked data in RDF format. More information: https://www.ontotext.com/knowledgehub/fundamentals/what-is-a-knowledge-graph/

**Label** A label is the natural language representation of a concept in a knowledge graph. A label should be annotated with a language in the form of `@<language code>`. For example, `"Berlin"@en` is the English label for the entity `wd:Q64`.

**Labelling property** A labelling property is a property indicating the connection between an entity and its label. The standard labelling property on the web of data is `rdfs:label`.

**Language tag** A language tag is used to mark the language of a string. Language tags are based on ISO codes for languages.[2] An example for a language tag could be `"<label>"@de`, where the language tag `@de` represents German, indicating the label is in German.

[1] https://www.wikidata.org/wiki/Help:Description, retrieved 11. May 2021

[2] https://www.w3.org/International/questions/qa-choosing-language-tags.en, retrieved 11. May 2021

**Linked data** Linked data is data that is machine readable, published in RDF or a comparable format, and part of the web of data.

**Linked open data (LOD)** Linked open data is freely accessible data on the web of data in RDF format, identified by URIs.

**Namespace** A namespace in RDF is the first part of the URI, often describing the knowledge graph an entity belongs to. For example, the namespace for entities in Wikidata is `http://www.wikidata.org/entity`. In Turtle notation, the namespaces are replaced by predefined prefixes.

**Ontology** Ontologies are a way of standardising the vocabulary across knowledge graphs.[3] For example, they can be used to indicate the type of an entity in all knowledge graphs with the standard `rdf:type`.

**Property** A property (or predicate) describes the relationship between two entities. In the triple, it is the link between two entities, i.e., *subject - property - object*.

**Resource Description Framework (RDF)** RDF is a format for describing concepts on the web of data in XML, designed to be machine readable. More information: `https://www.w3.org/RDF/`

**rdfs:label** `rdfs:label` is the standard labelling property in RDF to indicate a label in the form `<concept> rdfs:label ‘‘<label>’’@<language code>`. For example, the triple to describe the label of the entity `wd:Q64` is `wd:Q64 rdfs:label "Berlin"@en .`

**SPARQL** SPARQL is the query language of RDF. More information: `https://www.w3.org/TR/rdf-sparql-query/`

**Statement** A statement is a triple, or set of triples, expressing information about an entity. A statement might have additional information, such as provenance information (e.g., source of information) or other qualifiers (e.g., the time range a statement is valid in). Those qualifying triples can use blank nodes. More information on blank nodes: `https://www.w3.org/wiki/BlankNodes`

**Translation** Translation describes the task of transferring the meaning of a word from one language to another.

**Transliteration** Transliteration phonetically transfers a word from one script to another.

**Triple** Triples describe the relationship between two data points in a *subject - property - object* format, e.g., `wd:Q64 - wdt:P1367 - wd:Q183`, Berlin (`wd:Q64`) is the capital of (`wdt:P1367`) Germany (`wd:Q183`).

[3] According to the standard by `https://www.w3.org/standards/semanticweb/ontology`, retrieved 11. May 2021

**Turtle** Turtle is a compact version of RDF, making it easier for humans to read. Turtle defines prefixes for namespaces in the form `PREFIX wd: <http://www.wikidata.org/entity/>`, so that an entity in the form of `http://www.wikidata.org/entity/Q64` can be referred to as `wd:Q64`. We use turtle notation throughout this thesis.

**Uniform Resource Identifier (URI)** A URI is a unique identifier for an entity on the web of data. For example, the concept `Berlin`, the capital city of Germany, is described by the URI `http://www.wikidata.org/entity/Q64`. More information: `https://www.w3.org/Provider/Style/URI`

**Web of data** The web of data, or semantic web, defined by Berners-Lee et al. (2001) is an extension of the web that aims to make data on the web linked, interoperable, and machine-readable.

**World Wide Web Consortium (W3C)** The World Wide Web Consortium (W3C) defines standards for the web. More information: `https://www.w3.org/`

# Chapter 1

# Introduction

The web does not reflect the true diversity of language speakers using the internet. Currently, 60% of content online is in English.[1] However, only 17% of the world's population speak English.[2] This bias towards English online has material consequences. A growing number of non-English speakers are experiencing online knowledge inequality as they gain access to the internet. For example, from 2000 to 2020 the number of Arabic speakers using the internet increased by $9,348.0\%$, compared with an increase of 742.9% for English speakers in the same time frame.[3]

It is crucial to make the web more accessible to this increasing number of non-English speaking internet users and to provide information in their language (Peters et al., 2012). Supporting the multilingual reality of the web needs tools designed for the purpose.

To be accessible across languages, multilingual information needs to be machine readable and accessible in an interconnected format. The web of data extends the web with data in a machine readable format. It enables applications to access information on the web automatically, in ways that would not be possible in the traditional, human-readable web. W3C publishes the standards for making the web of data accessible across different vocabularies.[4] The web of data has been taken up by a number of data publishers, from cultural institutions to the British government's data portal (Shadbolt et al., 2012).[5]

Knowledge graphs can be used as a storage hub for multilingual information. In this thesis, we develop a framework to measure the coverage of languages across datasets

[1] https://w3techs.com/technologies/overview/content_language, retrieved 25.09.2020

[2] World population of $7,865,905,220$ according to https://www.worldometers.info/world-population/, retrieved 15. May 2021, 1348 million English speakers total according to https://www.ethnologue.com/guides/ethnologue200, retrieved 15. May 2021

[3] https://www.internetworldstats.com/stats7.htm, retrieved 25.09.2020

[4] https://www.w3.org/standards/semanticweb/data, retrieved 15.02.2021

[5] https://www.ontotext.com/knowledgehub/case%2Dstudies/linked%2Ddata%2Dintegration%2Dgalleries%2Dlibraries%2Darchives%2Dmuseums/, retrieved 15.02.2021

on the web of data. We demonstrate a lack of multilingual labels in both the web of data and Wikidata. In response, we propose a method of increasing the amount of data available across languages using translation and transliteration of knowledge graph labels. Further, we show two use cases of multilingual data on the web of data: ranking knowledge graphs based on their language information for question answering, and text generation for Wikipedia.

**Knowledge Graphs** Knowledge graphs are graph-structured data storage technology. Knowledge graphs link concepts, or *entities* (such as events, people, or places), with information about those concepts, e.g., `Berlin -- capital of -- Germany` and `Berlin -- population --` 3, 644, 826. Typically, knowledge graphs can focus on one specific domain, such as the linguistic information provided by BabelNet (Navigli and Ponzetto, 2010) or be domain independent, as in the case of Wikidata (Vrandecic and Krötzsch, 2014). Each concept or entity is identified by a Uniform Resource Identifier (URI), which describes an entity in the knowledge graph. In their functionality, URIs clearly differ from labels (Montiel-Ponsoda et al., 2011): while labels are a way for humans to interact with the data in natural language, URIs are supposed to be the unambiguous identifiers of entities, i.e., one URI refers to only one entity. Labels can be changed and exist in multiple languages. URIs, however, should be independent of the actual content of an entity, as they can not change.[6] For example, the entity for Berlin can be identified by `https://wikidata.org/wiki/Q64`.[7]

Information about an entity is expressed in *triples*. Triples describe the relationship between two data points in a *subject - property - object* format, e.g., `Q64 -- P1367 -- Q183`, Berlin (`Q64`) is the capital (`P1367`) of Germany (`Q183`).

**Labels** User-facing applications, such as question answering systems, make particular use of labels in the knowledge graph. A label is the human-readable description of an entity, such as the name of a person. Knowledge graphs are machine readable, i.e., they express information in a structured and linked format. Labels are the way humans can interact with this information. Human users depend on the availability of labels in their language, to make use of the data in a variety of applications such as question answering systems or natural language processing (NLP) tasks. Therefore, a comprehensive labelling of knowledge graphs across languages is needed in order for applications to be ported across languages.

Labels in a knowledge graph are represented as triples. Labels are indicated with the `rdfs:label` property. Figure 1.1 displays an example in graph structure from Wikidata. Wikidata's URIs are language independent. For example, Ada Lovelace is

[6] `http://www.w3.org/Provider/Style/URI`, retrieved 13. May 2021

[7] In the following, we omit the Wikidata namespace and address entities only by their ID in the Wikidata namespace, e.g., `Q64`.

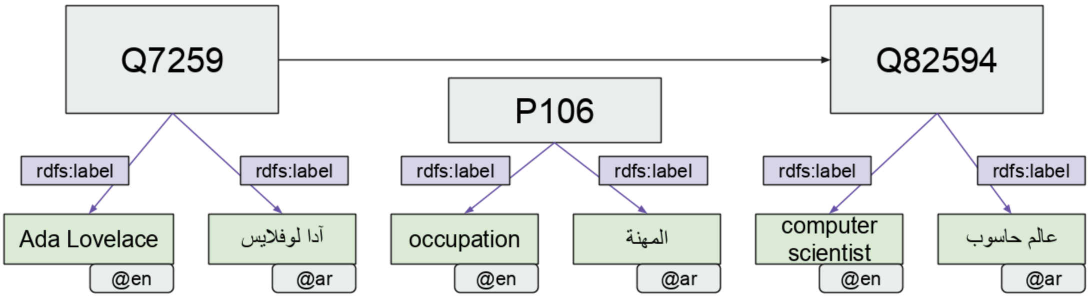
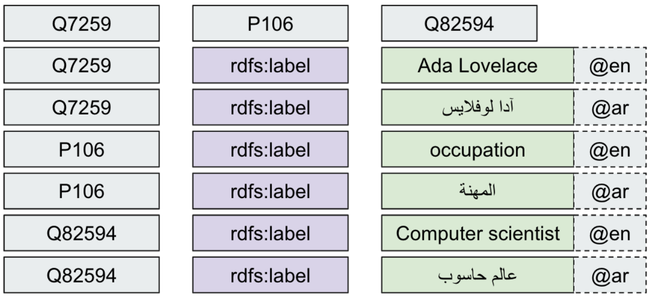


FIGURE 1.1: Example of a triple (*Ada Lovelace - occupation - computer scientist*) with the respective label for each entity and the property in English (en) and Arabic (ar).



| | | | |
|---|---|---|---|
| Q7259 | P106 | Q82594 | |
| Q7259 | rdfs:label | Ada Lovelace | @en |
| Q7259 | rdfs:label | آدا لوفلايس | @ar |
| P106 | rdfs:label | occupation | @en |
| P106 | rdfs:label | المهنة | @ar |
| Q82594 | rdfs:label | Computer scientist | @en |
| Q82594 | rdfs:label | عالم حاسوب | @ar |

FIGURE 1.2: Representation of Figure 1.1 in triples format. The `rdfs:label` property connects opaque URIs of the entity to their natural language representations in English and Arabic.

represented by `Q7259`. Figure 1.2 shows the representation as triples, where the triple to be expressed in natural language (`Q7259 - P106 - Q82594`) is shown in grey. I.e., with its English labels, the triple can be expressed as `Ada Lovelace - occupation - computer scientist`. Each of the entities in this triple are labelled in English and Arabic using the `rdfs:label` property. In addition to its label, each entity can have one or more *aliases* and *descriptions* in natural language. Aliases are alternative names for an entity. A label in RDF should be unique, i.e., one label per language.[8] This ensures that an application finds the preferred name for a concept easily. Aliases can then be used to indicate alternative names for the same entity. For example, an alias for Ada Lovelace is *Augusta Ada King, Countess of Lovelace* – a different way the entity's name can appear in text. A description is a longer, human-readable definition of an entity. For example, Ada Lovelace's English description on Wikidata is *English mathematician, considered the first computer programmer*.[9]

**Wikidata** Throughout this thesis, we focus on Wikidata (Vrandecic and Krötzsch, 2014), a knowledge graph which contains information about more than 93 million entities as of the writing of this thesis.[10] It is edited by an international community of

[8] `https://www.w3.org/2004/12/q/doc/rdf-labels.html`, retrieved 25. September 2020
[9] Ada Lovelace on Wikidata: `https://www.wikidata.org/wiki/Q7259`, retrieved 11. May 2021
[10] `https://www.wikidata.org/wiki/Wikidata:Main_Page`, retrieved 15 May 2021

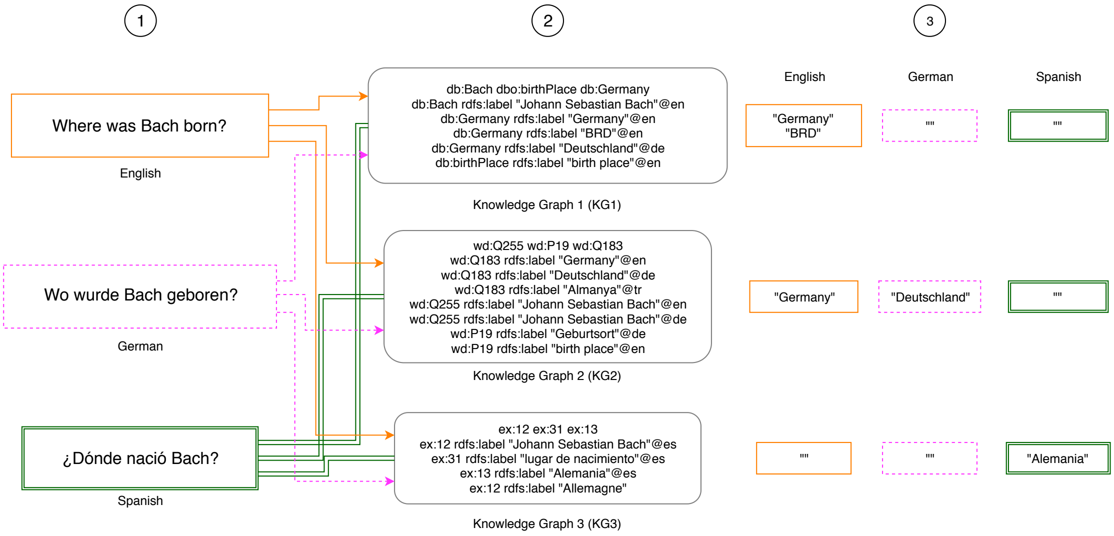

1
2
3
English
German
Spanish
Where was Bach born?
English
db:Bach dbo:birthPlace db:Germany
db:Bach rdfs:label "Johann Sebastian Bach"@en
db:Germany rdfs:label "Germany"@en
db:Germany rdfs:label "BRD"@en
db:Germany rdfs:label "Deutschland"@de
db:birthPlace rdfs:label "birth place"@en
Knowledge Graph 1 (KG1)
"Germany" "BRD"
""
""
Wo wurde Bach geboren?
German
wd:Q255 wd:P19 wd:Q183
wd:Q183 rdfs:label "Germany"@en
wd:Q183 rdfs:label "Deutschland"@de
wd:Q183 rdfs:label "Almanya"@tr
wd:Q255 rdfs:label "Johann Sebastian Bach"@en
wd:Q255 rdfs:label "Johann Sebastian Bach"@de
wd:P19 rdfs:label "Geburtsort"@de
wd:P19 rdfs:label "birth place"@en
Knowledge Graph 2 (KG2)
"Germany"
"Deutschland"
""
¿Dónde nació Bach?
Spanish
ex:12 ex:31 ex:13
ex:12 rdfs:label "Johann Sebastian Bach"@es
ex:31 rdfs:label "lugar de nacimiento"@es
ex:13 rdfs:label "Alemania"@es
ex:12 rdfs:label "Allemagne"
Knowledge Graph 3 (KG3)
""
""
"Alemania"

FIGURE 1.3: Motivating Example. (1) User queries, processed by a question answering system, in three different languages. (2) Three different knowledge graphs that are used by the QA systems independently to find an answer to the user question. (3) The collected answers. If there is a lack of data in the language to disambiguate and answer the question, the answer is empty (“”), if there is more than one label for the returned entity, the answer contains multiple, independent words. The second knowledge graph (KG2) is therefore most appropriate for English and German, the third knowledge graph (K3) is most appropriate for Spanish.

volunteers, with 25, 263 editors[11] working across 410 languages. Wikidata is multilingual by design. Each aspect of the data can be translated and rendered to the user in their preferred language. This makes it the tool of choice for a variety of content integration affordances in Wikipedia, including links to articles in other languages, and infoboxes. A range of other applications use Wikidata, such as question answering systems (Tanon et al., 2018) and chat bots (Athreya et al., 2018). Virtual assistants, used by a large number of people internationally, already integrate knowledge from Wikidata into their services.[12]

## 1.1 Motivation

As the motivation of this thesis, we present two use cases of multilingual knowledge graph data: question answering systems, and text generation for Wikipedia. Both use cases show the importance of multilingual knowledge graph data in their respective contexts.

[11] `https://www.wikidata.org/wiki/Special:Statistics`, retrieved 15 May 2021

[12] `https://www.wired.com/story/inside-the-alexa-friendly-world-of-wikidata/`, retrieved 15. May 2021

### 1.1.1 Question Answering

Amazon Alexa and Google Home have made question answering available to a large number of households over recent years. Amazon Alexa has sold over 100 million units[13]. Globally, products that answer user questions, from voice interaction systems to chatbots, are experiencing increased uptake

To support a larger number of users, Amazon has invested in its product's increasing ability to understand and answer questions in multiple languages (Gaspers et al., 2018).

Question answering systems draw on an existing set of background knowledge. Most commonly, question answering systems rely on linked data stored in knowledge graphs (Diefenbach et al., 2018).

We motivate our work with a use case in which a multilingual question answering system needs to choose between a set of knowledge graphs with different coverage of multilinguality. Consider three knowledge graphs that have different coverages of labels ((2), in Figure 1.3). The first knowledge graph (**KG1**) contains English and German labels, with more labels in English. One entity (db:Germany) has two labels in English. **KG2** contains English, German, and Turkish labels, with slightly more coverage of labels in German. **KG3** (unlike KG1 and KG2) contains only Spanish labels, and a label with no language tag. Consider a question answering (QA) system that answers questions in English, German, and Spanish. The QA system uses a knowledge graph as background knowledge to answer questions posed by a user.

In our example, the QA system wants to answer the same user question ((1), in Figure 1.3 in Chapter 1) about the birthplace of Bach, in Spanish, German, and English. The system needs to select among the three knowledge graphs the one that suits its needs best and can answer the question in three languages. Considering a multilingual QA system, all or most languages in the system need to be represented in the labelling of entities. KG1 contains labels in English, but they are ambiguous: the system cannot identify which label is the correct one. Further, there are no labels available in German or Spanish. Therefore, the second knowledge graph should be selected, as unique labels are available in English and German. Even though there are no Spanish labels, the system can answer the question in two of the three languages. The second choice is KG3, as it is able to answer a question unambiguously in one language, whereas KG1, the third choice, can also answer questions only in English, but even in English will return an answer with two labels. Systems have to select a background knowledge graph. If a system selects the wrong knowledge graph (e.g.,

[13] https://www.theverge.com/2019/1/4/18168565/amazon-alexa-devices-how-many-sold-number-100-million-dave-limp, retrieved 15 October 2020.

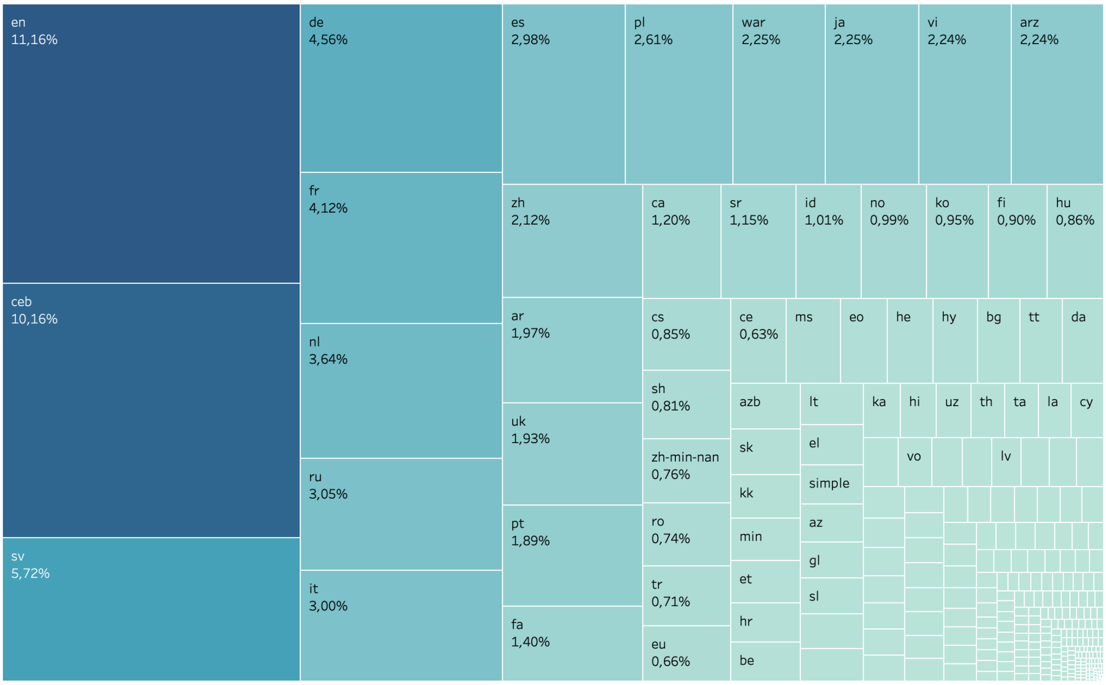


FIGURE 1.4: Wikipedia language versions and size in articles, based on https://en.wikipedia.org/wiki/List_of_Wikipedias, retrieved 28 April 2021

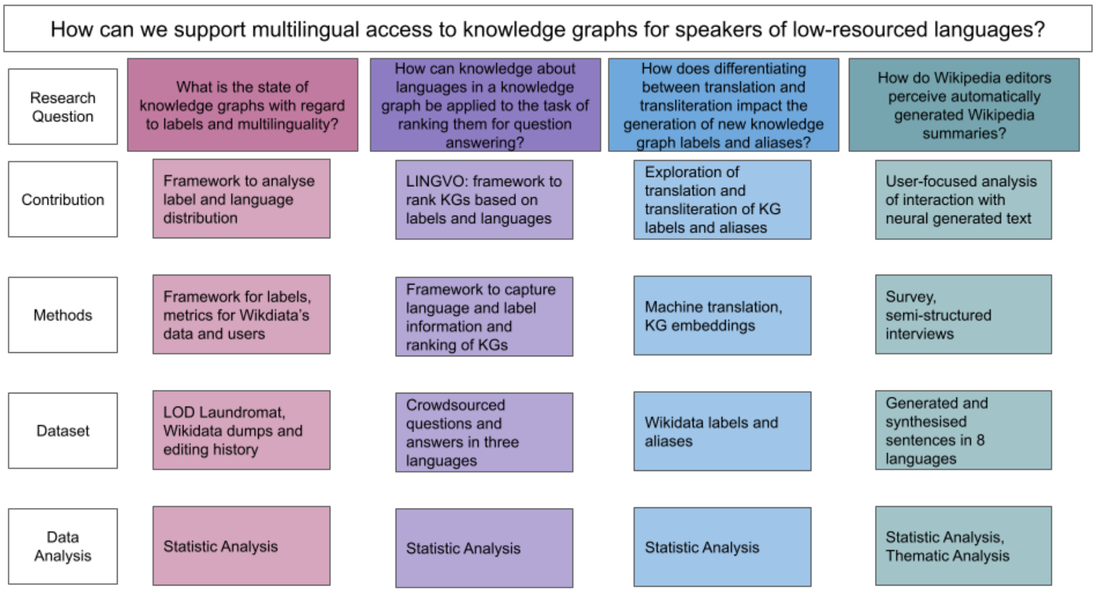
selecting KG1, making the system only able to answer the question in English, and ambiguously), the system will be rendered useless.

A ranking approach based on the language requirements of a question answering system is needed. Knowledge graphs can be ranked based on the range of topics that different systems require. Therefore, the first part of our work is to identify the labelling and multilinguality properties of different knowledge graphs, and represent them in a way that is easily accessible. We suggest a way of ranking knowledge graphs by their success in meeting end users' needs through the use of labels and multilinguality, based on the previously identified factors.

### 1.1.2 ArticlePlaceholder for Wikipedia

Wikipedia is one of the most accessed websites in the world.[14] The community-maintained encyclopedia is available in 303 languages.[15] However, its content is unevenly distributed (Hecht and Gergle, 2010), as seen in Figure 1.4. Language versions with less coverage than English Wikipedia face a recursive problem: fewer editors means less quality control, making that particular Wikipedia less useful for readers in that language; which in turn makes it more difficult to recruit new editors from among readers.

[14] Alexa rank 13, https://www.alexa.com/siteinfo/wikipedia.org, retrieved 13 October 2020.
[15] https://en.wikipedia.org/wiki/List_of_Wikipedias, retrieved 28 April 2021



How can we support multilingual access to knowledge graphs for speakers of low-resourced languages?

| | | | | |
|---|---|---|---|---|
| Research Question | What is the state of knowledge graphs with regard to labels and multilinguality? | How can knowledge about languages in a knowledge graph be applied to the task of ranking them for question answering? | How does differentiating between translation and transliteration impact the generation of new knowledge graph labels and aliases? | How do Wikipedia editors perceive automatically generated Wikipedia summaries? |
| Contribution | Framework to analyse label and language distribution | LINGVO: framework to rank KGs based on labels and languages | Exploration of translation and transliteration of KG labels and aliases | User-focused analysis of interaction with neural generated text |
| Methods | Framework for labels, metrics for Wikdiata's data and users | Framework to capture language and label information and ranking of KGs | Machine translation, KG embeddings | Survey, semi-structured interviews |
| Dataset | LOD Laundromat, Wikidata dumps and editing history | Crowdsourced questions and answers in three languages | Wikidata labels and aliases | Generated and synthesised sentences in 8 languages |
| Data Analysis | Statistic Analysis | Statistic Analysis | Statistic Analysis | Statistic Analysis, Thematic Analysis |

FIGURE 1.5: Research questions posed in this thesis, contributions, methods, and datasets for each research question.

Multiple approaches aim to address this problem, such as the content translation tool on Wikipedia, which enables a user to create a new article in their language based on (machine) translation from the same article in another Wikipedia language version.[16] Not all information on non-English Wikipedia sites is available in English Wikipedia. In fact, Hecht and Gergle (2010); Hecht (2013) have disproved the English-as-superset hypothesis, showing that different Wikipedia language versions contain different content.

In Chapter 8, we suggest using the available information in a knowledge graph, Wikidata, to automatically generate Wikipedia articles – the ArticlePlaceholder. We show that it is possible to use knowledge graph information to create placeholder pages in lower-resourced languages on Wikipedia such as Arabic and Esperanto. These placehold pages have automatically generated introduction sentences that are useful to readers and editors alike. This shows the importance of storing data centrally and multilingually, because this enables us to make information available across languages.

## 1.2 Research Questions

In the following, we present the research questions we aim to answer in this work (Figure 1.5). We aim to contribute to a better understanding of the role of knowledge

[16] `https://www.mediawiki.org/wiki/Content_translation`, retrieved 28 April 2021. Some Wikipedia language versions, such as English Wikipedia, disable machine translation in the content translation tool (see `https://en.wikipedia.org/wiki/Wikipedia:Content_translation_tool`, retrieved 28 April 2021).

graphs when supporting access to multilingual content online. Therefore, the primary question we aim to answer is: **How can we support multilingual access to knowledge graphs for speakers of low-resourced languages?** We explore this research question from four different perspectives. First, we establish the state of knowledge graphs with regards to languages and labels, and develop a framework that we validate by applying it on two different data sources and reuse it for the task of ranking knowledge graphs. We further propose an approach for translation and transliteration of knowledge graph labels based on work in the field of machine translation and transliteration. Finally, to make knowledge graph labels accessible across languages as text, we propose an evaluation methodology for neural text generation from knowledge graph data with a focus on the users.

### RQ1: What is the state of knowledge graphs with regard to labels and multilinguality?

**Motivation** Knowledge graphs have the structure to support multilingual applications, but in order to effectively support applications across languages, knowledge graphs must also have multilingual content in the form of labels. Therefore, we investigate the state of knowledge graphs with regard to labels and languages. To deepen our understanding of multilinguality in knowledge graphs, we also explore the provenance of the data in the graphs. In the case of Wikidata, a community contributes the content. By understanding how the existing language distribution came about, we gain an insight into how the community created the multilingual content. This can contribute to an understanding as to how and where to support communities of editors to create more diverse knowledge graphs. In order to have a basis for comparison of different datasets, and in an effort to unify different previously proposed frameworks and guidelines, we develop a comprehensive framework for measuring labelling and multilinguality.

**Data and Methods** We introduce a framework for the analysis of knowledge graphs with a focus on labelling of entities and languages covered in a dataset. We apply this framework to understand the label and language coverage of the web of data and Wikidata, by conducting two descriptive studies in which we present an analysis of the LOD laundromat and the Wikidata dataset. The LOD laundromat aggregates a large number of datasets on the web of data and unifies them. We therefore use it as a representation of the web of data. Based on the edit history of Wikidata, we deepen our understanding of the provenance of Wikidata's multilingual data and its users' multilinguality. Data and code can be found here:
`https://github.com/luciekaffee/metrics-label`,
`https://github.com/luciekaffee/Wikidata-User-Languages`

**Results** We develop a framework that can be applied to any dataset to measure labelling and multilinguality (Chapter 3) and apply it to two datasets: LOD Laundromat (Chapter 4) and Wikidata (Chapter 5). We show how this framework can be reused in applications (see RQ3). We find that the web of data still lacks labels for entities, and wider support for a large number of languages. While Wikidata generally has a more diverse coverage of languages, it is still largely biased towards English. However, the multilinguality of editors and their contributions give a promising direction for diversification of the knowledge graph in terms of labels.

### RQ2: How can knowledge about languages in a knowledge graph be applied to the task of ranking them for question answering?

**Motivation** Users pose questions in different languages to a question answering system. Given the large amount of knowledge graphs on the web of data, a question answering system can have access to multiple knowledge graphs for answering user questions. This poses the challenge of selecting the correct knowledge graph for a given question and user language.

**Data and Methods** We propose a method of ranking knowledge graphs for effectiveness in question answering by capturing knowledge about language and label coverage based on the framework introduced in Chapter 3. We investigate whether extracting label and language information can help select the knowledge graph containing the most appropriate answer for a user's question in their language. We introduce LINGVO, a framework able to compare and rank knowledge graphs based on multilingual knowledge at class level. The approach is tested on an extended version of the QALD dataset, including five widely used knowledge graphs (Wikidata, DBpedia, YAGO, MusicBrainz and LinkedMDB) in our analysis. To select the best knowledge graph to answer a given question, we create a gold standard based on the answers of human annotators in a crowdsourcing experiment. The data and code for the experiments can be found here: `https://github.com/luciekaffee/LINGVO`.

**Results** We empirically show that ranking at class level leads to precise results. Moreover, the ranking of these knowledge graphs is particularly effective when performed in a contextual domain, e.g., movies or people.

### RQ3: How does differentiating between translation and transliteration impact the generation of new knowledge graph labels and aliases?

**Motivation** Knowledge graphs lack multilingual coverage, which limits their applicability in many tasks. Wikidata's approach to correcting this limitation is to collate labels from its multilingual community. However, this is highly time- and cost-intensive, and cannot scale across the entire knowledge graph. Automation is needed in this aspect. We propose a method of automatically translating labels.

**Data and Methods** Our approach explores the generation of a Chinese alias given an English label in the company domain. The research challenges are as follows: (1) Company labels can be translated or transliterated without an indicator of translation or transliteration, creating the need for an automatic classification. (2) Since datasets are a mix of simplified and traditional Chinese, we explore the impact of conversion from one character set to the other on the overall translation. (3) We explore the impact of differentiating translation from transliteration, compared to a pure translation or transliteration model.

We tackle these research challenges by investigating each of the stages of a pipeline that takes an English label as an input and generates a Chinese alias, by differentiating between translation and transliteration. The generation of the alias is based on the neural models widely used for neural machine translation, transformers. We train the model on out-of-domain datasets and test them on the company domain in Wikidata.

**Results** Differentiating effectively between translation and transliteration before generating an alias improves results by 34.7% compared to using a translation- or transliteration-based model on its own for the transliteration model, and improves results by 25.4% for the translation model, in terms of character error rate. Converting all Chinese training and test datasets can improve performance by 77.12% in terms of BLEU-4 score.

### RQ4: How do Wikipedia editors perceive automatically generated Wikipedia summaries?

**Motivation** Knowledge graph information in the form of triples is accessible and readable by machines, but triples are not easy to read for human users. The field of data-to-text generation tackles this problem by generating natural language text from knowledge graph triples. In our work, we approach the problem of Wikipedia summaries, which give an overview of a topic for Wikipedia readers and editors. Our focus is on the target user group, i.e., the readers of the generated sentences, who need to understand them and potentially reuse them to create articles on Wikipedia.

**Data and Methods** We work on lower-resourced languages, i.e., languages with fewer resources available online.[17] Specifically, we work with Arabic and Esperanto. Using an encoder-decoder architecture to generate Wikipedia summaries from Wikidata triples in Arabic and Esperanto, we propose a methodology to evaluate these generated sentences with the community. We take a mixed-methods approach that includes surveys and semi-structured interviews with the communities we address. The data and code for the experiments can be found here: https://github.com/pvougiou/Mind-the-Language-Gap.

**Results** We show that natural language generation is a promising direction to support lower-resourced Wikipedias; however, it brings its own challenges, too.

## 1.3 Context

**Knowledge Graphs** Throughout this thesis, we work with knowledge graphs as the *ideal* technology for representing knowledge. There are some obvious limitations to using knowledge graphs in that way. From a social science perspective, knowledge graphs are eurocentric, they are largely developed by European researchers, following (mostly) European research methodologies to capture eurocentric knowledge. For more information regarding the topic of how the methodology of research my inflict its outcomes see: Smith (2021). The author also gives an outlook on how researchers from an underrepresented group can own research; this is a bright perspective also for the bias here implied in knowledge graphs. A large part of the world's knowledge cannot be represented in the structure of knowledge graphs, for example oral knowledge, which was discussed in the context of Wikipedia and is yet to be explored in the context of knowledge graphs.[18] The inherit structure of knowledge graphs in the form of *subject – predicate – object* represents only a subset of languages, and excludes a wide range of languages. To the best of our knowledge these limitations are yet to be addressed and are outside of the scope of our studies.

**Low-resourced languages** This thesis works on what is here called *low-resourced*, *lower-resourced*, or *under-resourced* languages. This is not a statement of how widely the

[17] We introduce the concept of *lower-resourced languages* analogously to *low-resourced languages*, a term common in natural language processing research. Lower-resourced languages are languages that are not among the 10 best covered languages, but still have more information online than, e.g., minority languages. See Section 1.3.

[18] https://medium.com/@lucie.kaffee/the-sum-of-all-knowledge-oral-citations-on-wikipedia-abaad65c5b0c

language is spoken. We borrow the term from natural language processing (NLP) research, in which any language lacking language resources (i.e., enough language information to automatically process the language) is defined as low-resourced. Singh (2008) point out the following problems defining low-resourced languages:

- **Linguistic study**: English is so widely studied that even widely spoken languages such as Hindi are not comparable
- **Language Resources**: Lack of (machine-readable) resources in the language
- **Computerization**: Lack of existing (NLP) tools in the language
- **Language Processing**: Lack of automated processing of the languages
- **Other Privileges**: The main factor identified by the author is the lack of privilege of many low-resource languages, namely lack of ”availability of finance, equipment, human resources, and even political and social support for reducing the lack of computing and language processing support“ (Singh, 2008)

Cieri et al. (2016) aim to define low-resourced languages overall and define *critical* languages as ”languages that suffer an undesirable ratio of supply to demand“ (Cieri et al., 2016), including among others Arabic and Chinese. Critical languages are a subset of low-resourced languages. The authors point out that projects working on low-resourced languages can include European languages, given they are not as well-covered as English in terms of resources. They further describe how low-resource can be defined in comparison to other languages, which is the approach we follow in our work. In other words, we create a context for the language we work with and define the language in terms of coverage in the context we work in.

Dependent on context, we define low-resourced as compared to the number of native speakers in the world (e.g., see Chapter 3) or the number of Wikipedia articles in the language compared to the larger, better-covered Wikipedias in terms of number of articles (e.g., see Chapter 8). While often low-resourced languages are associated to being less-privileged (Singh, 2008), the languages we work with in this thesis are widely spoken, and have their own large communities.

In Chapter 7, we focus on (simplified) Chinese, a character set used in Mainland China, Malaysia and Singapore for a language spoken by 1.31 billion speakers across the world.[19] In Chapter 8, we mainly work with Arabic, a language spoken by 319 million people.[20] Both languages are among the most spoken languages in the world.

[19] https://www.statista.com/chart/12868/the-worlds-most-spoken-languages/, retrieved 12. October 2021.

[20] https://www.statista.com/chart/12868/the-worlds-most-spoken-languages/, retrieved 12. October 2021.

We define low-resourced here also in relation to the number of speakers of the languages and their limitations to access to information online – only 1.2% of content online is in Arabic, 1.3% in Chinese.[21]

**Lack of Multilingual Labels** We speak in this thesis about lack of multilingual labels. This, as much as the definition of low-resource language, is context dependent. That is, when we speak about the lack of multilingual labels, we usually speak about it in the context of language distribution we can observe in the world. For example, in Chapter 3, we speak about the distribution of languages, and define lack of multilingual labels based on the bias towards one language (English). In Chapter 6, we define an ideal knowledge graph, which contains labels in all possible languages. While the current state of knowledge graph is still far from this ideal (see Chapters 4 and 5), we believe it is important to aim at an idealistic language distribution, which enables speakers of all languages to access information in their native languages.

## 1.4 Structure of the Thesis

This thesis is structured along the research questions posed in Section 1.2. First, we contextualise the work in Chapter 2. We address the first research question (RQ1) by introducing a framework to measure labelling and multilinguality in Chapter 3, and then applying it to a representation of the web of data in form of the LOD laundromat dataset in Chapter 4. We apply the same framework to Wikidata in Chapter 5, gaining further insight into the state of knowledge graphs with regard to labels and multilinguality.

Showing a use case for the framework introduced in Chapter 3, and to answer the second research question (RQ2), in Chapter 6 we propose an approach to ranking knowledge graphs based on information about each graph's range of languages and use of labels for question answering. Finding a lack of language coverage in the studies in Chapter 4 and Chapter 5, we investigate the automatic generation of Chinese aliases from English labels in Chapter 7. We show another use case for multilingual labels in Chapter 8. We test the utility for the community of using ArticlePlaceholder to generate placeholder pages using Wikidata triples and natural language generation from triples for Wikipedia. Finally, in Chapter 9 we discuss future research directions suggested by the findings in this thesis.

[21] https://w3techs.com/technologies/overview/content_language, retrieved 18. October 2021.

## 1.5 Previous Publications by the Author

This thesis is based on and expands previous publications by the author.

Chapters 3 and 4 are based on and extend the publication Kaffee and Simperl (2018a), under the supervision of Elena Simperl.

Chapter 5 is based on the publications Kaffee et al. (2017, 2019a), which explore multilinguality in the labels of Wikidata and its editors. The work was conducted in collaboration with Pavlos Vougiouklis, Alessandro Piscopo, and Kemele M. Endris, under the supervision of Elena Simperl. The author of this thesis has contributed the majority of the data collection, data analysis, and evaluation.

The experiments in Chapter 7 were conducted as part of the author's internship at Bloomberg L.P., London, under the supervision of Oana Tifrea-Marcius and Edgar Meij, both of Bloomberg L.P. The content of this chapter has been submitted to a research venue and is currently under review.

Chapter 6 is based on the publication Kaffee et al. (2019b), which works on the ranking of knowledge graphs based on their distribution of languages and use of labels. The work was conducted in collaboration with Kemele M. Endris, under supervision of Maria-Ester Vidal and Elena Simperl. The author of this thesis has worked on the data collection, crowdsourcing experiments, analysis, and evaluation.

Chapter 8 is based on the publications Kaffee et al. (2018b,a, 2021), which propose the generation of Wikipedia summaries from Wikidata triples with a focus on human evaluation. The work was conducted in collaboration with Hady Elsahar and Pavlos Vougiouklis, under supervision of Christophe Gravier, Frédérique Laforest, Jonathon Hare, and Elena Simperl. The author of this thesis has worked on the multilingual dataset creation, the result analysis, and the community interaction, and conducted the interviews.

# Chapter 2

# Background

In this chapter, we contextualise the studies of the following chapters. To frame the approaches proposed in this thesis, we explore the existing work in the fields of measuring multilinguality in knowledge graphs, machine translation over labels, multilingual question answering, and natural language generation from knowledge graph triples (data-to-text generation).

## 2.1 Web of Data

The *semantic web* or *web of data*, introduced by Berners-Lee et al. (2001), describes the part of the web that is expressed in linked data and is therefore machine-readable (Bizer et al., 2009). The data can be interlinked across different data sources, facilitating data processing for machines of the content and context of data.

Each concept, or entity, is described by a unique identifier (URI), which sets it apart from any other concept with the same name in natural language, e.g., the company `Apple` and the fruit `apple`. Entities can be identified by a hash URI or a URI that can be resolved in a 302 or 303 response in the HTML header. A URI, according to Montiel-Ponsoda et al. (2011), should not contain natural language. The authors encourage the usage of opaque URIs, that is, language-independent identifiers[1]. Entities describe things, such as cities or people; or classes of things, such as the set of all cities; as well as their properties, such as the number of people in a city, or the quality of a city being the capital of a country.

Each entity can then be linked to data, creating *statements* about this entity. Those statements can be displayed in the Resource Description Framework (RDF) (Lassila and Swick (1999)). RDF is a language for representing information on the web. With

[1] This also follows the recommendations of `http://www.w3.org/Provider/Style/URI`, retrieved 12. May 2021

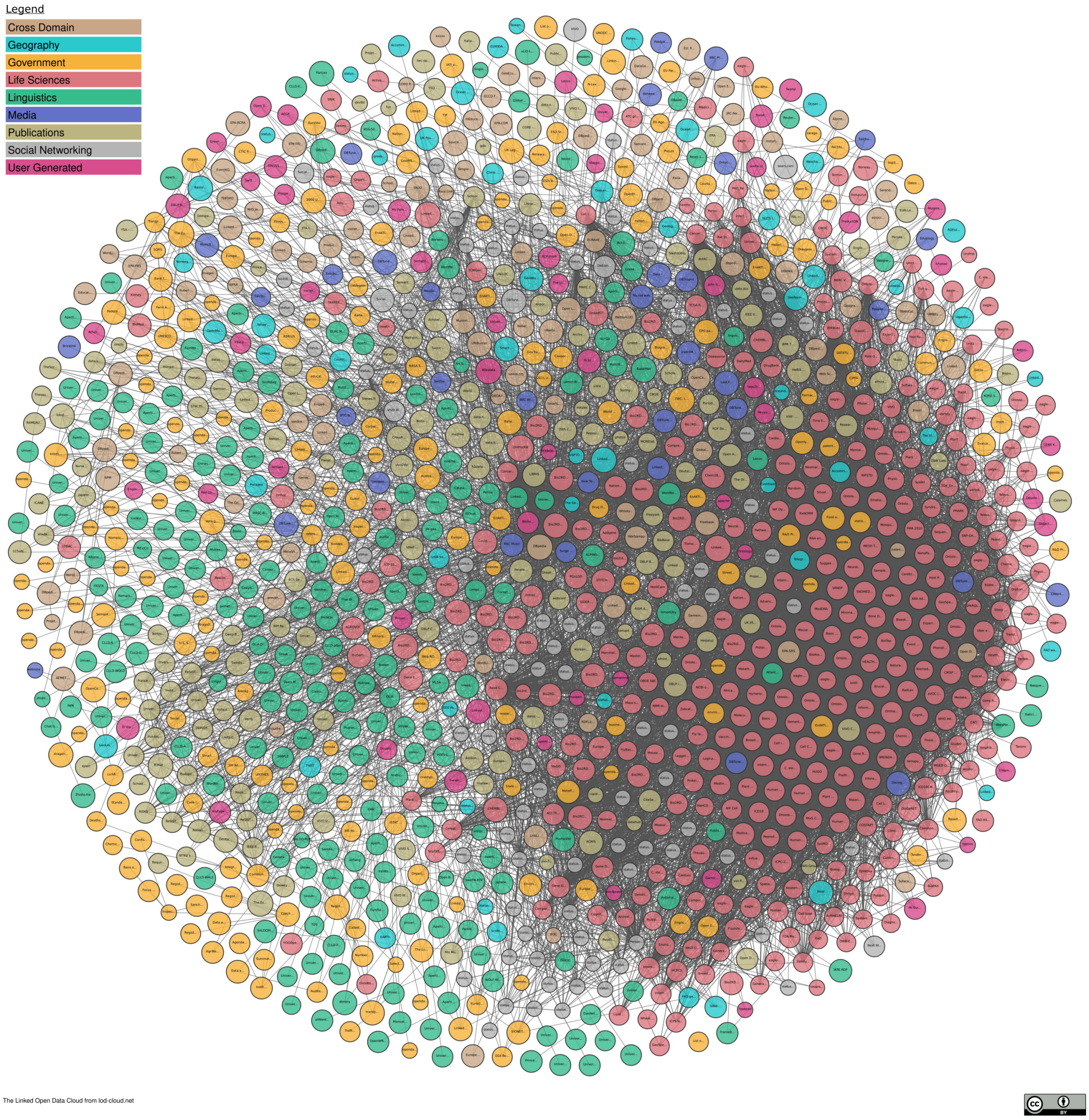


FIGURE 2.1: Linked Open Data cloud visualising the interconnections between different knowledge graphs. CC-BY `https://lod-cloud.net/`, retrieved 9. October 2020.

RDF, linked data can be modeled in an XML syntax. For example, about the entity `Berlin`, we can make the statement that it has a population of 4 million. In RDF, this triple could look like this: `Berlin -- population -- 4,000,000`, where Berlin is the subject, and population is the property and predicate of the statement. The numerical value is the object of the statement.

One of the goals of the web of data is to have different datahubs that a user can interact with easily as one. Therefore, a set of ontologies was introduced to serve as the standard of modeling. Examples of such ontologies in widespreade use are RDFS and OWL, as described by Allemang and Hendler (2011). These ontologies define a set of classes and properties that can be reused, and thus make interlinking of different datasets easier.

Knowledge graphs are a way of storing data in RDF. Knowledge graphs can be interlinked, so that one domain-specific knowledge graph, such as one containing health care data, is linked to a general domain knowledge graph that gives the health care data context. The linked open data cloud[2] in Figure 2.1 visualises these interlinks by representing knowledge graphs and their connections to each other in the web of data, .

### 2.1.1 Datasets

The web of data consists of a large number of datasets. Various attempts have been made to create a unifying dataset that would contain and standardise all other datasets in the web of data. We give here a brief overview over domain-dependent and -independent datasets, as well as datasets integrating the different datasets on the web of data.

Examples of domain-independent datasets include the knowledge graphs Wikidata (see Section 2.4) and DBpedia (Lehmann et al., 2015). These knowledge graphs aim to model the knowledge of different concept of the world, irrespective of domain, in a machine-readable format, and are interconnected. Wikidata, for example, uses so-called *external identifiers* to link to other data sources containing the same concept.[3] External identifiers facilitate the interoperability of different web sources. DBpedia is, similar to YAGO (Mahdisoltani et al., 2015), a knowledge graph extracted from information on Wikipedia. The aim of the project is to make the information collected on Wikipedia machine readable and therefore accessible for a large range of applications.

Another set of knowledge graphs are described by Abu-Salih (2021) in their survey of domain-specific knowledge graphs. These knowledge graphs focus on one domain, such as life sciences, diseases, or telecommunication.

Another direction of work in the web of data is to make use of the interoperability of these datasets and present them in a homogeneous, reusable way. One of the most significant efforts in this direction is the LOD Laundromat dataset (Beek et al., 2014). The laundromat unifies the different formats of datasets on the web of data. More information about the dataset can be found in Chapter 4, where we treat it as a representation of the web of data. Different data integration efforts are based on the approach of the LOD Laundromat, such as LOD-a-lot (Fernández et al., 2017).

[2] The linked open data cloud can be found at `http://lod-cloud.net`

[3] `https://www.wikidata.org/wiki/Wikidata:External_identifiers`, retrieved 12. May 2021

## 2.2 Knowledge Graphs

Knowledge graphs store linked data in a graph format. We use the term knowledge graph to refer to any graph-based representation of knowledge, as it is convention in the semantic web community – see, e.g., Kejriwal et al. (2019).

Paulheim (2017) consider a knowledge graph to have the following characteristics:

> ”(1) mainly describes real world entities and their interrelations, organized in a graph, (2) defines possible classes and relations of entities in a schema, (3) allows for potentially interrelating arbitrary entities with each other and (4) covers various topical domains”.

Färber et al. (2018) define a knowledge graph as a graph containing RDF triples. As Ehrlinger and Wöß (2016) detail in their attempt to define what a knowledge graph is, knowledge graph and knowledge base are typically used interchangeably, which we will do as well.

The most widely used knowledge graphs differ in the methods that were used to create them: (1) automatically constructed knowledge graphs, (2) community-maintained knowledge graphs, and (3) expert-maintained knowledge graphs. Automatically constructed knowledge graphs process data in other data formats and bring it into linked data format. Most prominently, DBpedia (Auer et al., 2007) and YAGO (Mahdisoltani et al., 2015) extract information from Wikipedia to represent it in knowledge graphs. Community-maintained knowledge graphs such as Wikidata (Vrandecic and Krötzsch, 2014) gather data by relying on a community of human editors to import data, i.e., they are publicly editable. Finally, expert-maintained knowledge graphs such as OpenCyc, the open-license version of Cyc (Lenat and Guha, 1993), rely on experts to create and maintain the data. They are not available to be edited by the public.

## 2.3 Labels and Languages in Knowledge Graphs

Labels in the web of data are fundamental for making data accessible. To allow people to engage with linked data effectively, whether as part of an end-user application such as a question answering system, or in a technical context, such as editing a knowledge graph, entities must have human-readable representations.

Most of the standard ontologies support labelling properties, with `rdfs:label` being the most used one according to Ell et al. (2011). In a knowledge graph such as Wikidata that uses opaque URIs, an entity could be described as detailed in Figure 2.3.

For example, the entity *Ada Lovelace* is labelled in a triple as following: `wd:Q7259 -- rdfs:label -- "Ada Lovelace"@en`. The language tag `@en` marks the language the string is in. The language tag is a standard way to mark the language of strings in RDF (Consortium et al., 2014).

One crucial factor that will shape the future of the multilingual web is the setting of standardised guidelines for representing multilingual resources on the semantic web. Multilingual structured data has been investigated by Gracia et al. (2012); Gómez-Pérez et al. (2013). However, none of these take into account the fact that a knowledge graph can be built by a community. Kaffee et al. (2019a) show that the engineering of knowledge by a community of users rather than automatic extraction, can have a fundamental impact on the coverage and development of a knowledge graph.

### 2.3.1 Guidelines

Clear guidelines can help data publishers to consider a multilingual layout in the development of their knowledge graph, as suggested by Gómez-Pérez et al. (2013). The authors give an insight into the present distribution of multilingual data on the web, and suggest a framework for working toward a future in which more online data is more multilingual. Ell et al. (2011) developed a framework for analysing label coverage of linked data resources, which sets a baseline for investigating the human readability of the web of data. They conclude with recommendations for growing multilingual knowledge graphs. Zaveri et al. (2016) survey methods of evaluating data quality. They describe metrics for *human readable labelling* in the understandability dimension, which describes the coverage of entities by labels. Debattista et al. (2016) set out a method for assessing linked data quality based on Zaveri's surveyed metrics, including human-readable labelling. Across the literature, the guidelines for multilingual knowledge graphs can be summarised as follows:

**Opaque URIs** When creating a knowledge graph, one should consider the implications of meaningful URIs, i.e., URIs with natural language vs opaque URIs. Natural language URIs, such as `http://dbpedia.org/page/Berlin`, can carry meaning and are harder to adapt in comparison to opaque URIs. Opaque URIs are not readable to humans and do not carry meaning or language-specific information, such as `https://www.wikidata.org/wiki/Q64`.

**Coverage** All entities in the knowledge graph should be labelled.

**Language tags** Language tags, such as `@en` for English labels, should be used to indicate the language of a label.

**Language coverage** All labels should have language tags, and ideally all labels should be labelled across all languages.

**Reusing existing vocabulary** When creating a knowledge graph, it is recommended to reuse the existing ontologies wherever possible for easier integration across multiple knowledge graphs.

**Unambiguity** To avoid confusion, only `rdfs:label` should be used to label the entities, and only one preferred label should exist per entity.

### 2.3.2 Use Cases

The use cases for multilingual data are diverse: for humans to understand the information, natural language multilingual data is necessary. Consequently, there is a strong relationship between the semantic web and natural language processing (Ehrmann et al., 2014). A well-established and comprehensive knowledge graph that includes labels in a large number of languages might serve as a base for applications interacting with humans via natural language. Gracia et al. (2012) suggest that, while the semantic web can be a source of multilinguality, there is still a lack of the services necessary to support a fully multilingual web. As they suggest in their work, most content is still mainly monolingual. Another possible use of multilingual data is question answering over linked data, as investigated by Aggarwal et al. (2013); Höffner et al. (2016). Pazienza et al. (2005) consider how multilingual ontologies can facilitate question answering. Similarly, Chaves and Trojahn (2010); Montiel-Ponsoda et al. (2011) look at how to enable multilinguality for ontologies.

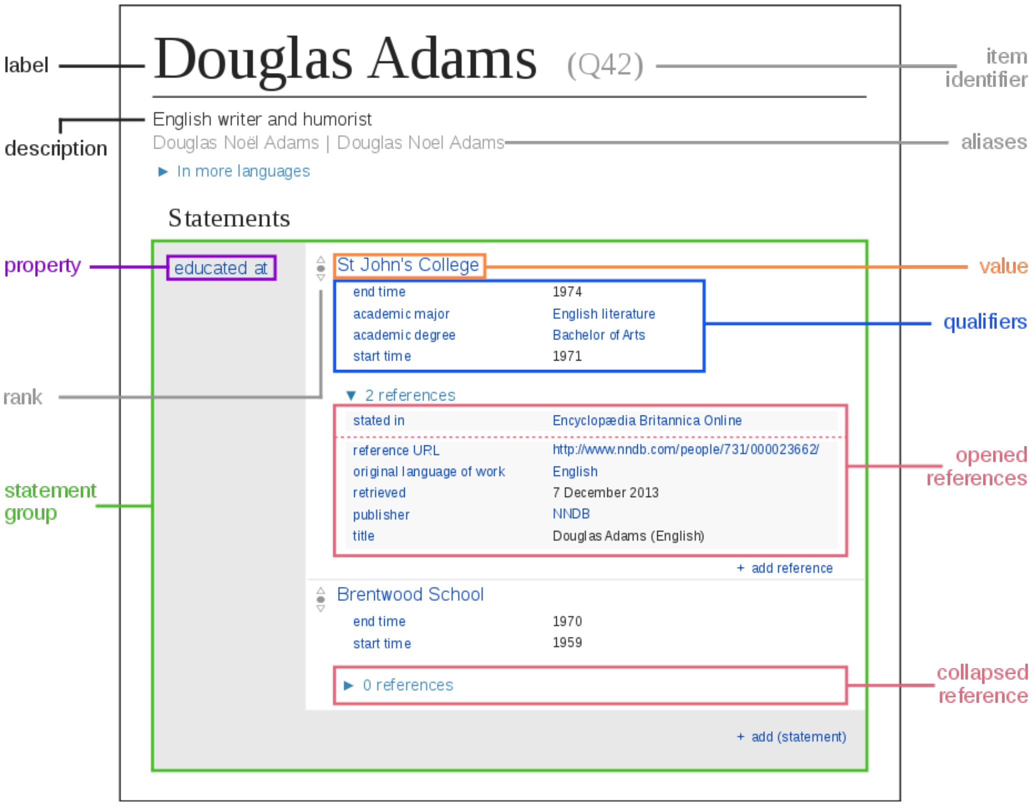
We describe the context for our use cases - multilingual question answering over knowledge graph data, and article generation for Wikipedia from Wikidata knowledge - in Section 2.6.

Previous studies demonstrated a lack of multilingual information in knowledge graphs, e.g., Kaffee and Simperl (2018a) show the lack of multilingual information across five knowledge graphs. We extend this study in Chapter 4 to cover a representation of the web of data. In Section 2.5 we discuss the use of machine transliteration and translation to increase the language coverage in knowledge graphs.

## 2.4 Wikidata

Wikidata[4] is a knowledge graph that is edited and maintained by a community of users. It is published under CC0 (public domain), making it reusable to anyone.

[4] Wikidata's main page at `https://www.wikidata.org/wiki/Wikidata:Main_Page`, retrieved 12. May 2021



Douglas Adams (Q42)

English writer and humorist

Douglas Noël Adams | Douglas Noel Adams

► In more languages

Statements

educated at

St John's College

| | |
|---|---|
| end time | 1974 |
| academic major | English literature |
| academic degree | Bachelor of Arts |
| start time | 1971 |

▼ 2 references

| | |
|---|---|
| stated in | Encyclopædia Britannica Online |
| reference URL | http://www.nndb.com/people/731/000023662/ |
| original language of work | English |
| retrieved | 7 December 2013 |
| publisher | NNDB |
| title | Douglas Adams (English) |

+ add reference

Brentwood School

| | |
|---|---|
| end time | 1970 |
| start time | 1959 |

► 0 references

+ add (statement)

FIGURE 2.2: Wikidata's data model. CC-0 Charlie Kritschmar, https://commons.wikimedia.org/wiki/File:Datamodel_in_Wikidata.svg

Wikidata (Vrandecic and Krötzsch, 2014) was originally created to support Wikipedia's language links, i.e., the connections between any given article across different Wikipedia language versions. Wikidata has been widely adopted outside Wikimedia projects, too. Two literature reviews around Wikidata show the breadth of Wikidata use in research. Farda-Sarbas and Müller-Birn (2019) give an overview of the research around Wikidata; Piscopo and Simperl (2019) focus on papers investigating the data quality of Wikidata.

Wikipedia and Wikidata are still closely linked. One use case of Wikidata in Wikipedia is the generation of Wikipedia infoboxes from Wikidata information. Sáez and Hogan (2018) propose an approach to generating inforboxes fully automatically, while the *Wikidata Bridge*[5] focuses on editing Wikidata from Wikipedia through infoboxes (Kritschmar, 2016).

Wikidata contains statements on general knowledge, e.g., about people, places, events, and other entities of interest. The data model expresses statements about those entities or items in triple form. Like Wikipedia, Wikidata understands itself as a tertiary source[6], collecting information from different primary and secondary sources and

[5] https://www.mediawiki.org/wiki/Wikidata_Bridge, retrieved 13. January 2021

[6] https://en.wikipedia.org/wiki/Wikipedia:Wikipedia_is_a_tertiary_source, retrieved 22. June 2021

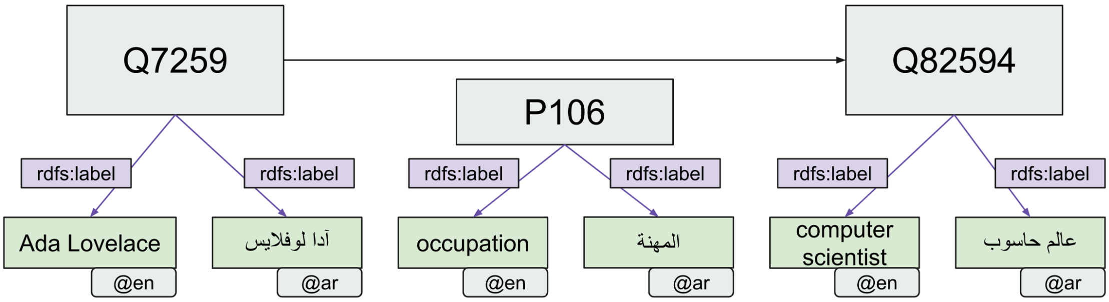


FIGURE 2.3: Example of labelling in Wikidata. Each entity can be labelled in multiple languages using the labelling property `rdfs:label`.

summarising them (Piscopo et al., 2017a,c). Therefore, each statement in the knowledge graph should contain a reference. These are expressed using RDF blank nodes.

The knowledge graph is created and maintained collaboratively by a community of editors, assisted by automated tools called bots (Steiner, 2014). Bots take on repetitive tasks such as ingesting data from different sources, or simple quality checks.

The basic building blocks of Wikidata are *items* and *properties*. URIs in Wikidata contain language-independent identifiers, in the form `Qx` for entities, e.g., `Q12345` for the entity of *Count von Count*. Properties are expressed as `Px`, e.g., `P941` for the property *inspired by*. Classes in Wikidata are not differentiated from entities. Previous work has referred to classes as the entities used as object in `P31` (*instance of*) and `P279` (*subclass of*) relationships (Brasileiro et al., 2016). However, the standard for classes by the Wikidata community is vague.[7] Figure 2.3 shows an example of a statement, including labels in natural language in English and Arabic. As with any other part of the knowledge engineering in Wikidata, these labels are created and maintained by the editor community.

### 2.4.1 Labels and Languages

Wikidata is inherently multilingual. To support editors of different languages, a simple mechanism called the *UniversalLanguageSelector* changes the interface language at the editor's will.[8] Changing the interface language of the website enables editors to partake in the knowledge engineering in their native or preferred language. But in addition to this, Wikidata's content comprehensively supports multilinguality in its end user community.

[7] `https://www.wikidata.org/wiki/Wikidata:WikiProject_Ontology/Classes`, retrieved 12. May 2021

[8] `https://www.mediawiki.org/wiki/Extension:UniversalLanguageSelector`, retrieved 12. January 2021

Both entities and properties can be labelled in up to 410 languages (Kaffee and Simperl, 2018b). The criteria for including a language on Wikidata are defined by the Wikimedia Foundation as follows[9]:

- The language has to have an ISO_639-3 code.
- Historic and constructed languages are permitted; newly created words are not.

There are multiple potential sources for Wikidata's labels. They can be created by an editor who also created the Wikipedia page for the corresponding Wikidata item; they might be imported from existing Wikipedia articles; they can be translated from other labels; or they can be imported. Samuel (2018) visualises translation patterns in Wikidata properties. Since properties form the ontology of Wikidata, their labelling is particularly important. Further, as properties are needed to form statements, their reuse is high, and therefore the lack of labels can make the knowledge graph less accessible to human readers. Another necessity is that the label of a property does not change frequently, as it can change the meaning of all relationships between entities in the knowledge graph. Tanon and Kaffee (2018) show that properties' labels across six languages are rarely changed over time

Previous work has explored the coverage of languages and labels in Wikidata and other knowledge graphs. Abián et al. (2017) propose the metric *understandability* for the coverage of labels, because labels are needed for the end user to understand concepts in the knowledge graph. They compare Wikidata, which has multiple languages in the same graph, with DBpedia, which has one graph per language. Wikidata has the higher number of labels and descriptions (136.85 million labels and 222.80 million descriptions in Wikidata vs 38 million labels and abstracts in DBpedia). Färber et al. (2018) propose the dimension *ease of understanding* with the following metrics: *description of resources; labels in multiple languages; understandable RDF serialization; and self-describing URIs*. Wikidata is compared with DBpedia, Freebase, OpenCyc, and YAGO. It performs as well as, or better than, the other knowledge graphs, except in the metric *self-describing URIs*, because Wikidata uses language-independent URIs. Thakkar et al. (2016) explore whether linked open datasets are good candidates for question answering based on Wikidata and DBpedia. Among other metrics, they define two metrics with a focus on languages and labels: *multiple language usage* to measure whether literals are available across languages, and *human-readable labelling*, which considers only `rdfs:label` as labelling property. Since they consider all entities in the RDF format of Wikidata, Wikidata performs worse in the label coverage dimension, because blank nodes are not labelled. In the dimension of *multiple language usage*, Wikidata outperforms DBpedia.

[9] `https://diff.wikimedia.org/2013/11/06/any-language-allowed-in-wikidata/`, retrieved 12. January 2021

## 2.5 Knowledge Graph Transliteration and Translation

The lack of language coverage in knowledge graphs hinders their reuse. However, the manual translation of labels is costly. We therefore present here some different approaches to knowledge graph transliteration and translation, which automatically transfer a label from one language to a target language.

Transliteration transfers a word from one script to another, while translation transfers the meaning of a word from one language to another. For example, people's names are always transliterated.[10] The name *Ada Lovelace* is *transliterated* to 爱达·勒芙蕾丝 in Chinese. The word *thesis* is *translated* into Chinese as 论文. Distinguishing these approaches can be important, because transliteration works on the character or syllable level, while translation needs the context of the concept to translate it correctly.

### 2.5.1 Transliteration

Transliteration has been studied extensively in the context of machine translation (see, e.g., Knight and Graehl (1998); Virga and Khudanpur (2003)). In neural machine translation it has an important role for rare words, as in the work of Sennrich et al. (2016). There are two ways of approaching transliteration: generation and discovery. In the following, we focus on generation, i.e., generating a new transliteration from a source label, as opposed to discovery, in which a transliteration is discovered from an existing dictionary (Upadhyay et al., 2018). The authors describe the challenges presented by the lack of data when working with transliterations. Lin et al. (2016) reuse named entity linking and leveraging semantic information to improve transliteration. One of the possible application domains of transliteration is explored by Udupa and Khapra (2010). Wikipedia users, even if multilingual, might have problems expressing their information needs in English. Especially in the case of names, the transliteration chosen by non-English speakers might be ambiguous and difficult for spellcheckers to correct. Therefore, they argue for a multilingual experience whereby users are permitted to enter their search query in their native language and the system works on cross-lingual name search.

Merhav and Ash (2018) explore neural transliteration systems based on a dataset created from Wikidata. They align person names between English aand a set of four languages that use different scripts (Russian, Hebrew, Arabic, and Japanese Katakana). They split the names into tokens, such that the same first name is always transliterated in the same way. They explore three different systems, inspired by

[10] There are exceptions to names being transliterated but for simplicity we assume people's names are always transliterated. For more information on person name translation see `https://translationjournal.net/journal/50proper.htm`, retrieved 8. June 2020.

libraries: they evaluate the traditional weighted finite state transducer (WFST)) agains two neural approches: the encoder-decoder recurrent neural network method, and the Tensor2Tensor Transformer architecture. The Tensor2Tensor architecture outperforms the other two approaches, but also has a significantly higher training time. Evaluating based on word error rate, they find that English as a source language always brings better results.

### 2.5.2 Knowledge Graph Translation

Enrichment of knowledge graphs with regard to multilinguality is a rising field in both the knowledge graphs and machine translation communities. LabelTranslator (Espinoza et al., 2008a,b) is an early approach to translating the ontology labels of a knowledge graph. Labels are translated and the translations ranked based on their similarity to labels of entities in the context. Similarly, Arcan and Buitelaar (2013) engage with the problem of enriching knowledge graphs with multilingual labels. Using statistical machine translations, the authors translate ontology labels using context vectors based on textual documents about the entities and knowledge graph information. Arcan and Buitelaar (2017) work on domain-specific knowledge graph translation using statistical machine translation. Moussallem et al. (2019) propose neural machine translation of knowledge graph concepts from one language to another using out-of-domain datasets. They use knowledge graph embeddings to facilitate the translation. Those approaches all work with European languages in Latin script, namely, English, Spanish, and German. When working with languages in the same script, transliteration is not needed. However, to support non-European languages, other scripts need to be considered. Yang et al. (2019) translate knowledge graphs based on an attention mechanism for the triples. They work with an English-Chinese dataset, but focus on the translation of whole triples rather than entity labels. Tsai and Roth (2018) differentiate between transliteration and translation of concept names in Wikipedia, but neither use knowledge graph information to make a decision on transliteration or translation. Being able to differentiate between transliteration and translation supports translation approaches besides knowledge graph translation, as can be seen in the work of Hermjakob et al. (2008), because it can be integrated into overall machine translation models.

### 2.5.3 Knowledge Graph Embeddings

In knowledge graph translation it is necessary to encode the information about an entity when translating its label. One approach to encoding the entity's information in the graph is the use of knowledge graph embeddings. This is an emerging field in the research space of embeddings, in which entities in a knowledge graph are embedded in a vector space (Wang et al., 2017). Moussallem et al. (2019); Yang et al. (2019) use fasttext[11] to translate entities in the knowledge graph. However, fasttext is not the state of the art for knowledge graph embeddings, as it is implemented for word embeddings and treats each triple as a set of words rather then entities in a graph. Lerer et al. (2019) provide pretrained embeddings by BigGraph for Wikidata[12], which are designed to support knowledge graphs.

## 2.6 Background on the Motivating Scenarios

We describe two motivating scenarios, or use cases, in Section 1.1: Multilingual question answering and article generation for Wikipedia. In the following, we contextualise our work on these use cases.

### 2.6.1 Multilingual Question Answering

Question answering is the task of retrieving an answer $a$ to a given user's query $q$ over a set of documents $D = d_1, d_2, \ldots, d_n$. We focus on question answering over one or a set of knowledge graphs $kg$, i.e., where the answer to a user's question is retrieved from a knowledge graph, so that $d_i \in kg$.

In multilingual question answering, a user can pose a question in a language and the system is able to process this question and retrieve an answer in the user's language. Multilingual question answering is still a largely unexplored field, as detailed by Loginova et al. (2020). While searches in English can return impressive results, the domain of multilingual question answering is yet to improve its results. The authors identify the three most common approaches to multilingual question answering:

1. Machine translation as part of the system in which user queries, answers, and/or documents are translated. For example, García Santiago et al. (2010) translate user queries to match document languages. Sugiyama et al. (2015) show that cross-lingual question answering is important when answering questions across knowledge graphs, as each knowledge graph is limited in the languages it supports. However, they find that translation aimed at humans does not necessarily correlate with high accuracy for question answering systems across knowledge graphs.
2. Mapping terms to multilingual knowledge graphs. Cabrio et al. (2012) explore translation of user's natural language questions into SPARQL queries for a question answering system (QAKiS) over DBpedia.

[11] `https://radimrehurek.com/gensim/models/fasttext.html`, retrieved 14. January 2021

[12] `https://github.com/facebookresearch/PyTorch-BigGraph#pre-trained-embeddings`, retrieved 14. January 2021

3. Using cross-lingual representations. Zhou et al. (2016) apply this approach in the domain of community question answering.

Multilingual question answering over knowledge graphs requires the availability of a multilingual knowledge graph, to be able to provide the answer in the target language (Thakkar et al., 2016). Diefenbach et al. (2018) survey question answering systems, which answer questions over knowledge graphs. They describe the challenge of multilinguality, e.g., when a user's query is formulated in a different language than the knowledge graph providing the answer. QALD is a well-known dataset in the work of question answering. QALD 4 (Unger et al., 2014) introduces the first multilingual question answering challenge, focusing on seven languages (English, Spanish, German, Italian, French, Dutch, Romanian). However, few systems made use of the availability of the multilingual dataset. Hakimov et al. (2017) are an exception, mapping English, German, and Spanish questions to SPARQL queries to retrieve answers from a knowledge graph. Tanon et al. (2018) introduce Platypus, a question answering system making use of the multilingual data available in Wikidata.

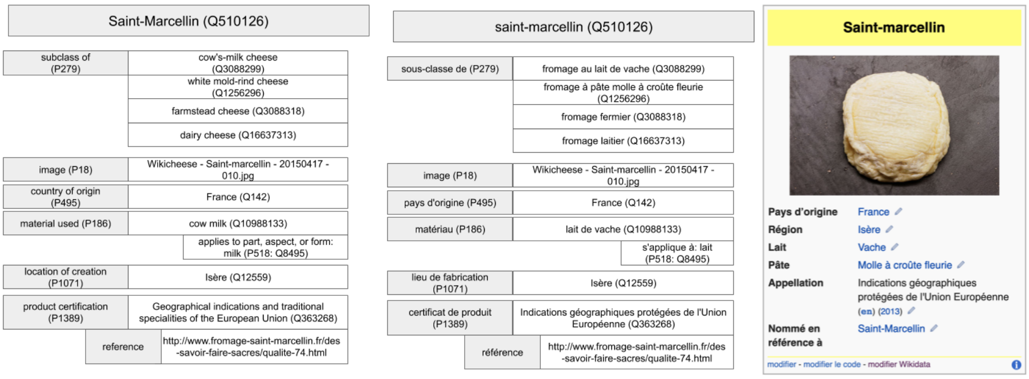
### 2.6.2 Wikipedia

Wikipedia is a community-built encyclopedia and one of the most visited websites in the world[13]. The community of Wikipedia contributes to its content by writing articles, and discussing the content on the discussion pages for each article (Viégas et al., 2007). There are currently 303 active language versions of Wikipedia[14], though coverage is unevenly distributed. Previous studies have discussed several biases, including gender of the editors (Collier and Bear, 2012), and topic bias, for instance a general lack of information on the Global South (Graham et al., 2014).

Language coverage tells a similar story. Pat Wu (2016) noted that only 67% of the world's population has access to encyclopedic knowledge in their first or second language. Another significant problem is caused by the extreme differences in content coverage between language versions. As early as 2005, Voss (2005) found huge gaps in the development and growth of Wikipedias, which has made it more difficult for smaller communities to catch up with the larger ones. Alignment cannot be simply achieved through translation – putting aside the fact that each Wikipedia needs to reflect the interests and points of view of their local community rather than iterate over content transferred from elsewhere, studies have shown that the existing content does not overlap, discarding the so-called *English-as-superset* conjecture for as many as 25 language versions (Hecht and Gergle, 2010; Hecht, 2013). To help tackle these imbalances, Bao et al. (2012) built a system to give users easier access to the language

[13] Alexa rank 13, https://www.alexa.com/siteinfo/wikipedia.org, retrieved 13 October 2020.

[14] https://en.wikipedia.org/wiki/List_of_Wikipedias, retrieved 13 October 2020



| Saint-Marcellin (Q510126) | |
|---|---|
| subclass of (P279) | cow's-milk cheese (Q3088299) |
| | white mold-rind cheese (Q1256296) |
| | farmstead cheese (Q3088318) |
| | dairy cheese (Q16637313) |
| image (P18) | Wikicheese - Saint-marcellin - 20150417 - 010.jpg |
| country of origin (P495) | France (Q142) |
| material used (P186) | cow milk (Q10988133) |
| | applies to part, aspect, or form: milk (P518: Q8495) |
| location of creation (P1071) | Isère (Q12559) |
| product certification (P1389) | Geographical indications and traditional specialities of the European Union (Q363268) |
| reference | http://www.fromage-saint-marcellin.fr/des-savoir-faire-sacres/qualite-74.html |

| saint-marcellin (Q510126) | |
|---|---|
| sous-classe de (P279) | fromage au lait de vache (Q3088299) |
| | fromage à pâte molle à croûte fleurie (Q1256296) |
| | fromage fermier (Q3088318) |
| | fromage laitier (Q16637313) |
| image (P18) | Wikicheese - Saint-marcellin - 20150417 - 010.jpg |
| pays d'origine (P495) | France (Q142) |
| matériau (P186) | lait de vache (Q10988133) |
| | s'applique à: lait (P518: Q8495) |
| lieu de fabrication (P1071) | Isère (Q12559) |
| certificat de produit (P1389) | Indications géographiques protégées de l'Union Européenne (Q363268) |
| référence | http://www.fromage-saint-marcellin.fr/des-savoir-faire-sacres/qualite-74.html |

| Saint-marcellin | |
|---|---|
| Pays d'origine | France |
| Région | Isère |
| Lait | Vache |
| Pâte | Molle à croûte fleurie |
| Appellation | Indications géographiques protégées de l'Union Européenne (en) (2013) |
| Nommé en référence à | Saint-Marcellin |

FIGURE 2.4: Representation of Wikidata statements and their inclusion in a Wikipedia infobox. Wikidata statements in French (middle, English translation to their left) are used to fill out the fields of the infobox in articles using the *fromage* infobox on the French Wikipedia.

diversity of Wikipedia. Our work is motivated by, and complements, previous studies and frameworks that argue that the language of global projects such as Wikipedia (Hecht, 2013) should express cultural reality (Kramsch and Widdowson, 1998).

Wikidata is multilingual by design, and each aspect of the data can be translated and rendered to the user in their preferred language. This makes it the tool of choice for a variety of content integration affordances in Wikipedia, including links to articles in other languages and infoboxes. An example can be seen in Figure 2.4: in the French Wikipedia, the infobox shown in the article about cheese (right) automatically draws in data from Wikidata (left) and displays it in French.

Not only are knowledge graphs used to integrate data in Wikipedia, but the fact that Wikipedia covers various languages has also been the inspiration to build multiple multilingual knowledge graphs extracting Wikipedia information, most prominently YAGO (Mahdisoltani et al., 2015) and DBpedia (Lehmann et al., 2015).

### 2.6.3 Natural Language Generation from Knowledge Graph Data

Generating text from knowledge graph data is an important part of making knowledge graph data accessible to a larger community, which is not able to "read" triples. Especially for the Wikipedia community, this can be a valuable way of creating more content in languages that have a low number of editors. Many of the approaches to generating text from triples rely on templates, which are either based on linguistic features e.g., grammatical rules (Wanner et al., 2010) or are hand crafted (Galanis and Androutsopoulos, 2007). An example is *Reasonator*, a tool for lexicalising Wikidata triples with templates translated by users.[15] Such approaches face many challenges – they cannot be easily transferred to different languages or scale to broader domains, as

[15] https://tools.wmflabs.org/reasonator/

templates need to be adjusted to any new language or domain they are ported to. This makes them unsuitable for Wikipedias which rely on small numbers of contributors.

To tackle this limitation, Duma and Klein (2013) and Ell and Harth (2014) introduced a distant-supervised method to verbalise triples, which learns templates from existing Wikipedia articles. While this makes the approach more suitable for language-independent tasks, templates assume that entities will always have the relevant triples to fill in the slots. This assumption is not always true.

Sauper and Barzilay (2009) and Pochampally et al. (2016) generate Wikipedia summaries by harvesting sentences from the Internet. Wikipedia articles are used to automatically derive templates for the topic structure of the summaries and the templates are afterward filled using web content. Both systems work best on specific domains and for languages like English, for which suitable web content is readily available (Lewis and Yang, 2012).

There is a large body of work that uses the encoder-decoder framework from machine translation (Cho et al., 2014; Sutskever et al., 2014) for NLG (Sleimi and Gardent, 2016; Gardent et al., 2017; Chisholm et al., 2017; Mei et al., 2016; Lebret et al., 2016; Wiseman et al., 2017; Vougiouklis et al., 2018; Liu et al., 2018; Gehrmann et al., 2018; Yeh et al., 2018). Adaptations of this framework have shown great potential for tackling various aspects of triples-to-text tasks, ranging from microplanning, by Gardent et al. (2017) to generation of paraphrases, by Sleimi and Gardent (2016). Mei et al. (2016) sought to generate textual descriptions from datasets related to weather forecasts and RoboCup football matches. Wiseman et al. (2017) used pointer-generator networks (See et al., 2017) to generate descriptions of basketball games, while Gehrmann et al. (2018) did the same for restaurant descriptions.

A different line of research explores knowledge bases as a resource for NLG (Duma and Klein, 2013; Bouayad-Agha et al., 2014; Lebret et al., 2016; Chisholm et al., 2017; Vougiouklis et al., 2018; Liu et al., 2018; Yeh et al., 2018). In all these examples, linguistic information from the knowledge base is used to build a parallel corpus containing triples and equivalent text sentences from Wikipedia, which is then used to train the NLG algorithm. Directly relevant to the model we propose are the proposals by Lebret et al. (2016), Chisholm et al. (2017), Liu et al. (2018), Yeh et al. (2018) and Vougiouklis et al. (2018, 2020), which extend the general encoder-decoder neural network framework from Cho et al. (2014); Sutskever et al. (2014) to generate short summaries in English. Generation of English biographies was introduced by Lebret et al. (2016), who used feed-forward language model with slot-value templates to generate the first sentences of Wikipedia summaries from their corresponding infoboxes.

All these approaches use structured data from Freebase, Wikidata, and DBpedia as input and generate summaries consisting either of one or two sentences that match the style of the English Wikipedia in a single domain (Lebret et al., 2016; Chisholm et al., 2017; Vougiouklis et al., 2018; Liu et al., 2018; Yeh et al., 2018) or, more recently, generate summaries in open-domain scenarios (Vougiouklis et al., 2020). While this is rather a narrow task compared to other generative tasks such as translation, Chisholm et al. (2017) discuss its challenges in detail and show that it is far from being solved.

#### 2.6.3.1 Evaluating Text Generation Systems

Related literature suggests three ways of determining how well an NLG system achieves its goals. The first, which is commonly referred to as *metric-based corpus evaluation* (Reiter and Belz, 2009), use text-similarity metrics such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004) and METEOR (Lavie and Agarwal, 2007). These metrics essentially compare how similar the generated texts are to texts from the corpus. The other two involve people, and are either *task-based* or *judgement/rating-based* (Reiter and Belz, 2009). Task-based evaluations assess how an NLG solution assists participants in undertaking a particular task, for instance learning about a topic or writing a text. Judgement-based evaluations rely on a set of criteria against which participants are asked to rate the quality of the automatically generated text (Reiter and Belz, 2009).

Metric-based corpus evaluations are widely used as they offer an affordable, reproducible way to automatically assess the linguistic quality of the generated texts (Reiter and Belz, 2009; Angeli et al., 2010; Konstas and Lapata, 2013; Lebret et al., 2016; Chisholm et al., 2017; Kaffee et al., 2018b). However, they do not always correlate with manually curated quality ratings (Reiter and Belz, 2009).

Task-based studies are considered most useful, as they allow system designers to explore the impact of the NLG solution to end users (Mellish and Dale, 1998; Reiter and Belz, 2009). However, they can be resource intensive - previous studies by Reiter et al. (2003); Williams and Reiter (2008) cite five-figure sums when data analysis and planning are included (Reiter, 2010). The system by Williams and Reiter (2008) was evaluated for the accuracy of the generated literacy and numeracy assessments by a sample of 230 participants, which cost as much as £25, 000. Reiter et al. (2003) describe a clinical trial with over 2000 smokers costing £75, 000, assumed to be the most costly NLG evaluation at this point. All of the smokers completed a smoking questionnaire in the first stage of the experiment, in order to find what portion of those who received the automatically generated letters from STOP had managed to quit.

Given these challenges, most research systems tend to use judgement-based rather than task-based evaluations (Sun and Mellish, 2007; Reiter and Belz, 2009; Angeli et al., 2010; Konstas and Lapata, 2013; Duma and Klein, 2013; Ngonga Ngomo et al., 2013; Ell and Harth, 2014; Chisholm et al., 2017). However, beside the problem of their limited scope, most studies in this category do not recruit from the relevant user population, relying on more accessible options such as online crowdsourcing. Sauper and Barzilay (2009) are a rare exception. In their paper, the authors describe the generation of Wikipedia articles using a content-selection algorithm that extracts information from online sources. They test the results by publishing 15 articles about diseases on Wikipedia and measuring how the articles change (including links, formatting, and grammar). Their evaluative approach is not easy to replicate, as the Wikipedia community tends to disagree with conducting research on their platform.[16]

[16] https://en.wikipedia.org/wiki/Wikipedia:What_Wikipedia_is_not#Wikipedia_is_not_a_laboratory, retrieved 16. January 2021

# Chapter 3

# Framework to measure labels and languages

The web of data is an invaluable resource for humans and computers alike. While its main benefits are often explained in the context of linked data principles, in many applications making linked data genuinely useful also means attaching natural language representations to URIs, in one or more languages. There are many examples to illustrate this, from search (Cheng and Qu, 2009), text generation (Kaffee et al., 2018a), browsing (Berners-Lee et al., 2006), and visualisation (Helmich et al., 2014) to question answering (Diefenbach et al., 2018; Höffner et al., 2017) and ontology modeling (Peroni et al., 2013).

In linked data, resources can be accompanied by human-readable labels, descriptions, or comments using a range of pre-defined properties. Additionally, text can be marked with a language tag, such as *@en* for English, to support multilingual applications.

In previous work, Ell et al. (2011) introduced a framework to study the human readability of the web of data. The framework has two parts. One is a method for collecting different natural language representations of URIs in a linked dataset. The other is a set of metrics to assess different dimensions of human readability: completeness, efficiency of access, unambiguity, and multilinguality. They apply the framework to the 2010 edition of the Billion Triple Challenge (BTC) corpus, a representative sample of the web of data at the time of publication, and conclude that more labels are needed to encourage uptake of the use of the data across a greater range. To take into account more factors important to the languages and labels in a knowledge graph, we extend the metrics of Ell et al. (2011) and create a comprehensive framework that can be used for any linked data graph to assess the coverage of human-readable data, i.e., labels and their languages.

In this chapter, we introduce this framework, which is used for the studies detailed in the chapters that follow. In Chapter 4, we use this framework to measure labels and languages in the web of data, and in particular the LOD Laudromat dataset. In Chapter 5, the framework is applied to Wikidata. Finally in Chapter 6, we show a use case of the framework in the domain of question answering systems.

Establishing a framework for labels and languages aims to integrate previous efforts on defining guidelines and recommendations for knowledge graph labels. It also sets the baseline for our work on translations and generation of text from triples – we gain insight into what data is currently available and which parts of the data still need improvement.

In the following, we outline our framework for measuring labels and languages in knowledge graphs. We apply different metrics for different studies, as suited to the observed data. An overview of all metrics in the framework can be found in Table 3.2. We split the metrics into three categories: the metrics that describe the dataset (*dataset description*), the core metrics needed to identify coverage of languages in a dataset (*core metrics*), and finally the set of metrics to describe multilinguality in the data creators (*data creator metrics*).

A dataset $D$ can be defined as a set of triples $s, p, o$, where each $s \in S$, $p \in P$, and $o \in O$ denotes the set of unique subjects properties, and objects of $T$. Classes are a type of entity, typically denoted with the `rdf:type` property. We define a class as $c \in C$. A $label \in Label$ is a string value describing the preferred name of an entity. A label is associated with a language $l \in L_D$, where $L_D$ is the set of all languages in dataset $D$. In the *data creator* metrics, we describe the interaction of a user (or editor) $u \in U$ with the dataset in edits to the labels. We will use edit and label edit interchangeably if not specified otherwise. An edit $e \in E$ denotes such a label edit. $E$ is an array ordered by time containing all edits of all users $U$, so that $E_u$ is a subarray of $E$ containing all edits of a user $u$. For notation simplicity, we define a handy operator lang, which returns the unique number of languages associated with a collection. This collection could be a set of users, an array of edits, or a set of entities. For example $\text{lang}(E_u)$ returns a set of unique languages for all edits of a user $u$, and $\text{lang}(S)$ is the set of the unique languages of all labels for the entities in $S$, and so forth.

**Creation of the Framework** The framework is based on literature on guidelines and existing frameworks to measure languages and labels for multilingual knowledge graphs. We select the appropriate metrics and make them actionable in this chapter. In Table 3.1, we lay out the different sources for the framework, not all of them applicable to actual data in the original work. An overview on existing guidelines that create the base for our framework can be found in Section 2.3.1. Some of the metrics (such as the

| Metric | |
|---|---|
| Entity label completeness | Ell et al. (2011); Zaveri et al. (2016); Debattista et al. (2016) |
| Class label completeness | Ell et al. (2011); Zaveri et al. (2016); Debattista et al. (2016) |
| Property label completeness | Ell et al. (2011); Zaveri et al. (2016); Debattista et al. (2016) |
| Unambiguity | Ell et al. (2011) |
| Multilinguality | Gómez-Pérez et al. (2013); Ell et al. (2011) |
| Monolingual Islands | Gracia et al. (2012) |
| Contextual comparison | Work in low-resource languages (see Section 1.3) |

TABLE 3.1: Core metrics in the framework and literature the metrics are based on.

unambiguity, see Table 3.2) are directly adapted from previous work such as Ell et al. (2011), some metrics (such as the completeness measures) are extended for granularity. In the case of completeness, we want to measure it also on class- and property-level. A third set of metrics, such as monolingual islands, are mentioned in previous work but have not been previously formalised for application in a framework. The data creator metrics are created based on preliminary investigations of the data and derived from conversations with researchers and community members working on Wikidata. The metrics together cover a wide range of metrics, previously described or created for this framework, which enables researchers and practitioners to understand the coverage of knowledge graphs by languages and labels.

## 3.1 Dataset Description

In this set of metrics, we focus on describing the dataset at hand. Those descriptive metrics are essential, foir gaining a first insight into the dataset overall.

**Dataset size** *Size of the dataset in triples.* To be able to set the following metrics in context, we want to measure the size of the dataset. We measure the size of a dataset $D$ in number of triples. Let $T$ be the set of triples, so that any triple in $D$ is $t_n \in T$. The size of dataset $D$ is then $|T|$. For the experiments in Chapter 6, we add the same measurement (size in triples), to measure the size of classes (see more details in Section 6.2.2). Let $C$ be a class in the dataset $D$, and $T_C$ be the set of all triples $T_C \in C$, so that we can measure the number of triples in a class is $|T_C|$.

**Natural Language URI** *Type of URI used* Each entity in the semantic web has a unique ID, which can be identified with a so-called URI. In their functionality URIs clearly differ from labels (Montiel-Ponsoda et al., 2011). While labels are a way for humans to interact with data in natural language, URIs serve as identifiers for concepts that, ideally, do not have to change. Montiel-Ponsoda et al. (2011) encourage

| Metric | |
|---|---|
| **Dataset description** | |
| Dataset size | *Size of the dataset in triples* |
| Natural Language URI | *Type of URI used* |
| Labelling Properties | *Properties used for labelling* |
| **Core metrics** | |
| Entity label completeness | *Coverage of entities in terms of labels*<br>$C_S = \frac{\lvert S_L \rvert}{\lvert S \rvert}$ |
| Class label completeness | *Coverage of classes in terms of labels*<br>$C_C = \frac{\lvert C_L \rvert}{\lvert C \rvert}$ |
| Property label completeness | *Coverage of properties in terms of labels*<br>$C_P = \frac{\lvert P_L \rvert}{\lvert P \rvert}$ |
| Unambiguity | *Conflicting labels for one entity in the same language*<br>$UN = \frac{\lvert S_U \rvert}{\lvert S \rvert}$ |
| Multilinguality | *Language diversity and coverage*<br>$M_l = \frac{\lvert \{label \in Label \mid \mathrm{lang}(label)=l\} \rvert}{\lvert Label \rvert}$ |
| Monolingual Islands | *Entities labelled in more than one language*<br>$I_n = \frac{\lvert \{s \in S \mid \mathrm{lang}(s) \leq n\} \rvert}{\lvert S \rvert}$ |
| Contextual comparison | *Comparison of language coverage*<br>*Multilinguality* metric applied on $D$ and native speakers |
| **Data creator metrics** | |
| User language | *Language distribution in users' languages*<br>$UL_l = \frac{\lvert \{u \in U \mid \mathrm{lang}(u)=l\} \rvert}{\lvert U \rvert}$ |
| User language editing | *Correlation between user and label languages*<br>$ULE = \mathrm{corr}(M^L, UL^L)$ |
| User activity | *Set of metrics to understand languages edited by users*<br>$UA_1 = \frac{\lvert Edit \rvert}{\lvert U \rvert}, UA_2 = \mathrm{lang}(Edit),$<br>$UA_3 = \frac{\sum_u \lvert \mathrm{lang}(Edit_u) \rvert}{\lvert U \rvert}, UA_4 = \frac{\lvert U \rvert}{\lvert \mathrm{lang}(Edit) \rvert}$ |
| Edit patterns | *Editors' tendency to either translate labels or create new ones*<br>$jlc = \frac{\sum_u \mathrm{jumps}([e_{lang} \mid e \in E_u])}{\lvert U \rvert}$<br>$jec = \frac{\sum_u \mathrm{jumps}([e_{entity} \mid e \in E_u])}{\lvert U \rvert}$ |
| Language overlap | *Language edit graph* |
| Activity and Multilinguality | *Correlation between user activity and multilinguality*<br>$AM = \mathrm{corr_u}(\lvert E_u \rvert, \lvert \mathrm{lang}(E_u) \rvert)$ |

TABLE 3.2: Overview of all metrics in the framework.

the usage of opaque URIs, that is language-independent identifiers[1]. Opaque URIs can contain any form of ID that is *not* a word from any natural language, such as a numeric value. They should be independent from the actual content of an entity. The authors argue these will prevent bias toward English or any other language, and are therefore a better choice for ontologies which will support descriptions of concepts in multiple languages. URIs by definition should not change. Therefore, if names of

[1] This follows also the recommendations of http://www.w3.org/Provider/Style/URI

concepts are amended, a natural language URI might point to a deprecated label, while an opaque URI do not face these problems, as these are defined independently of natural language labels.

We describe the dataset in terms of natural language URIs by extracting a sample of URIs and manually sorting these into types. In Chapter 4, we attempt an automation based on the categories we find in the data (see Section 4.3). We identify three categories of URI: (1) identifier in natural language (mostly English), such as `Tiger`; (2) identifier that is a mix of natural language (mostly English) and a numeric identifier, e.g., `person-164999`; (3) completely numeric identifier, e.g., `980891`. We automate the classification of URIs into these three categories by sorting identifiers into characters only; mix of character types; and numerical values only. This does not give us complete insight into whether a full word is formed. However, it gives some preliminary information about the distribution of URIs. We choose to sort by character types rather than by doing dictionary look-ups, as we cannot assume language in the URIs.

**Labelling Properties** *Properties used for labelling.* We compile a list of properties that are used to add human-readable labels to entities. In Ell et al. (2011), this list consists of 36 properties, which have been curated manually based on data from the BTC 2010 corpus. These include `rdfs:label`, as well as several other properties in commonly used vocabularies such as FOAF, SKOS, and Dublin Core. Most datasets use several properties to attach textual information to entities besides the recommended `rdfs:label` (Brickley and Guha, 2004). This complicates the automatic reuse of this information – applications need to be aware of the different ways in which the information is expressed (Saleem et al., 2016) and decide which parts to display to the user and how. Based on the list of properties, we then collect the labels and analyse them. To understand how a dataset is labelled, we identify the most used properties in a corpus that refer to a string value. From this set of properties, we manually select the ones used for labelling. After collecting the labelling properties, we measure the occurrence of each labelling property over the whole dataset.

## 3.2 Core Metrics

This set of metrics describes the coverage, in terms of labels and languages, of a dataset $D$.

**Entity label completeness** *Coverage of entities in terms of labels.* To improve data accessibility, each entity in the data should have at least one label. This metric is based on previous work on labelling on the web of data, as conducted by Ell et al. (2011), Debattista et al. (2016), and Zaveri et al. (2016), who call this metric *human-readable labelling*. Considering a dataset consisting of triples made of subjects, predicates, and objects, entity label completeness $C_S$ is defined as the ratio of subjects that have at least one label compared to all subjects in the dataset. Let $S$ be the set of all the unique entities and $S_L$ the set of entities that have at least one label, we compute $C_S$ as follows:

$$C_S = \frac{|S_L|}{|S|} \quad , \tag{3.1}$$

where $|S_L|$ and $|S|$ denote the cardinality of those two sets such that $|S_L| \leq |S|$. We only consider `rdfs:label` as it is the most used labelling property according to Ell et al. (2011), and the only labelling property considered in applications such as question answering systems (Diefenbach et al., 2018). The metric does not differentiate between languages. Each label, English or otherwise, with or without a language tag, is considered. We want to understand the overall coverage of entities with labels, as we measure other factors taking language into account in other metrics and we propose an approach of translating labels between languages in Chapter 7.

**Class label completeness** *Coverage of classes in terms of labels.* Analogously to *entity label completeness*, class label completeness $C_C$ measures the coverage of classes by labels. We showed in a previous study that a lack of class labelling contributes to low coverage of labels overall (Kaffee and Simperl, 2018a). Therefore, we standardise testing for class labelling in this metric. Let $C$ be the set of all classes in $D$, which are identified by the `rdfs:type` property, and $C_L$ the set of classes that have at least one label using `rdfs:label`, we compute $C_C$ as follows:

$$C_C = \frac{|C_L|}{|C|} \tag{3.2}$$

**Property label completeness** *Coverage of properties in terms of labels.* Analogously to *entity label completeness* and *class label completeness*, property label completeness $C_P$ measures the coverage of properties by labels. Properties are essential for the structure of a knowledge graph and are highly reused across a knowledge graph (Tanon and Kaffee, 2018). Let $P$ be the set of all properties in $D$, and $P_L$ the set of properties that have at least one label using `rdfs:label`, we compute $C_P$ as follows:

$$C_P = \frac{|P_L|}{|P|} \tag{3.3}$$

**Unambiguity** *Conflicting labels for one entity in the same language.* If a user wants to access an entity, a system has to decide which natural language label should be displayed. We define unambiguity as a resource having only one label per entity using the `rdfs:label` property, making accessing the label for the entity e.g., for querying simple. In other words, each entity $s_j \in S$ in the dataset $D$ should have only one label. As one entity can be labelled multiple times across languages, we focus on the most used language across the dataset. Labelling the same entity across languages is desirable, as we explain in the *multilinguality* metric. We define unambiguity $U$ as the proportion of entities that have no duplicated language information compared to the number of all entities in $S$. Formally, let $S_U \subseteq S$ be the set of entities that have no duplicated language information, we compute:

$$UN = \frac{|S_U|}{|S|} \tag{3.4}$$

**Multilinguality** *Language distribution.* To be able to cater to readers of different languages, it is necessary to provide information in multiple languages. Compared to previous metrics, where we define human readability across all languages, we want to gain an insight here into the languages provided by the dataset. We measure the multilinguality of the dataset in two steps. First, we want to understand how many languages in total are present in the dataset as of now. Multilinguality of a dataset $D$ is measured by the number of languages the entities cover overall, $|L_D|$.

Then, we want to gain a sense of the distribution of all languages across the dataset, so we can find, for example, 30% of all labels are in Spanish. Each entity $s$ can have multiple labels $label$, of which each is associated with a language code for the language $l$. For example, the entity `Berlin` can have the English label `"Berlin"@en`, so that the $(label, l)$ pair would be $(Berlin, en)$.

To measure language distribution of the dataset, we calculate the share of labels in each language $l \in L_D$. We count the number of labels for each language, so that $Label$ is the set of all labels in the dataset, and $\{label \in Label | \operatorname{lang}(label) = l\}$ the set of all labels in language $l$.

For each of the languages $l$ in $L_D$, we calculate the language distribution as:

$$M_l = \frac{|\{label \in Label | \operatorname{lang}(label) = l\}|}{|Label|} \tag{3.5}$$

We calculate $M_l$ for all languages $l$ in $L_D$.

**Monolingual islands** *Entities labelled in more than one language.* Gracia et al. (2012) define monolingual islands as subsets of data in a dataset which are labelled in one language and not linked to labels in any other language. This could be, for example, a topic area being labelled only in English, making it inaccessible to other language speakers. In terms of multilinguality, it is not only important to measure how many languages a dataset covers, but also how well information between those languages is connected. Therefore, we measure how many entities are available in multiple languages. We calculate the monolingual islands metrics as:

$$I_n = \frac{|\{s \in S|\, \text{lang}(s) \leq n\}|}{|S|} \tag{3.6}$$

Monolingual islands can be observed when $n = 1$, meaning if $I_1 = 90\%$ the largest share of the dataset is only labelled in one language. This creates a high amount of monolingual islands.

**Contextual comparison** *Comparison of language coverage.* To set the findings of multilinguality into context, we compare the findings with the world at large. It is challenging to define an *ideal* language coverage, so we opt to compare our findings with native speakers in the world. In an ideal scenario, anyone would be able to access information in their native language, i.e., all entities would be labelled in all languages. While choosing to compare with native speakers only is very simplified (e.g., not taking multilingual speakers into account), we want to understand how close the datasets in their language distribution are to the world at large, to create a starting point for comparison.

To explore how diverse the language distribution of a dataset is, we compare it with the native speakers in the world. We first collect the Wikipedia language codes for the top 100 languages by number of native speakers, as in Parkvall (2007). We then compare the distribution of native language speakers with the labels in the dataset, to see how well each language community is covered by human-readable knowledge in the dataset.

In the case of Wikidata (Chapter 5), we are also interested in how the knowledge graph compares to Wikipedia, given their close relationship (see Section 2.6.2). We use the ranking of Wikipedias based on the numbers of articles for each language version[2]. We compare it to the ranking based on the *multilinguality* metric, to see whether they are similar and to get an insight into the relation between Wikipedia's and Wikidata's multilingual information.

## 3.3 Data Creator Metrics

In a knowledge graph that is edited by a community, we can access information about the users and how they edit language information. This is useful for describing the diversity of the community, and how likely the community is to cover all languages

[2] https://meta.wikimedia.org/wiki/List_of_Wikipedias, retrieved August 2017

represented in the knowledge graph. This information can also be used to develop tools for the community, to support them in editing labels in the knowledge graph.

**User Language** *Language distribition in users' languages.* In this metric, we take two possibilities of languages of users into account:

1. **Language setting** The language setting of a website is the setting of its interface language. In the case of Wikidata (see Chapter 5), the language setting is used to select the language in which interface elements, such as the edit button, are displayed. A user can set only one language in the language setting.
2. **User-identified languages** User-identified languages let a user self-report the languages they know, and their level of fluency in each language. In the case of Wikidata, the scale for fluency in a language is from 0 (no knowledge) to N (native speaker). These user-identified languages are not mandatory, i.e., some users identify their language and others choose not to do so. We define *known languages* as languages with a score higher than 1, i.e., excluding languages of levels 0 and 1. We define *unknown languages* as languages undeclared and languages with level 0 or 1.

Analogously to the metric *Multilinguality*, we calculate the User Language setting $UL$ as the distribution of user languages for each interface and the user set language. $\mathrm{lang}(u)$ denotes the preferred language as provided by user $u$. We calculate for each language $l$ in $L_U$:

$$UL_l = \frac{|\{u \in U|\ \mathrm{lang}(u) = l\}|}{|U|} \tag{3.7}$$

**User Language Editing** *Correlation between user languages and label languages.* An important factor for the data creator metrics is to understand how the languages of the users are connected to the languages of the labels in the dataset. Based on the metrics *Multilinguality* and *User language*, the metric User Language Editing $ULE$ connects the languages users self-identify with (*user-identified languages*) and the label languages in the dataset $D$.

Let $M_L = \{M_{l_1}, M_{l_2}, \ldots, M_{l_n}\}$ be the set of all distributions of languages $l$ calculated in the Multilinguality metric, Equation (3.5). And let $UL_L = \{UL_{l_1}, UL_{l_2}, \ldots, UL_{l_n}\}$ be the set of distributions of user-identified languages calculated in the User Language metric, Equation (3.7). Using a correlation algorithm corr[3], we calculate the correlation between user and label languages as:

$$ULE = \mathrm{corr}(M_L, UL_L) \tag{3.8}$$

[3] Correlation is calculated with python's numpy.

**User Activity** *Set of metrics to understand languages edited by users.* The user activity metrics contain descriptive metrics regarding the label editing activity of editors. These can be applied to different user types to compare them, as done in Chapter 5. We describe a set of metrics $UA_n$, summarising the edits of editors to labels in a language. We define editor $u$ as any community member who has edited a label, so that $U$ is the set of all editors who have made at least one edit $(edit, l)$ to a label *label* in language $l$, so that $Edit_u$ is the set of all label edits for an editor and $Edit$ the label edits of all editors in $U$, i.e., $Edit_u \subseteq Edit$. ${Edit_u}^l$ are all label edits for an editor in language $l$. We define an edit as an edit to a label, if not otherwise stated. The descriptive metrics are as follows:

**UA₁** The average number of edits per editor is calculated as

$$UA_1 = \frac{|Edit|}{|U|} \tag{3.9}$$

**UA₂** The overall languages covered by all editors

$$UA_2 = \text{lang}(Edit) \tag{3.10}$$

**UA₃** The average number of languages edited per editor

$$UA_3 = \frac{\sum_u |\,\text{lang}(Edit_u)|}{|U|} \tag{3.11}$$

**UA₄** The average number of editors per language

$$UA_4 = \frac{|U|}{|\,\text{lang}(Edit)|} \tag{3.12}$$

**UA₅** Edit timeline, exploring the edits over time by summing the edit counts per month

**UA₆** Comparison of edit count and editor count per language

**Edit Patterns** *Editors' tendency to either translate labels or create new ones.* The metric edit patterns describes the different ways of editing over time by an editor $e$. We measure editing patterns $EP$ by measuring the *jumps* between different languages and entities. We define the operator jumps, which returns the number of changes or jumps on an array. For each edit made, we count the number of jumps between languages over time. For example, an editor editing (en, en, fr) would have a jump count *jlc* of 1, i.e., from en to fr; someone editing (fr, de, fr) would have a jump count *jlc* of 2, i.e., from fr to de and then de to fr, i.e., $\text{jumps}([fr, de, fr]) = 2$. Analogously, we measure jumps between entities. A user editing Berlin's (Q64) label in German and then in

French, moving on to the label of the item for London (Q84) in Amharic, i.e., (Q64, Q64, Q84) would have an entity jump count *jec* of 1, i.e., $\text{jumps}([Q64, Q64, Q84]) = 1$. Generally, there are two editing patterns we focus on: first, the part of the community that edits more in one language, and therefore has a higher count in jumps of entities *jec* and lower in languages *jlc*, i.e., $jec > jlc$; and second, the editors that have a higher count in jumps of languages and lower in entities $jec < jlc$, meaning they translate labels on entities.

$$jlc = \frac{\sum_u \text{jumps}([e_{lang} | e \in E_u])}{|U|} \tag{3.13}$$

$$jec = \frac{\sum_u \text{jumps}([e_{entity} | e \in E_u])}{|U|} \tag{3.14}$$

where $e_{lang}$ denotes the language of an edit, and $e_{entity}$ the entity the edit was made for.

**Language Overlap** *Language edit graph.* The metric language overlap measures how editors edit languages. We create a language network graph where each node represents a language and the edge represents the cross-lingual edits by one or more editors. The weight of the edges represents the number of editors that share this language pair. A language pair is the overlap of an editor that edits those two languages. For example, an editor that edits French, German, and English creates three connections between those languages (fr-de, de-en, fr-en).

**Activity and Multilinguality** *Correlation between user activity and multilinguality.* The metric activity and multilinguality tests the hypothesis that a higher number of distinct languages per editor is connected to a higher edit count. This is based on the work of Hale (2014), who show that multilingual editors are more active than their monolingual counterparts on Wikipedia. We calculate the correlation of those values with Pearson's r[4]. Using Pearson's r correlation $corr_u$, the set of edits of an editor $u$ as $|E_u|$, and the number of languages an editor edited in $|L_e|$ for all editors $u$ in $U$, we calculate[5]:

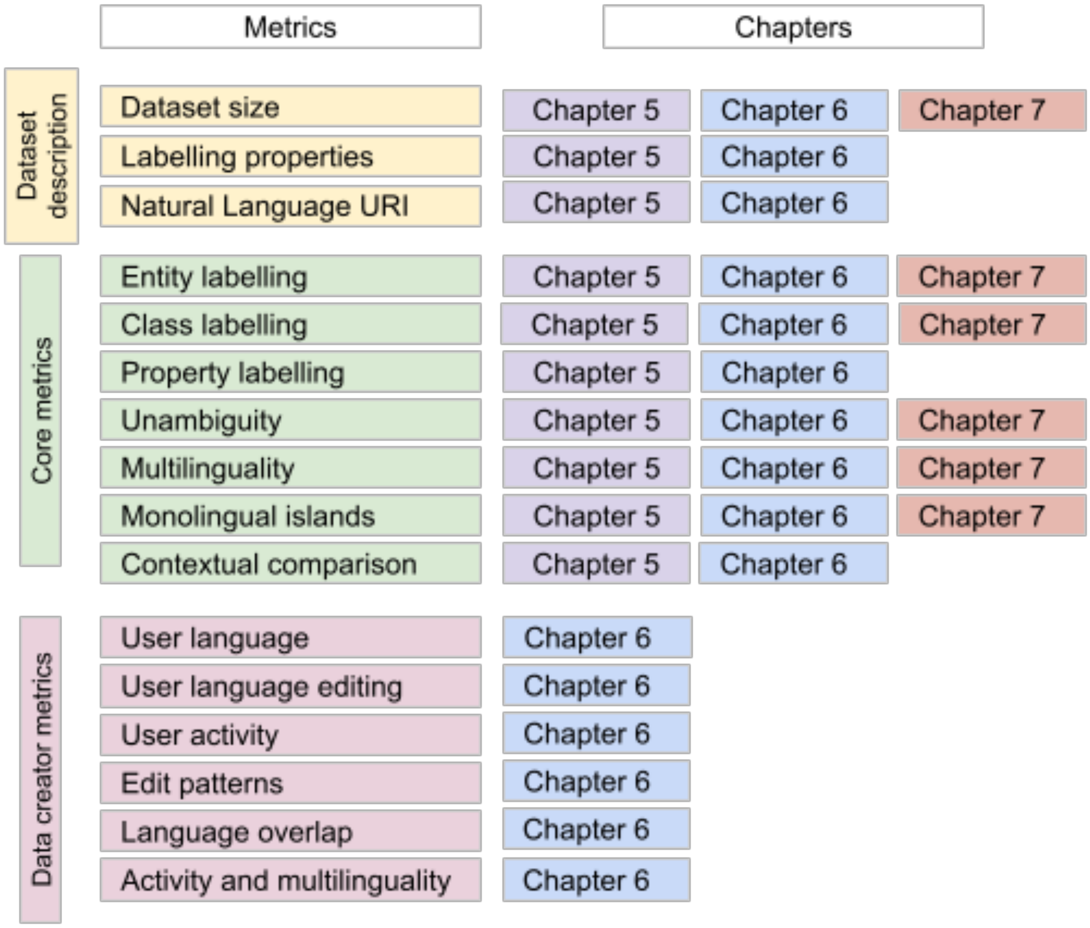
$$AM = \text{corr}_\text{u}(|E_u|, |\,\text{lang}(E_u)|) \tag{3.15}$$

## 3.4 Application of the framework

The framework described in this chapter is used in the following studies to understand the coverage of labels and languages on the web of data at large, using the

[4] `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html`, retrieved 16. May 2021

[5] For clarity we use $\text{corr}_\text{u}(x_u, y_u)$ to denote Pearson's r correlation between the different values of x and y across each user $u$.



| | Metrics | Chapters | | |
|---|---|---|---|---|
| Dataset description | Dataset size | Chapter 5 | Chapter 6 | Chapter 7 |
| | Labelling properties | Chapter 5 | Chapter 6 | |
| | Natural Language URI | Chapter 5 | Chapter 6 | |
| Core metrics | Entity labelling | Chapter 5 | Chapter 6 | Chapter 7 |
| | Class labelling | Chapter 5 | Chapter 6 | Chapter 7 |
| | Property labelling | Chapter 5 | Chapter 6 | |
| | Unambiguity | Chapter 5 | Chapter 6 | Chapter 7 |
| | Multilinguality | Chapter 5 | Chapter 6 | Chapter 7 |
| | Monolingual islands | Chapter 5 | Chapter 6 | Chapter 7 |
| | Contextual comparison | Chapter 5 | Chapter 6 | |
| Data creator metrics | User language | Chapter 6 | | |
| | User language editing | Chapter 6 | | |
| | User activity | Chapter 6 | | |
| | Edit patterns | Chapter 6 | | |
| | Language overlap | Chapter 6 | | |
| | Activity and multilinguality | Chapter 6 | | |

FIGURE 3.1: Metrics and the respective chapters they are applied in. Metrics are used and applied based on the need of the respective chapter.

LOD Laundromat dataset in Chapter 4, and looking at Wikidata specifically in Chapter 5. We show the usability of the framework in comparing the two datasets in Section 5.20. We further show a use case of the metrics in Chapter 6, in which we use a subset of the metrics for ranking question answering systems. Figure 3.1 displays the usage of the metrics across the three chapters. We apply the metrics as needed in the different chapters. For example, for the datasets used in Chapter 4 and 6, we do not have information about data creators, so we do not apply the set of data creator metrics.

# Chapter 4

# Multilinguality in the Web of Data

## 4.1 Dataset

Datasets on the web of data are published by a variety of authors and institutions and often do not follow the guidelines for linked data. This complicates integration across different datasets. The LOD laundromat (Beek et al., 2014) extracts the data of the web of data, processes it, and unifies it, to publish it in N-triples format. The dataset is published at `https://krr.triply.cc/krr/lod-a-lot`. The provided version accessed in January 2021 contains $28,362,196,682$ triples, from a variety of knowledge graphs accessible on the web. Unifying the data makes it easy to reuse, allowing insight into a variety of datasets on the web of data. While the LOD cloud contains 1301 datasets[1], we can extract $1,569,320$ different namespaces (i.e., prefixes of URIs such as `https://tr.dbpedia.org`) from the LOD Laundromat dataset. These include many large knowledge graphs on the web of data such as DBpedia, and Wikidata. Therefore, we treat the LOD laundromat as a representation of the web of data.

## 4.2 Size

LOD Laundromat contains $28,362,196,682$ triples, containing various other knowledge graphs on the web of data, such as DBpedia.

[1] As stated on the website, `https://lod-cloud.net/`, retrieved 11.May 2021

## 4.3 Natural Language URIs

To identify the types of URIs in the set, we sampled 1% of the dataset randomly, i.e., 283,607,766 lines of the file. In a first manual investigation, we were able to detect three different types of identifier for entities:

1. Identifier in natural language (mostly English), such as `Tiger`
2. Identifier is a mix of natural language (mostly English) and a numeric identifier, e.g., `person-164999`
3. Identifier completely numeric, e.g., `980891`

Among the sampled 283,607,766 entities, there are 235,304,338 unique URIs. Those can be split into the above mentioned categories as follows: there are 19,579,809 identifiers containing only natural language (8.3%); there are 207,863,969 identifiers containing an identifier that is a mix of characters and numerical values (88.3%); and there are 7,860,560 identifiers that only contain numeric values (3.3%).

As described in Section 3.1, we automate classification into these three categories by sorting the identifiers into only letter characters; mix of alphanumeric characters; and only numerical values.

Applying this approach to the entire dataset, we get the following distributions: 4.7% of the identifiers contain only natural language; 88.8% of the identifiers contain a mix of characters and numerical values; and 6.5% of the identifiers contain only numerical values.

| Labelling property | Usage |
|---|---:|
| <http://www.w3.org/2000/01/rdf-schema#label> | 246201989 |
| <http://xmlns.com/foaf/0.1/name> | 16590050 |
| <http://www.w3.org/2004/02/skos/core#prefLabel> | 12270826 |
| <http://www.loc.gov/mads/rdf/v1#authoritativeLabel> | 10064682 |
| <http://purl.org/dc/terms/title> | 6721999 |
| <http://purl.org/dc/elements/1.1/title> | 3497188 |
| <http://rdf.freebase.com/ns/type.object.name> | 3496386 |
| <http://lexvo.org/ontology#label> | 2204386 |
| <http://www.w3.org/2008/05/skos-xl#literalForm> | 1028504 |
| <http://sw.cyc.com/CycAnnotations_v1#label> | 516098 |
| <http://purl.org/rss/1.0/title> | 507321 |
| <http://www.livejournal.org/rss/lj/1.0/journaltitle> | 269947 |
| <http://usefulinc.com/ns/doap#name> | 219296 |
| <http://purl.org/goodrelations/v1#name> | 189025 |
| <http://www.w3.org/2006/03/wn/wn20/schema/lexicalForm> | 146842 |
| <http://www.geonames.org/ontology#officialName> | 96647 |
| <http://www.w3.org/2006/03/wn/wn20/schema/gloss> | 84591 |
| <http://www.w3.org/2004/02/skos/core#hiddenLabel> | 80628 |

| | |
|---|---|
| <http://rdf.freebase.com/ns/biology.organism_classification.scientific_name> | 77292 |
| <http://rdf.insee.fr/geo/nom> | 41230 |

TABLE 4.1: The 20 most frequently used labelling properties used in the LOD Laundromat dataset, manually selected. The most used labelling property is rdfs:label. Full list in Appendix A.

## 4.4 Labelling Properties

Using one labelling property across the whole web of data facilitates the reuse of the data as the human-readable labels of an entity are clearly and unambiguously identifiable. While in our further analysis we use only the labelling property `rdfs:label`, because it is the unified standard to label entities[2], we want to understand the complexity of the labelling on the web of data overall by extracting the label properties and their usage across the graphs.

We analyse the dataset by selecting all properties that connect at a string value with a language code. This totals 671, 768 properties. These properties are used between 246, 201, 989 and 1 times across the dataset, with a mean/median of 1898.66/6. As properties can have a string value without labelling the property, we need to manually select the properties that can be considered labelling properties. First, we limit the properties by selecting the ones that are used over 100 times across the dataset. Given the large size of the dataset, we consider labelling properties used multiple times as more valuable. There are 448 properties that are used over 100 times and that match our criteria. After manually investigating these, we select all properties that are used for labelling entities. Other examples of properties with string values and an associated language code are, for example, descriptions (such as `http://eunis.eea.europa.eu/rdf/schema.rdf#description`), aliases (such as `http://rdf.freebase.com/ns/biology.organism_classification.synonym_scientific_name`), and other string values, such as usernames (e.g., `http://rdf.freebase.com/ns/base.myspace.myspace_user.username`).

Manual processing of the data results in a total of 78 labelling properties. The manually selected labelling properties are used between 246, 201, 989 and 110 times for labelling across the dataset. On average, they are used 3, 903, 026.64 times (median 1, 151.5).

All labelling properties and their frequency of use across the dataset are displayed in Table 4.1. The labelling property `rdfs:label` is used by far the most across the dataset (246, 201, 989 times). Compared to `foaf:name`, the second-most-used labelling property (used 16, 590, 050 times), it is used 1, 384% more.

[2] `https://www.w3.org/2004/12/q/doc/rdf-labels.html`, retrieved 25. September 2020

| Metric | % labelled |
|---|---:|
| Entity labelling | 5.4% |
| Class labelling | 37.6% |
| Property labelling | 80.9% |

TABLE 4.2: Results for the entity, class and property labelling.

## 4.5 Entity Labelling

Any dataset on the web of data should aim to have each entity labelled in at least one language. Since labels determine how well humans can access the knowledge stored in it, it is important to thoroughly label the dataset. We consider all subjects in the dataset in this metric, i.e., all entities that have been used as a subject in any triple at least once. In the web of data, only 5.42% of entities are labelled. This number is extremely small, and might be due to the size and heterogeneity of the datasets included in the LOD Laundromat dataset.

## 4.6 Class Labelling

Classes are made accessible by being of type class. In a triple, the property `<http://www.w3.org/1999/02/22-rdf-syntax-nstype>`, i.e., `rdf:type`, has the object `<http://www.w3.org/2000/01/rdf-schemaClass>`. We extract all entities, which are of type `rdf:Class` from the dataset, and compare them with all labelled entities. Classes are highly reused across the dataset, therefore their labelling is a high priority. In the LOD Laundromat dataset, classes are used as objects 25,143.4 times per class on average (3.0 median).

Of $202,925$ classes we identify in this way, 37.6% are labelled.

## 4.7 Property Labelling

We identified properties by considering all predicates as properties. In other words, all entities used as a predicate in a triple are considered properties. Of the $1,165,269$ properties we identified this way, 80.9% are labelled.

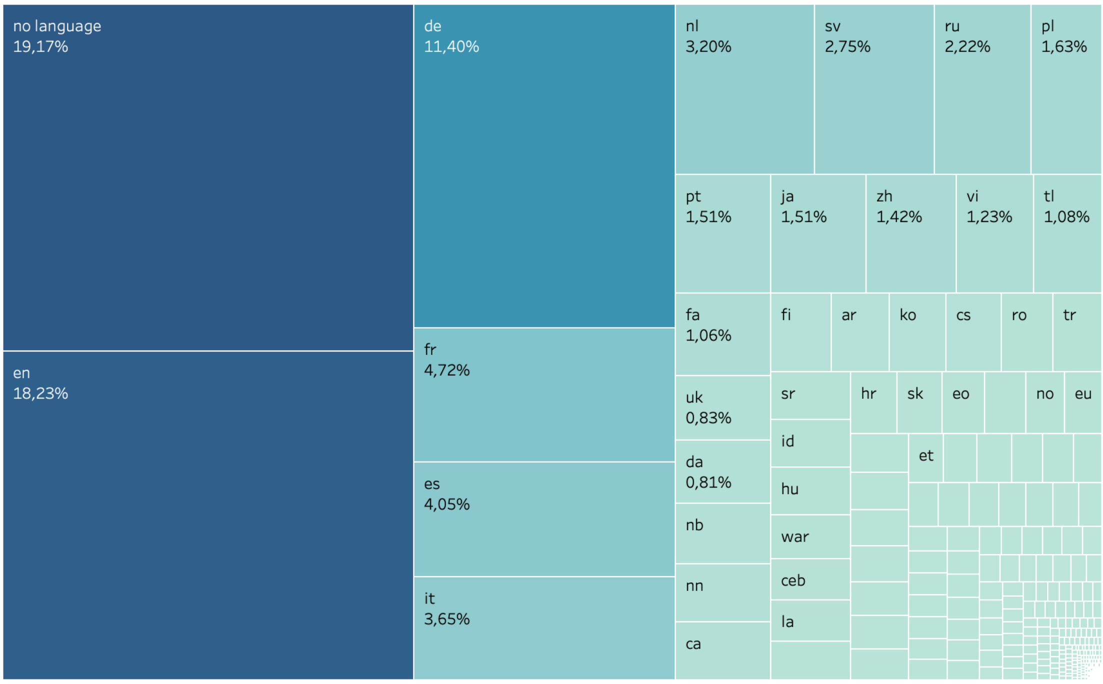


FIGURE 4.1: Language distribution in the LOD laundromat dataset, including labels that do not have a language code (no language). English is the most prominent language on the web of data, followed by German (de), French (fr), and Italian (it).

## 4.8 Unambiguity

Ambiguity complicates the task of selecting the appropriate label for an entity. We focus the measuring of unambiguity on the largest language, i.e., English (see Section 4.9).

Of the $51,479,960$ entities labelled in English using `rdfs:label`, $1,820,686$ are labelled with more than one English label (3.5%). 1.5% of entities have more than two English labels with the same labelling property.

Given all labelling properties extracted in Section 4.4, 22.4% of the $69,086,756$ English-labelled entities are labelled with more than one label. 4.6% of entities are labelled with more than two labels.

## 4.9 Multilinguality

We measure the number and distribution of languages on the web of data to understand how widely languages are covered and to what degree the LOD laundromat dataset is accessible to different language speakers. In this analysis, we focus on labels which are identified by the `rdfs:label` property. We detect the language of the label by processing the language code of a label, which is given in the form `‘‘label’’@en` for English.

19.2% of the labels do not contain a language code. Therefore, the language of the label cannot be identified. This hinders reuse, as they cannot then be used for language-specific applications. The dataset covers 467 languages in total. The largest proportion of labels are in English (18.2%), followed by German (de), French (fr), and Spanish (es). Figure 4.1 depicts the language distribution on the web of data, showing the extensive coverage achieved in a few prominent languages, and the relative scarcity of labelling in other languages. The top five languages cover over 50% of the labels. The top 10 languages are English, German, French, Spanish, Italian, Dutch, Swedish, Russian, Polish, and Portuguese.

## 4.10 Monoglingual Islands

For the web of data to be truly multilingual, all concepts should be available across users’ various languages. But in the web of data, the vast majority (83.2%) of entities are currently labelled in only one language, as seen in Table 4.3. Given the large number of languages covered in the web of data, this indicates there are in fact monolingual islands, i.e., entities labelled in only one language without translations into other languages.

This distribution of language coverage should be improved by labelling entities across languages. We demonstrated in our previous study (Kaffee and Simperl, 2018a) that datasets such as BTC10 (Harth, 2010) and BTC14 (Käfer and Harth, 2014) (the billion triple challenge datasets of 2010 and 2014, each comparable in size to the LOD Laundromat) with a comparable size to the LOD Laundromat, score worse in terms of monolingual islands. In BTC10, 99% of entities are labelled in only one language. The more recent version of BTC, BTC14, saw a slight improvement, with 93% of entities labelled in only one language. The LOD Laundromat datasets has a better distribution of languages, but we show that community-contributed knowledge graphs such as Wikidata (see Chapter 5) still score higher.

| # languages | # entities | % |
|---|---:|---:|
| 1 | 105,875,515 | 83.2% |
| 2 | 6,616,125 | 5.2% |
| 2 -5 | 4,902,600 | 3.9% |
| 5-10 | 7,375,341 | 5.8% |
| $> 10$ | 2,474,601 | 1.9% |

TABLE 4.3: Share of entities having labels in multiple (1, 2, 2-5, 5-10, over 10) languages over $127, 244, 182$ entities.

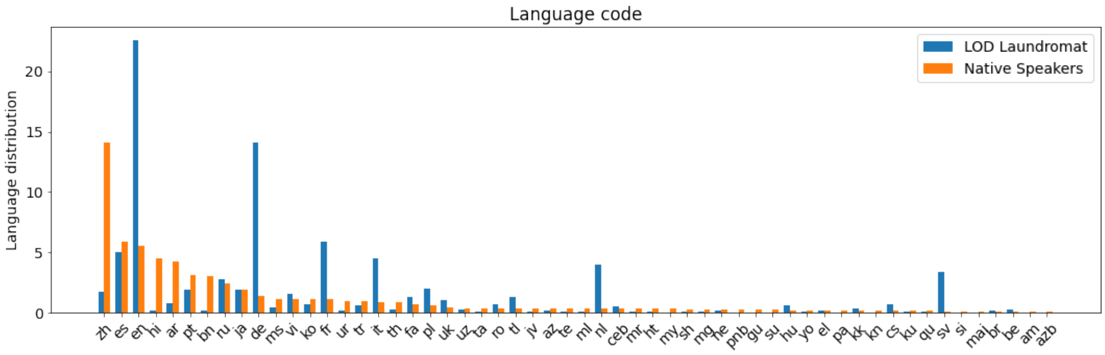


FIGURE 4.2: Comparison of language distribution of the LOD Laundromat and native speakers in the world.

## 4.11 Contextual Comparison

We compare the language distribution of the LOD Laundromat dataset with the numbers of native speakers in the world of each of the target languages. In Figure 4.2 we clearly see a disparity between the languages provided on the web of data and native speakers in the world. Bridging this gap in language information is important to ensure an equal access to information online for speakers of all languages. Dutch (nl) and Swedish (sv) have comparatively high coverage in terms of labels. One possible explanation is that Wikidata is included in the LOD Laundromat because it is part of the web of data, and we can observe a similar distribution for these languages in Wikidata (see Section 5.12).

## 4.12 Discussion

We analysed the LOD Laundromat based on a framework that combines different metrics to assess language and label coverage of the web of data. We find a severe lack of labelling of entities, and a maldistribution of languages compared with numbers of native speakers of each language. Following the results of the study presented here and the one we conducted in Kaffee and Simperl (2018a) on a smaller scale, we draw recommendations for data publishers.

**Datasets should be thoroughly labelled.** Given the current lack of labels overall, data publishers need to prioritise the labelling of concepts. Labels are the human-accessible part of the web of data, and need to be present for applications to display information to a human user. Further, to increase language coverage in approaches such as the one proposed in Chapter 7, a label in a source language such as English needs to be provided.

**Labelling properties should be coherent and limited in number.** A limited number of labelling properties makes it easier to differentiate the preferred label for an entity. Even if the property is not standardised, this reduces ambiguity. The frequent use of the standard labelling property `rdfs:label` is promising. As the LOD Laundromat dataset is an aggregation of different knowledge graphs, we find a large number of different labelling properties. This aligns with the findings of Kaffee and Simperl (2018a) with regard to BTC, a large dataset with a high number of labelling properties, compared to smaller datasets with fewer labelling properties.

**All entities should be labelled in multiple languages.** Multilinguality allows different communities to access the same datasets. We find that having more languages in the dataset does not necessarily mean better coverage. While there is a high number of languages covered in the web of data overall, there is a lack of translations between languages, i.e., most entities are only covered in one language. Translation of existing English labels could be a way to improve coverage of existing labels. As manual translation is cost intensive, we propose a method of increasing language coverage using transliteration and translation in Chapter 7.

# Chapter 5

# Multlinguality in Wikidata

We introduced a framework to measure languages and labels in Chapter 3. In the previous chapter, Chapter 4, we described the coverage of languages and labels in the web of data based on the LOD Laundromat dataset. In this chapter, we apply the same framework to Wikidata, the community-driven knowledge graph created as a central knowledge store for Wikimedia projects such as Wikipedia, and now widely used in third-party applications as well. In Wikidata, the community contributes to every part of the data, including natural language labels. Labels of items can be imported from Wikipedia or added via support tools. Examples of such tools include bots - user-written scripts that usually perform repetitive tasks - or the *Wikidata Terminator*[1], which encourages users to translate the most frequently used items. Wikidata maintains the links between different language versions of Wikipedia and other Wikimedia projects, which means that many items are connected to a given Wikipedia article. Titles of connected articles are often imported as labels for the respective Wikidata items. Wherever Wikidata's data is used in Wikipedia, it displays the label of the entity. One example is infoboxes, summaries of information in articles in Wikipedia[2]. Infoboxes reuse the data of Wikidata. Another example of Wikidata language information being used in Wikipedia is ArticlePlaceholders (Kaffee, 2016), which generate an overview of a topic with data provided by Wikidata. For more on these, see Chapter 8. There is clearly, therefore, a strong interest in improving the coverage of languages in Wikidata, given its impact on, among other resources, Wikipedia. Further, understanding language and label coverage in Wikidata is crucial to identifying weaknesses, which can then be remedied. For example, the lack of language coverage previously identified in the Multilinguality metric can be addressed by translation, as described in Chapter 7.

In contrast to the previous chapter, we apply the whole framework, including the metrics in the data creators' dimension, in this analysis. Wikidata is edited and

[1] https://tools.wmflabs.org/wikidata-terminator/

[2] https://www.wikidata.org/wiki/Wikidata:Infobox_Tutorial, retrieved 04.02.2020.

maintained by a community of editors, including registered users, anonymous users, and bots. We define these user groups in Section 5.1. Understanding editors' behaviour in terms of label editing is crucial in two aspects: it gives an insight into the provenance of the multilingual data; and it can support future work on how to support editors in creating more language information.

## 5.1 Editors in Wikidata

The community of Wikidata consists of humans and bots working alongside each other. This community can work to close the language gap, given the right tools. To understand the provenance of the current label data, we analyse the different editor groups and how they contribute to the distribution of languages within labels. This gives an insight how much the community contributes to the language distribution of the knowledge graph, and it also supports the development of applications to support editors in their editing of labels in under-served languages in the future.

There are different actors contributing to the content of the knowledge graph. We define three groups of editors, analogously to Steiner (2014):

1. *Registered users*: Editors with an account and a user name. We treat each user name as a different user.
2. *Anonymous users*: Anonymous users edit without a user account. Instead of a user name, their IP address is recorded. We treat each IP address as one user.
3. *Bots*: Bots are automated tools that typically work on repeated tasks.

We focus on a comparison of these three different types of editors in the metrics regarding data creators. Understanding how different user groups shape the knowledge graph can facilitate the creation of applications for these user groups, and thus tackle the maldistribution of languages in the knowledge graph. We investigate the multilinguality of the three user groups (and, in particular, whether automated tools are more or less multilingual than humans); which group is the most active in label editing; and what kind of patterns can be seen in their edit activity over time. We hypothesise that human editors tend to edit in different languages on the same items, i.e., translating labels of one concept; while bots edit different entities in the same language, i.e., importing labels in the same language for a variety of concepts. This would align with the assumption that, for a bot, a repetitive task (such as importing labels in one language) is easier than a complex task (such as the translation of labels into different languages within the context of one item's information). If this assumption holds, bots could be used in future to import labels created automatically, with approaches such as the one described in Chapter 7, where labels are automatically translated. We focus on two editing patterns: (1) a high number of different entities edited and a low number of languages, i.e., monolingual editing over different topics; and (2) a low number of different entities and a high number of languages, i.e., translation of labels. Demonstrating that bots do not currently translate labels can help us show the gap in research work on automatic translation, which we address in Chapter 7. Further, we want to understand the connection between languages that editors contribute to.

Finally, we investigate the connection between multilinguality and number of edits. The hypothesis is that the higher the edit count of an editor, the higher the number of distinct languages. This assumption follows the work of Hale (2014), who concludes that multilingual editors are more active than their monolingual counterparts on Wikipedia. Here, we test whether this holds also for Wikidata editors.

We split the dataset into three parts based on user type: registered users that edit with a username; anonymous users that edit without a username; and bots, automated tools marked with a bot flag or the *bot* prefix or suffix. In total, we considered $64,836,276$ edits to labels. Out of all $3,093,684$ registered users[3], 62,091 users edited labels. This group of editors is responsible for 46.5% of all label edits. The largest group of editors are anonymous editors – a total of $219,127$ unique IP addresses edited Wikidata's labels. However, they contributed to only 0.7% of the label edits. From all bots currently registered with a bot flag[4] and all bots marked with a bot pre- or suffix, 187 bots edited labels. Bots have the highest share of label edits – 52.8% of edits are made by bots.

## 5.2 Datasets

### 5.2.1 Wikidata's Entities

For the first set of metrics, the *dataset description* and *core metrics*, we analyse a database dump of Wikidata in turtle format (NT) from March 2017.

### 5.2.2 Editor Data

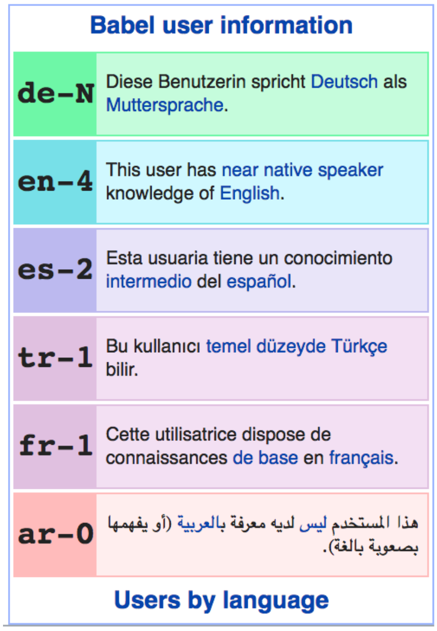
Every registered user of Wikidata can change the language of the interface. Not only interface elements are switched to their preferred language, but also all data displayed. This setting is called *User Language Setting*[5]. We extracted the aggregated

[3] Statistics on users, retrieved March 2019: https://www.wikidata.org/wiki/Special:Statistics

[4] https://www.wikidata.org/wiki/Wikidata:List_of_bots

[5] https://www.wikidata.org/wiki/Help:Navigating_Wikidata/User_Options#Language_settings

number of user's languages via Wikimedia's Grafana installation[6], as of January 2018. The data can be downloaded as JSON. We use the user language setting as a starting point for our work to understand the language distribution of Wikidata users. However, there are certain limitations to this approach: English is the default language. That means that, even if a user would be more comfortable in another language but understands English reasonably well, we can assume they keep their interface in English. Furthermore, we cannot identify their editing language(s) because this is a setting that indicates the reading of Wikidata. And neither does it give any indication on the possible multilinguality of users.



Babel user information

| | |
|---|---|
| de-N | Diese Benutzerin spricht Deutsch als Muttersprache. |
| en-4 | This user has near native speaker knowledge of English. |
| es-2 | Esta usuaria tiene un conocimiento intermedio del español. |
| tr-1 | Bu kullanıcı temel düzeyde Türkçe bilir. |
| fr-1 | Cette utilisatrice dispose de connaissances de base en français. |
| ar-0 | هذا المستخدم ليس لديه معرفة بالعربية (أو يفهمها بصعوبة بالغة). |

Users by language

FIGURE 5.1: Example for a BabelBox of User:Frimelle

**BabelBox** Each registered user in Wikidata has a user page where they can add and edit content about themselves. One of the templates that can be added indicates languages spoken: *BabelBox*[7]. BabelBox lets a user self-assess their range of languages and the respective levels of these languages from 0 (no knowledge) to N (native understanding) as follows:

N Native understanding

5 Professional knowledge

4 Near native speaker knowledge

3 Advanced knowledge

[6] Active User Language on Grafana: https://grafana.wikimedia.org/dashboard/db/wikidata-site-stats?orgId=1

[7] https://en.wikipedia.org/wiki/Template:Babel

2 Intermediate knowledge

1 Basic knowledge

0 No knowledge

When we talk in the following of a *known language*, we usually refer to all levels excluding level 0 and 1. Accordingly, *unknown languages* are defined as all languages of level 0 or 1, and all languages that are undeclared, as we assume that the user has no knowledge of these languages. In order to access the user pages, we download the complete Wikidata dump[8] and extract user pages with BabelBoxes. Since Wikimedia projects are connected, users can also have a so-called global user page[9], which we take into account. We process all user pages with a BabelBox in order to collect data. In total, 4,120 users have a BabelBox enabled on Wikidata or Meta as of 2018. Meta is a Wikimedia website established for tasks *"from coordination and documentation to planning and analysis"*[10]. Global user pages on Meta enable a user to create their user page (where the BabelBox is stored) which is displayed across all Wikimedia websites (including Wikipedia, Wikidata and other Wikimedia projects).[11] Including the global user pages on Meta for active Wikidata users is important because it enables us to investigate the BabelBoxes of users who do not create an additional, local Wikidata user page. Wikidata has 19,333 active users (out of 2,930,072 total registered users). For our exploration of multilinguality in Wikidata users and their editing, we treat those users as a sample of all users. However, there are clear limitations to this approach - our sample is not truly random. Only the data of users who had enabled BabelBox could be captured for this study, which may have led to demographic bias. The close connection between Wikipedia and Wikidata users (Piscopo et al., 2017b) lets us assume that users of certain Wikimedia project sites, such as German Wikipedia, are more likely to enable the BabelBox[12]. Furthermore, other factors are likely to influence our results: multilingual users might be more likely to enable a BabelBox than monolingual users. Therefore, we extend our study to include the full edit history of Wikidata to include all users of Wikidata and their editing of labels.

**Edit History** Wikidata provides whole dumps of its current data as well as the entire editing history of the project. We worked with a database dump of Wikidata's history, as of 2019-03-01. The data is provided in XML, and we converted the data to a PostgreSQL database. The database fields resemble the fields of the XML structure. We extracted only label edits, and no other kind of edit, by filtering on the *wbsetlabel-set* or *wbsetlabel-add* tag in the edit comment. The history dump includes all

[8] https://dumps.wikimedia.org/wikidatawiki/, as of January 2018.

[9] https://meta.wikimedia.org/wiki/Global_user_pages

[10] Description of Meta from https://meta.wikimedia.org/wiki/Main_Page, retrieved 13. May 2021

[11] https://meta.wikimedia.org/wiki/Global_user_pages, retrieved 13. May 2021

[12] This could be an explanation, e.g., for the high number of native German users.

information from 2012-10-29 to 2019-03-01. We split the database into three tables (one for each of the user types): registered, anonymous, and bots. We define an edit as any alteration of a label; creation and updating of a label are treated as the same. This limits the amount of investigation we can do on, e.g., how long a label is kept in the graph. It also treats vandalism as a regular label edit, as we do not consider how long a label is in the graph until another user changes it. It does give an insight into the overall language edits, however. We leave investigation of vandalism in labels, based on the work of vandalism detection in Wikidata (Heindorf et al., 2016) and work on quick changes of property labels (Tanon and Kaffee, 2018), to future research efforts. In the following, we use the term *edit* only for edits to labels unless specified otherwise.

**Users** We split the users into three groups: registered, anonymous, and bot editors. Bots on Wikidata are created by community members to import or edit data in an automated, repetitive manner. To ensure that their editing follows the standards of the knowledge graph, bots need community approval. Each bot has a unique username and is flagged as a bot. We use the list of bots that have a bot flag on Wikidata[13]. Since historical bots might not currently have a bot flag, we add to the list of bots all users that have a *bot* pre- or suffix, as this is how bots are supposed to be named. Registered users are all users that have a username and do not have a bot flag (and are not otherwise marked as bots). Anonymous users do not have a username but an IP address, which we treat as a username. This has the disadvantage that we treat each IP address as a single user, not knowing whether the IP address is used by several users. However, this gives us an insight of anonymous users at large, as we can observe their editing patterns in comparison to the other user types.

## 5.3 Size

At the time of the investigation (in 2017), there are 26 million entities in Wikidata with 134 million labels, and 3,000 properties.

## 5.4 Natural Language URI

URIs in Wikidata are opaque. An example of URIs can be found in Figure 1.1. For example, the full URI for the entity *Ada Lovelace* is `https://www.wikidata.org/wiki/Q7259`. Entities use a URI starting with the letter `Q`, followed by a numeric ID, properties start with the letter `P`, followed by a numeric ID.

[13] List of bots with bot flag: `https://www.wikidata.org/wiki/Wikidata:Bots`

## 5.5 Labelling properties

Wikidata uses the standard labelling property `rdfs:label`. The knowledge graph does not allow for entering labelling properties - a user entering a label cannot chose which labelling properties should be used. Therefore, all labels are connected by the same labelling property.

## 5.6 Entity Labelling

Wikidata's coverage in entity labelling is complete, as no entity can be created without a label. Therefore, each entity has to have at least one label in any language. The distribution of languages in entities is described in the *multilinguality* metric.

## 5.7 Class Labelling

Wikidata does not have explicit classes. Piscopo and Simperl (2018) note that Wikidata does not distinguish between entities in classes in the knowledge graph in any formal way. As classes are the same as entities (see above), we can describe the classes of the knowledge graph Wikidata as fully labelled, too.

## 5.8 Property Labelling

Properties have a special position in Wikidata's ontology, as Wikidata does not have explicit classes (see Section 5.7). Properties, however, are easily distinguishable by their identifiers. Due to the high reuse and importance of properties in the knowledge graph, the community process to create a new property is more complex than that for creating a new item (Müller-Birn et al., 2015). As for any entity, properties have to have at least one label in one language at creation.

## 5.9 Unambiguity

Wikidata uses only one labelling property, `rdfs:label`. Further, the backend of Wikidata does not allow for storing more than one label per entity. Therefore, all entities in the knowledge graph are unambigiously labelled.

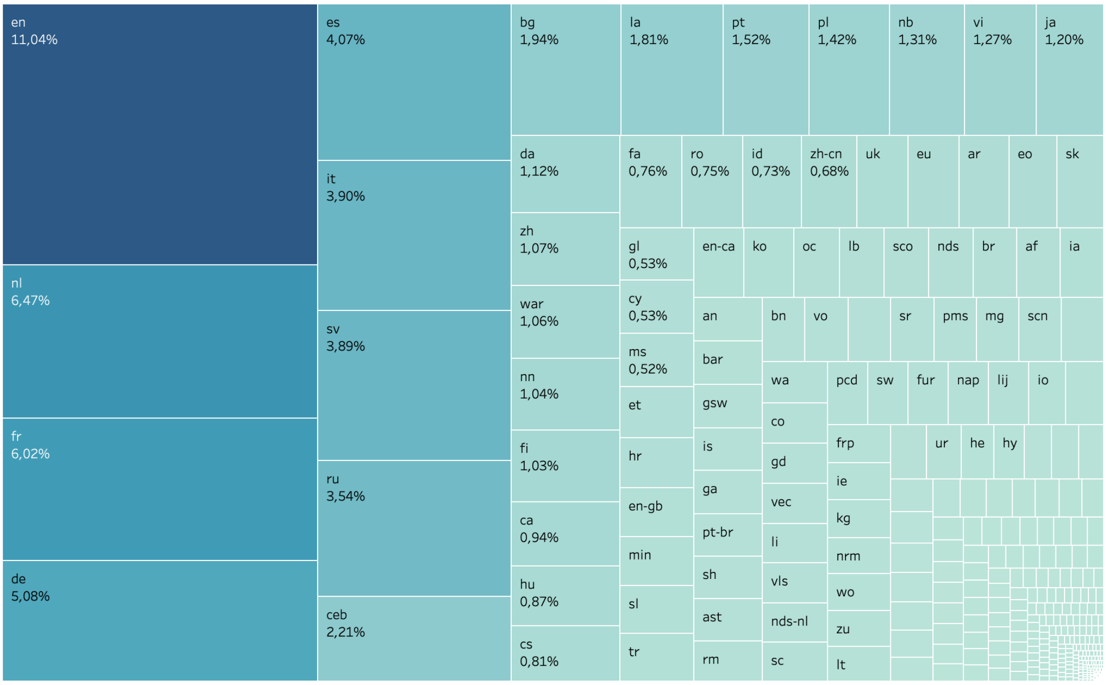


FIGURE 5.2: Percentage of all labels per language in Wikidata

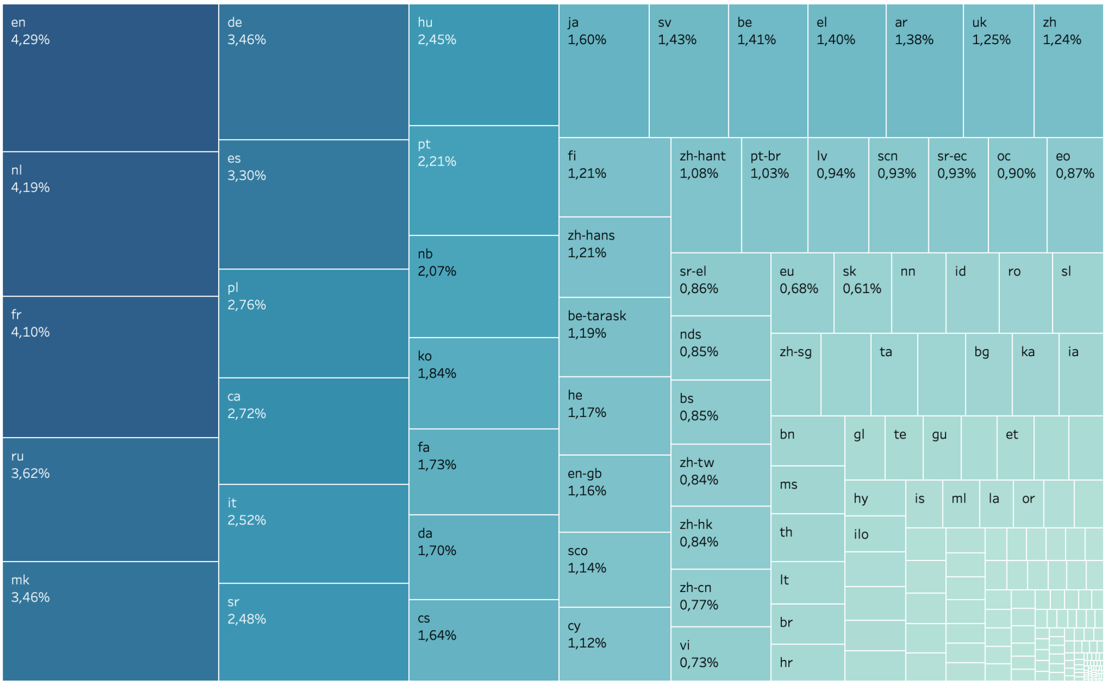


FIGURE 5.3: Distribution of languages for properties in Wikidata

## 5.10 Multilinguality

As an orientation, we look at the state of languages on the web at large. English is the language of around 51.9% of all websites[14] even though it is spoken by only 25% of

[14] https://w3techs.com/technologies/overview/content_language/all

the world's population. Chinese is the second largest language in terms of users on the web, but only 2% of the content on the web is in Chinese, according to the report published by Mozilla (2017). In Wikidata, 11 languages hold almost 50% of all language knowledge, as evident in Figure 5.2, which indicates a similar problem of language maldistribution. Editors can change this distribution by adding new labels; we therefore expect to see a change in this distribution over the coming years. Currently, however, most of the content is covered only in a small set of languages, while the majority of languages are covered by only a few labels in the knowledge graph. The language best covered is English (11.04%). However, the language coverage is not as extremely homogeneous in Wikidata as on the web in general.

**Properties** It is especially important that the properties are translated, as the properties are frequently used across the graph. At the time of the study (2017), there are only $3,386$ properties, while there are close to 26 million items. We showed in Tanon and Kaffee (2018) that the labelling of properties rarely changes over time, which is important to ensure access and reusability. For English labels, the current property label has been the label for 87% of that properties lifetime. The more diverse distribution for properties in Figure 5.3 compared to Figure 5.2 is promising, given their importance in the graph. Properties are used widely across Wikidata; therefore, it is more likely for missing translations to be detected by the community. Consequently, the distribution of languages is more balanced; while English is still the most-covered language, the margin is narrow. English has a share of 4.29% in the distribution of property labels in all of Wikidata's languages, followed by Dutch with 4.19%.

| # languages | % |
|---|---|
| 1 | 58% |
| 2 | 17% |
| 2 -5 | 27% |
| 5-10 | 9% |
| $>10$ | 8% |

TABLE 5.1: Share of entities having labels in one or multiple (2, 2-5, 5-10, over 10) languages.

## 5.11 Monolingual Islands

Monolingual islands describes the phenomenon of an entity or a group of entities being labelled in only one language, making it difficult to access the entity in other languages. In Wikidata, just over half of the entities are labelled in only one language; a large share of entities (27%) are available in 2 to 5 languages. The distribution can be found in Table 5.1.

## 5.12 Contextual Comparison

When comparing Wikidata's label distribution with the distribution of first language speakers in the world in Figure 5.4, we can see that there is still a large gap between language speakers and their information needs in Wikidata. Most notable here is the case of Chinese, which is the the most spoken native language in the world but is barely covered in Wikidata. `zh` is the language code combining multiple Chinese scripts (standard and simplified Chinese, see Chapter 7), which does not reflect the entire breadth of Chinese information on Wikidata. Further, Wikipedia and its sister projects, such as Wikidata, are blocked and censored in China in the time of writing this thesis (Bamman et al., 2012).[15] This leads to a majority of the edits in Chinese being made from outside China.[16] However, the biggest Chinese version, with the language code `zh`, is still very under-served, especially given the number of people speaking the language. Examples such as Dutch (`nl`) or Cebuano (`ceb`) show that it is not strictly necessary for a language to be spoken by many people to have good coverage in the knowledge graph.

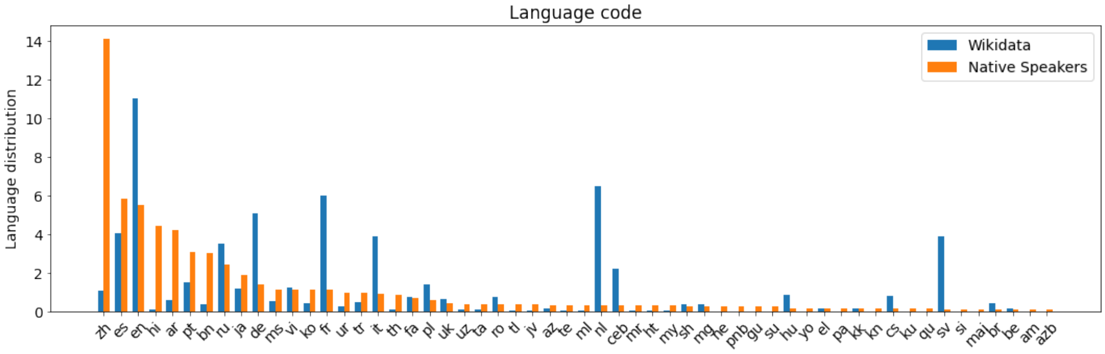


FIGURE 5.4: Comparison of distribution of languages in Wikidata and first language speakers in the world

Swedish (sv) and Cebuano (ceb) are especially interesting with regard to the relationship of Wikipedia and Wikidata. In Wikipedia, there is one contributor whose bot, called `lsvbot`, automatically adds stub articles[17] to Swedish and Cebuano Wikipedia. As article titles have been imported as Wikidata labels, we can assume the language coverage due to the high number of articles in Swedish and Cebuano, too. That there is a strong connection between Wikipedia and Wikidata is also visible in Table 5.2. It reflects the import of Wikipedia article titles as entity labels in Wikidata, as well as how the communities are intertwined, as also described by Piscopo et al. (2017b). The fact that many titles are imported can be seen also in the comparison to Wikidata property labels of Table 5.2. Swedish is ranked only 20th, while Cebuano

[15] Wikipedia page on the block of Wikipedia in China: `https://en.wikipedia.org/wiki/Censorship_of_Wikipedia#China`, retrieved 13. May 2021

[16] `https://en.wikipedia.org/wiki/Chinese_Wikipedia#Origin_of_edits`, retrieved 14. May 2021

[17] Stub articles explained on Wikipedia: `https://en.wikipedia.org/wiki/Wikipedia:Stub`

does not appear in the top 25 anymore at all. Since there are no Wikipedia articles linked to properties, these cannot be imported, and have to be translated by the community of either project on Wikidata. Dutch (nl) is a relatively well-covered language in Wikipedia, but has a higher rank for both Wikidata and Wikidata properties, indicating the high level of community involvement in Wikidata by Dutch speakers.

In the following, we will shed light on the origin of the language distribution, analysing Wikidata editors' languages based on the data creator metrics.

| Rank | Wikipedia | Wikidata | Wikidata properties |
|---|---|---|---|
| 1 | en | en | en |
| 2 | **ceb** | **nl** | **nl** |
| 3 | **sv** | fr | fr |
| 4 | de | de | ru |
| 5 | **nl** | es | mk |
| 6 | fr | it | de |
| 7 | ru | **sv** | es |
| 8 | it | ru | pl |
| 9 | es | **ceb** | ca |
| 10 | war | bg | it |
| 11 | pl | la | sr |
| 12 | vi | pt | hu |
| 13 | ja | pl | pt |
| 14 | pt | nb | nb |
| 15 | zh | vi | ko |
| 16 | uk | ja | fa |
| 17 | ca | da | da |
| 18 | fa | zh | cs |
| 19 | ar | war | ja |
| 20 | no | nn | **sv** |
| 21 | sh | fi | be |
| 22 | fi | ca | el |
| 23 | hu | hu | ar |
| 24 | id | cs | uk |
| 25 | cs | fa | zh |

TABLE 5.2: Ranking of number of Wikipedia articles by language, all labels in Wikidata, and labels for properties in Wikidata

## 5.13 User Language

The users observed through their usage of Babelboxes know a total 298 languages.

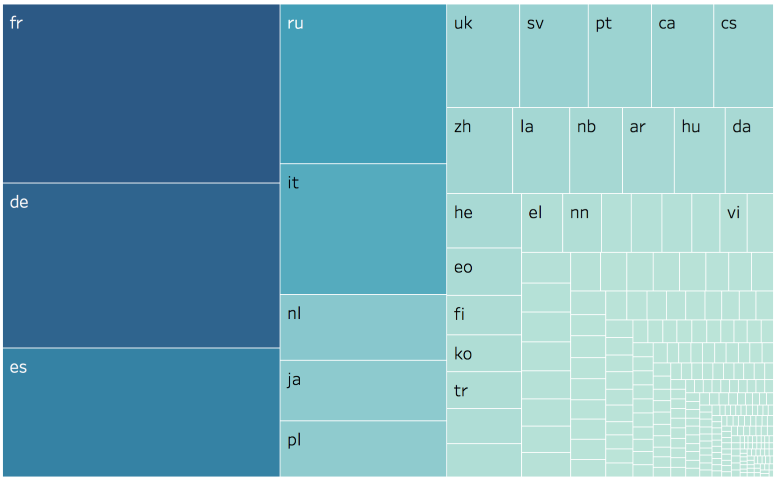


FIGURE 5.5: Language Distribution in User Language Setting, excluding English

**User Language Setting** The user language setting can be set by a registered user to change the language of the content displayed on the website. Since the default language is English, English is set by over 50% of users. Excluding English, as in Figure 5.5, we get a more interesting overview of the multilinguality of Wikidata users: The most prominent language is French, followed by German, Spanish and Russian.

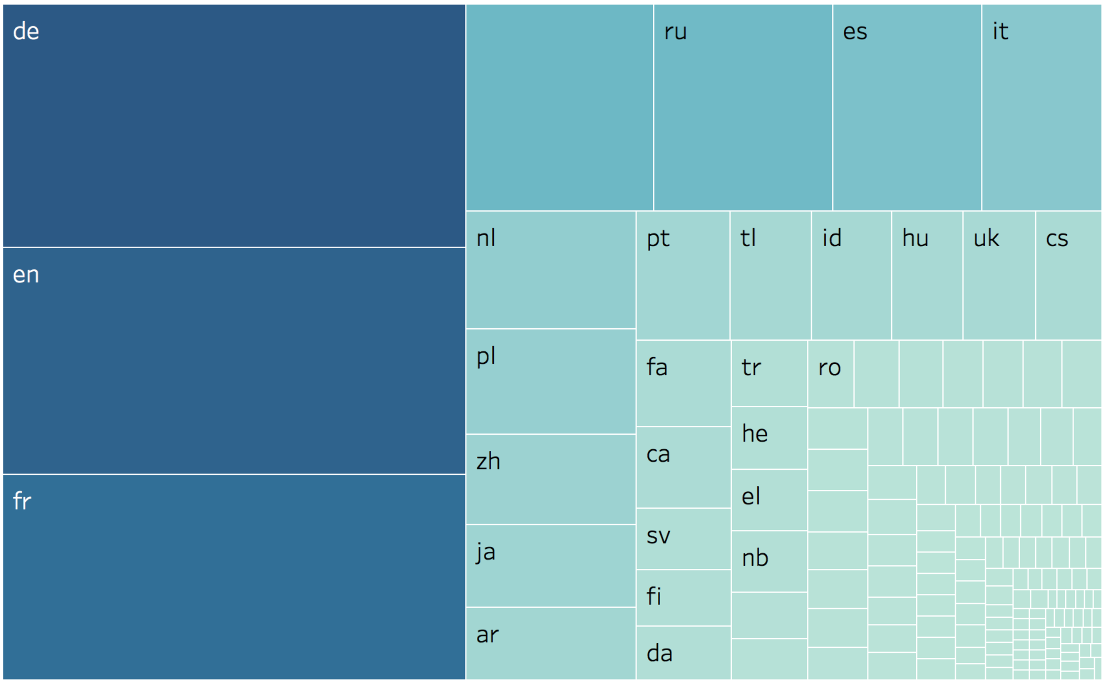


FIGURE 5.6: Language Distribution in BabelBox, Native Speakers

**BabelBox** BabelBoxes are used to indicate user languages on a user's page. Compared to users enabling the User Language Setting, BabelBox users are more active in editing labels. The mean of edits of BabelBox users is 55, compared to 2 for

| Description | % User | |
|---|---|---|
| Monolingual | 11.4% | 🟦 |
| $\leq$ 3 Languages | 58.3% | 🟪 |
| $\leq$ 5 Languages | 84.2% | 🟩 |
| $>$ 5 Languages | 15.7% | 🟦 |

TABLE 5.3: Distribution of multilingual users

non-BabelBox users with at least one edit (5440.16 vs 907.41 in average)). On Wikidata, 4120 users enabeled a BabelBox. Excluding all languages where users claim a knowledge level of 0 and 1 (one enthusiastic user declared over 161 languages for which they had a knowledge level of 0), we can observe a wide range of language knowledge.

As can bee seen in Table 5.3, most users are multilingual. Users know between one and, for one user, 47 languages. The high number of languages can be attributed to users adding languages they can interact with on any level - reading the script of that language, for example. In total, BabelBox users speak 298 different languages to at least level 2. The distribution of native speakers, visualised in Figure 5.6, shows an interesting image of the community: the majority of users speak languages that are also well represented in Wikidata labels (see Section 5.10). German is the most prominent language among native speakers using BabelBox. The three most prominent languages are English, German, and French, which corresponds to the language setting data. When we look at all the known languages of BabelBox users in Figure 5.7, English is by far the most widely spoken language. Since most of the community discussion on Wikidata is currently in English, this is to be expected.

## 5.14 User Language Editing

As we show in Section 5.10, Wikidata labels are not equally distributed between languages, with English being the most well-covered language. Based on the previous analyses, we can see that there is an overlap of the most prominent languages regarding labels and the languages spoken by the community – English, German, and French. Since we are analysing languages of users, we want to understand how and if the languages spoken by the community correlate with the labels available. The correlation coefficient is **0.8979**. Therefore, we find a strong correlation between the number of users speaking a language based on their BabelBox and the number of labels in that language. As users seem to edit in the language they use, i) outreach and growth of the community will also lead to more diversity in data; and ii) users are willing to contribute using all of their language knowledge, which can be helpful for future tools.

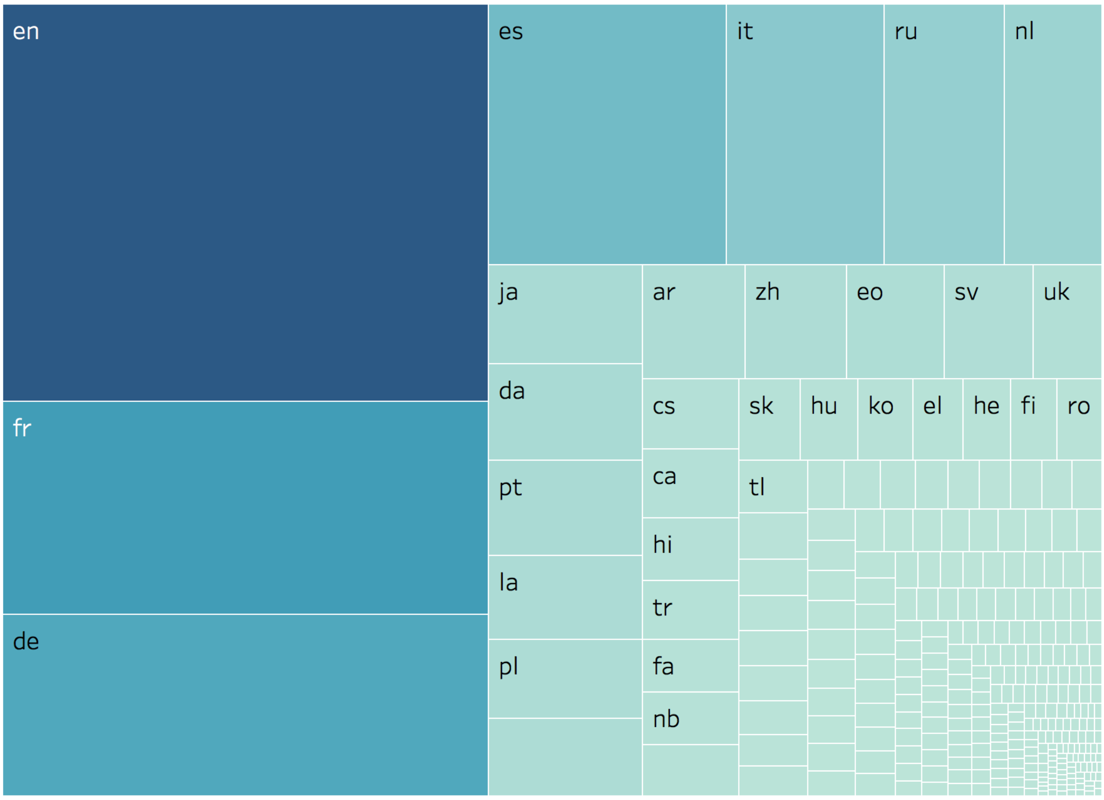


FIGURE 5.7: Language Distribution in BabelBox, excluding language levels 0 and 1

**BabelBox Language Edits** Based on our BabelBoxes results, we investigate the editing of labels on Wikidata by users. 1,107 of the BabelBox users do not have label edits in Wikidata. Therefore, we are working with 3,013 users. We plot the edits of users in Figures 5.8 and 5.9 to compare how much they edit in languages they know compared to edits in languages they do not know. In Figures 5.8 and 5.9, each user is a point. The x-axis represents the total number of edits in known languages, the y-axis the total number of edits in unknown languages. In Figure 5.9, we remove the outliers, however, still display most users (3,007 out of 3,013).

We find that the majority of users edits mainly in a language they reported to know. Over all users, edits to known languages are higher than to unknown languages. However, it is common to have at least a few edits in an unknown language. Especially users with high edits in labels are likely to contribute to languages they do not know. Furthermore, there are a few interesting outlier: editors who edit more in unknown languages than in their own languages. This indicates that editing language information in Wikidata is not very challenging - e.g., names in Latin languages can usually be transferred between different languages. Those cases should be investigated further, as such tasks can easily be automated once identified by repetition in all languages with the same script or by transliteration, as described in Chapter 7.

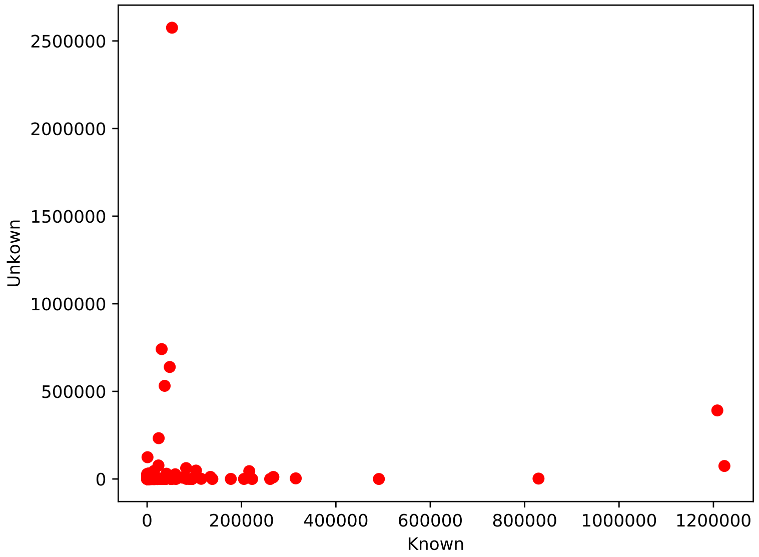


FIGURE 5.8: Users plotted according to their label edits in languages they are familiar with (known) and unknown languages

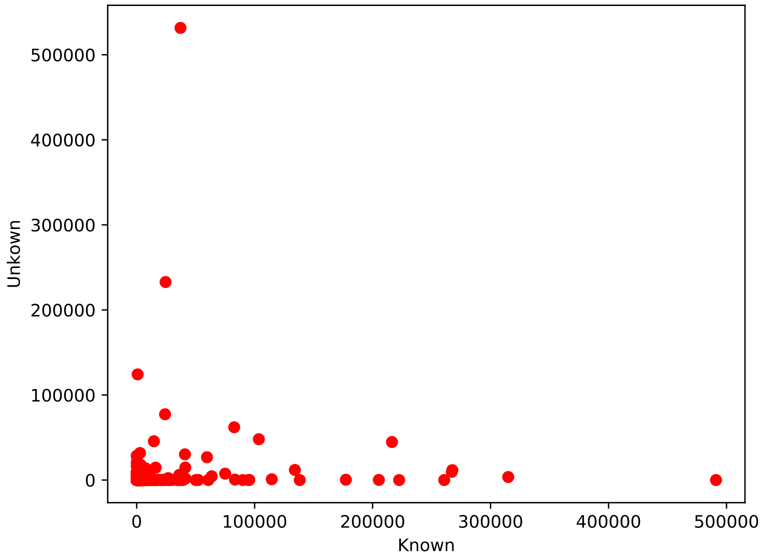


FIGURE 5.9: Users plotted according to their label edits in languages they are familiar with (known) and unknown languages, excluding most extreme outliers

## 5.15 User Activity

Looking at the average number of edits per editor in Table 5.4, we find that bots contribute a large number of edits, not only in total but also in the average per bot (183, 107.6). The most active bot (SuccuBot) made 14, 202, 481 total edits. While there are many anonymous users (219.127), they have a very low edit count per editor (2.1).

For the average number of language per editor, all editor types have a median of 1.0, showing that a majority of editors are monolingual over all three editor types.

| | Registered | Bots | Anon |
|---|---|---|---|
| # Editors | 62,091 | 187 | 219,127 |
| Avg Edits/Editor | 485.2 | 183,107.6 | 2.1 |
| Avg Language/Editor | 2.2 | 10.3 | 1.2 |
| Languages | 442 | 317 | 369 |
| Avg Editors/Language | 310.4 | 6.13 | 712.2 |

TABLE 5.4: Results of the analyses of Wikidata's editing history of 2019 for the user activity metric. The total number of editors is highest for anonymous editors; their average edit per editor is lowest however. Bots are the smallest group of editors, but have the highest number of average edit per editor.

| Bot name | Languages edited |
|---|---|
| KLBot2 | 247 |
| KrBot | 240 |
| QuickStatementsBot | 150 |
| Cewbot | 126 |
| Dexbot | 116 |

TABLE 5.5: Bots with the highest numbers of languages edited

However, on average, registered users and bots edit the widest range of language per user, indicating that a small number of editors contribute a large proportion of labels in Wikidata. In Wikipedia, Steiner (2014) found that bots are rarely multilingual, showing that only ten bots are active in more than five languages. In Wikidata, however, bots interact with multiple languages - up to 247 (see Table 5.5). In fact, only just over half of the bots (51.3%) are monolingual, which is an even lower proportion than among registered users (63.7%) and anonymous users (87.2%, explained by the low edit count per editor), see Figure 5.10. Even though registered editors edit fewer languages on average, the multilingual users edit up to 348 languages. Given the small number of edits per editor among the anonymous users, the low number of edits over languages in anonymous users is to be expected.

Figure 5.11 shows the timeline of user edits between the three user groups.

Figure 5.12 shows the ranking of languages by edit count and editor count. While the languages overlap neatly for anonymous users (Figure 5.12c), for the other groups there are strong differences. Given the low edit count by user for anonymous users, the alignment of edit count and editor count is evident. In the other groups, it indicates that more people can edit the language but are less active overall. In all graphs, English is leading for edit count and editor count, which aligns with the overall content in Wikidata.

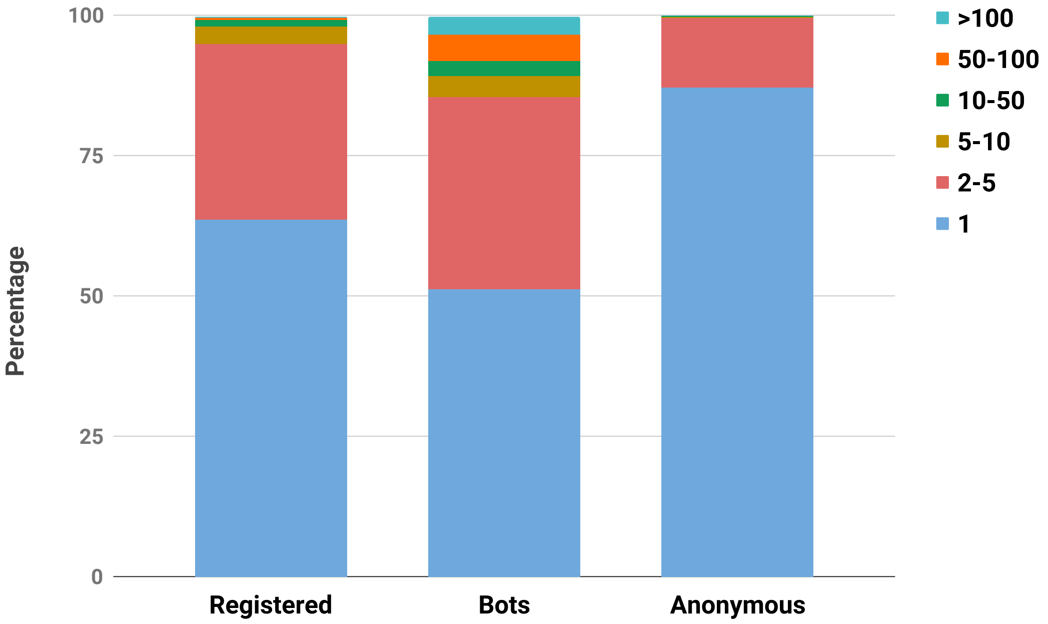


FIGURE 5.10: Measuring the distribution of multilingual editors: each editor type is represented by one bar, and split by the number of languages they edit. The majority of editors edit in one language.

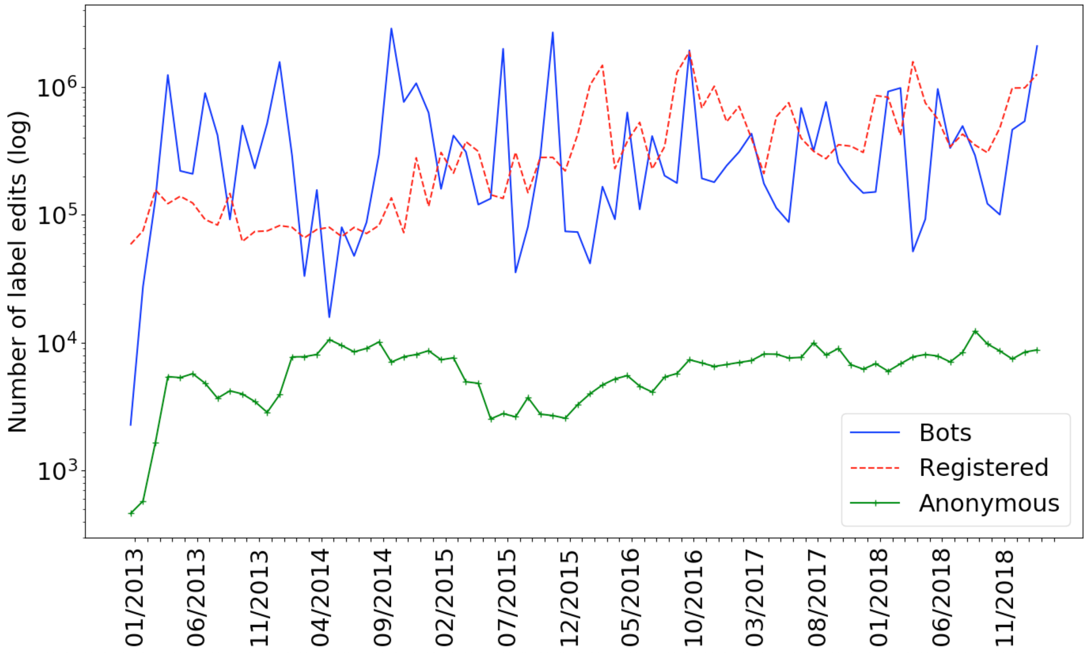
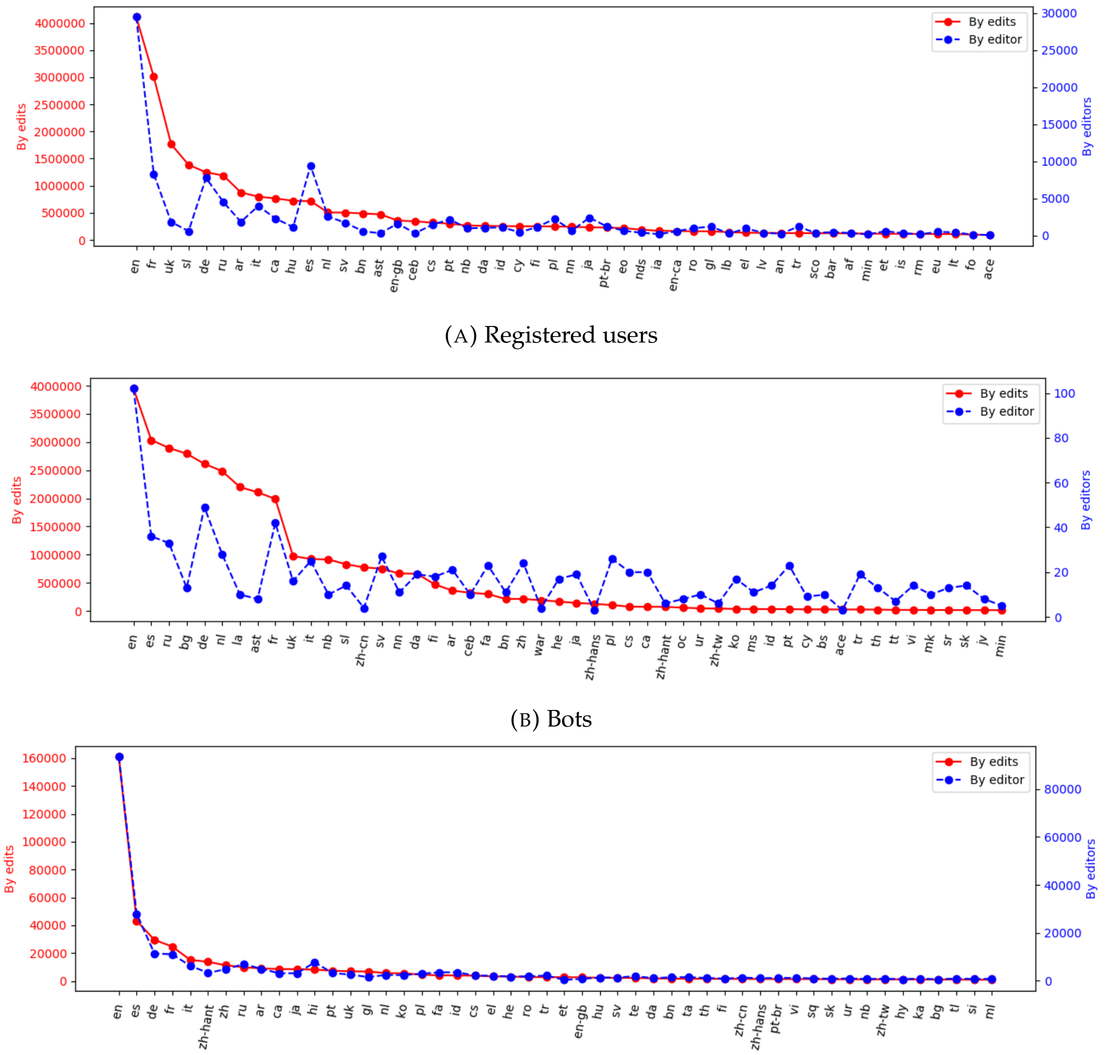


FIGURE 5.11: Timeline of number of edits (log) of the three different editor groups from January 2013 to March 2019. Edits are aggregated by month. The highest number of edits for registered users is in October 2016, for bots October 2014, and for anonymous users in September 2018.



(A) Registered users

(B) Bots

(C) Anonymous users

FIGURE 5.12: Language distribution over the three different editor groups, sorted by number of edits, including language ordering by number of editor in that language

## 5.16 Edit Patterns

We explore the different ways of editing over time between the three different groups. We hypothesise that human editors tend to edit in different languages on the same items, i.e., translating labels of one concept, while bots edit different entities in the same language, i.e., importing labels in the same language for a variety of concepts. We measure the changes to labels over time in *jumps*. The number of jumps is normalised over the total number of the edits. We limit this metric to experienced editors, here defined as editors with at least 500 edits over all time. The results for the normalised numbers of jumps between entities and languages can be found in Table 5.6. Generally, editors tend to switch more between entities than languages, i.e., there is less translation and more editing of labels in one language over multiple entities. However, there is a slight preference among registered editors to switch

| | Registered | Bots | Anon |
|---|---|---|---|
| Languages (Median) | 0.2 | 0.01 | 0.5 |
| Languages (Avg) | 0.3 | 0.1 | 0.4 |
| Entities (Median) | 0.9 | 1 | 0.8 |
| Entities (Avg) | 0.8 | 0.9 | 0.8 |

TABLE 5.6: Average number of *jumps* between languages and edits for all three user groups.

between languages compared to bots. Over all their edits, bots tend to edit in one language before switching to the next one.

## 5.17 Language Overlap

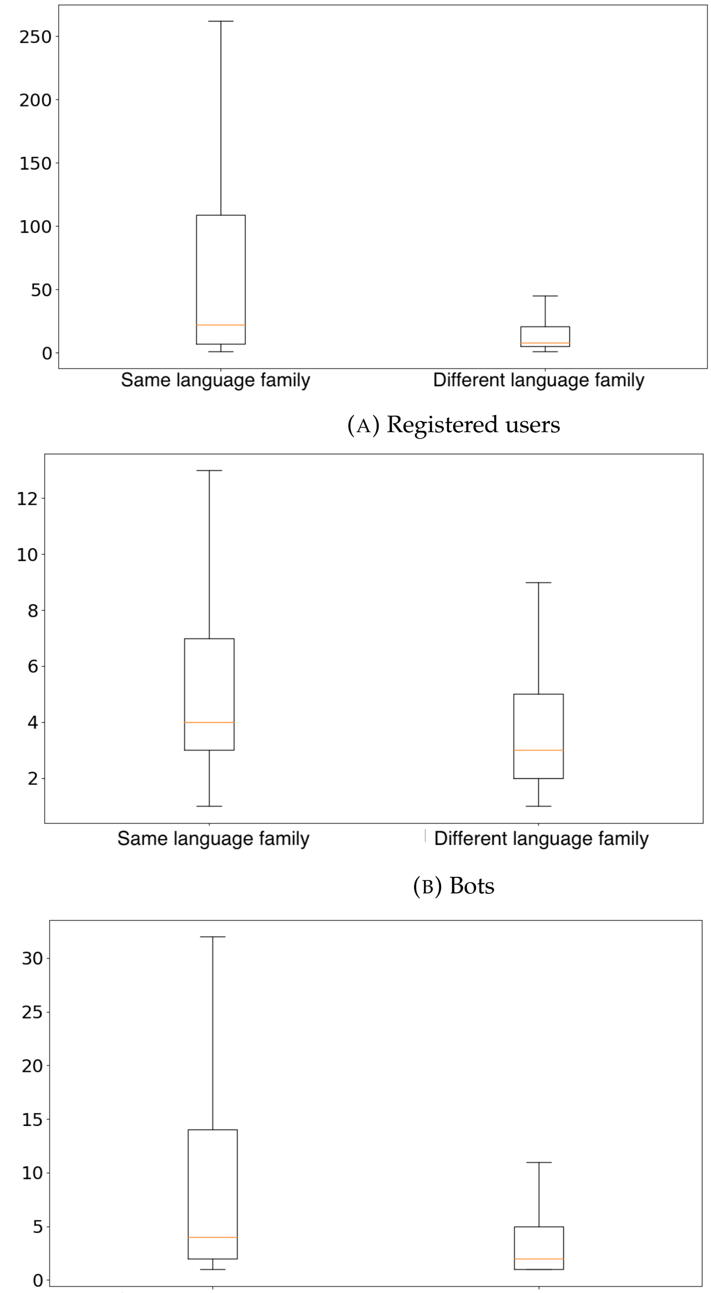
In this metric, we aim to gain an understanding of the languages editors contribute to, based on their editing behaviour. We create a language network graph where each node represents a language and the edge represents the cross-lingual edits by a single or more editors. In Figure 5.14 we visualise the language connections, limiting them to the ones that are higher than the average, following the work of Hale (2014). For registered users (Figure 5.14 (a)) we note a higher overlap of languages than for bots and anonymous users. While we showed in the previous section and Table 5.5 that bots edit a variety of languages, the low number of connections in the graph can be explained by the fact that those diverse editing patterns are rare and therefore do not pass the threshold for the weight. Anonymous users have mostly languages connected to only one other node, such as Vietnamese. These are usually connected to English.

Further, to understand the connection between languages that are edited together and language families[18], we counted the number of connections that are in the same language families and compared them to connections in other language families. Figure 5.13 shows the number of connections for each user group. Even though there is a tendency towards edits in the same language family for all user groups, overall there is no clear connection between language families and editors editing those languages together.

## 5.18 Activity and Multilinguality

We tested the hypothesis that multilingual editors are more active than their monolingual counterparts. First, we looked into the percentage of multilingual users,

[18] Language families based on: https://github.com/haliaeetus/iso-639/blob/master/data/iso_639-1.json

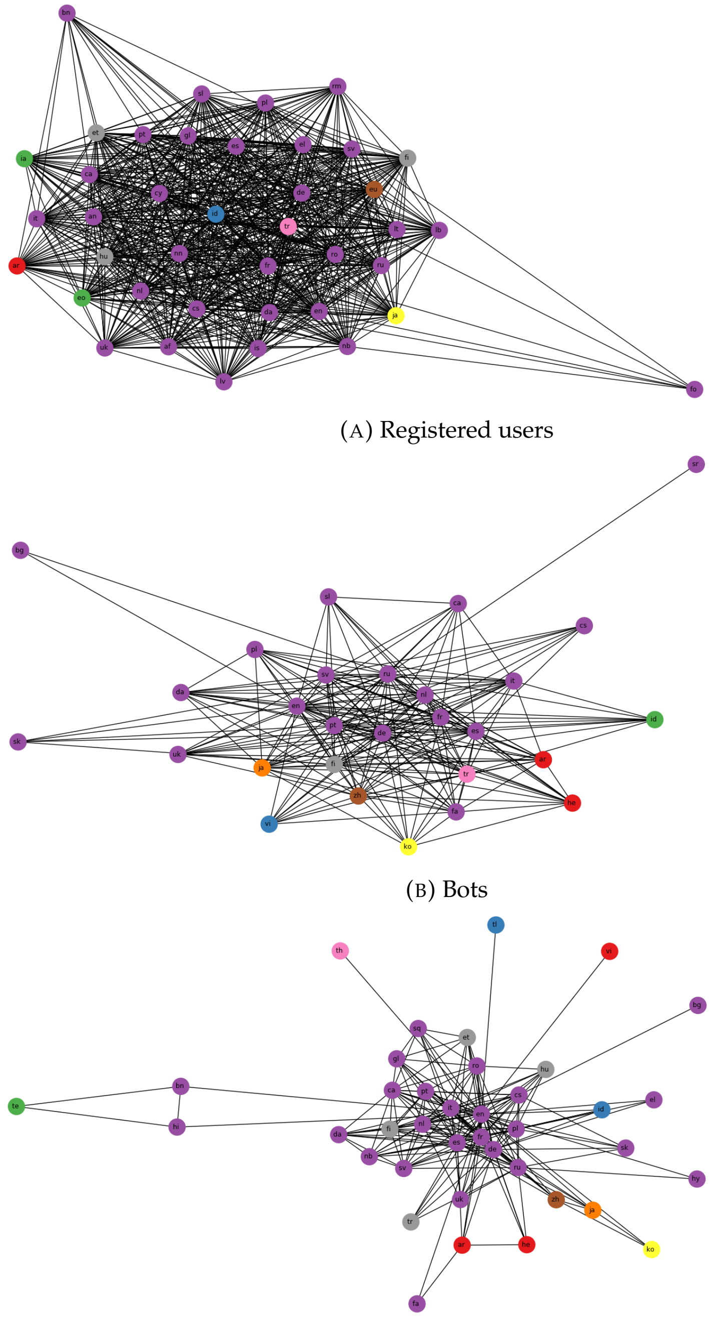


(A) Registered users

(B) Bots

(C) Anonymous users

FIGURE 5.13: Boxplot comparing the number of edits in languages of the same and different language families.



(A) Registered users

(B) Bots

(C) Anonymous users

FIGURE 5.14: Displaying the connections between languages, where the number of connections is greater than the average. Nodes are colored by language family.

as shown in Figure 5.10. The majority of users edit in only one language, even though a single edit on a label in a different language would make them *multilingual* in this graph. Figure 5.15 shows the number of edits (y-axis) and the number of languages edited in by the editor (x-axis). There is no clear correlation between the number of languages and the number of label edits, as can be seen in the figure. We measured Pearson's r to test the correlation between number of edits and number of languages edited. We used a two-tailed test. As shown in the previous figure, none of the user groups show a correlation between number of edits and languages (registered editors: (0.21, 0.0), bots: (0.24, 0.001), anonymous: (0.31, 0.0)). Activity level overall is therefore not an indicator for an individual's likelihood to edit across languages in Wikidata.

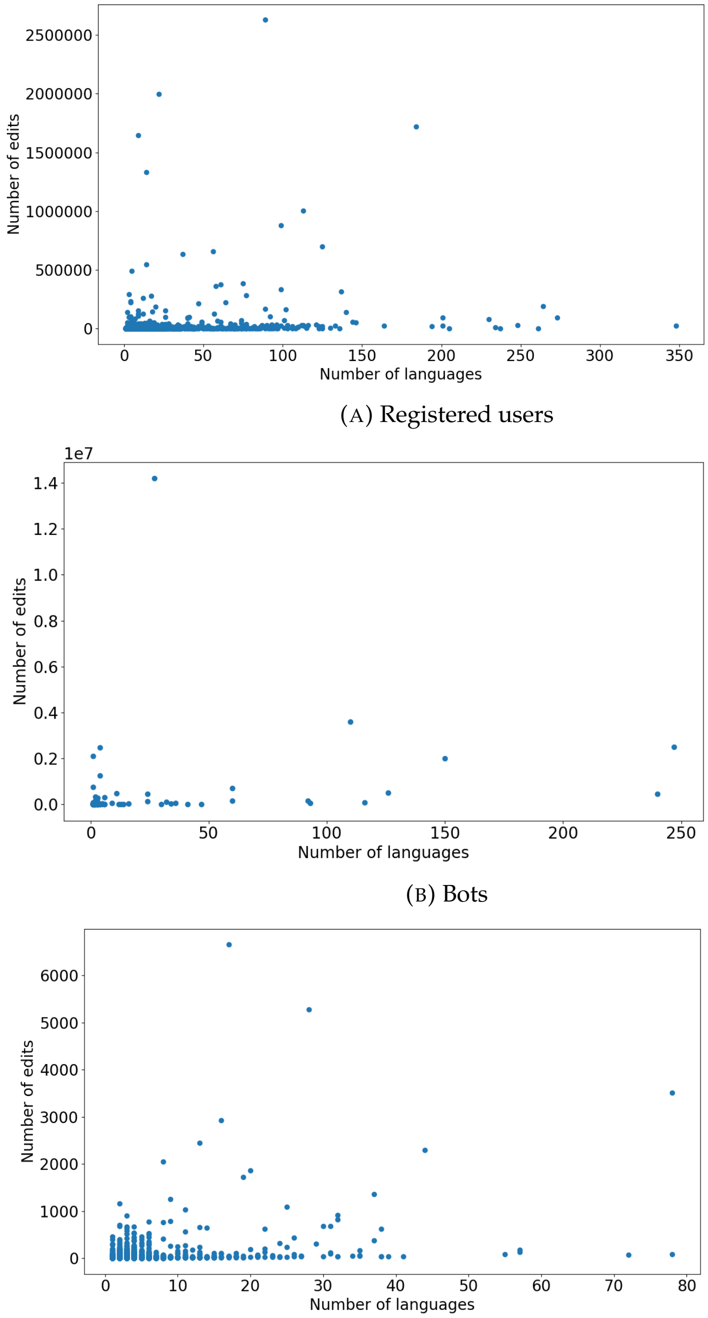
## 5.19 Comparison by User Group

Following the analysis of edits of the three user groups on Wikidata (registered editors, bots, and anonymous editors), we summarise our findings for each of the user groups below.

**Registered Editors** Registered users form the middle ground between bots and anonymous users in a number of ways; there are fewer of them than anonymous users, but they have a higher count of edits per editor. While they do edit between languages, they edit fewer languages per editor on average than bots. However, they show a much higher connection between languages than either of the other user groups. While they are likely to edit different entities with each edit, they have a higher count of translation (editing different languages after another) than bots. Based on this knowledge, we propose an automated approach to translating and transliterating Wikidata labels in Chapter 7.

**Bots** Bots, automated tools on Wikidata, have by far the highest edit count and contribute the most label data even though they are much fewer in number than registered or anonymous users. A few bots edit a lot of languages, but overall they are not as multilingual as their human counterparts. Compared to bots on Wikipedia, however, they reach much higher counts of languages edited. They are less likely to switch between languages; rather, they tend to edit in one language after another.

**Anonymous Editors** Anonymous users are the largest in number but make the lowest contribution to label edits. Their low number of edits makes it difficult to compare them to the previous groups. However, we note that there is a high degree of cross-lingual activity among anonymous users relative to their low number of edits.



(A) Registered users

(B) Bots

(C) Anonymous users

FIGURE 5.15: Scatter plot of the number of languages and the number of edits, testing correlation for all users.

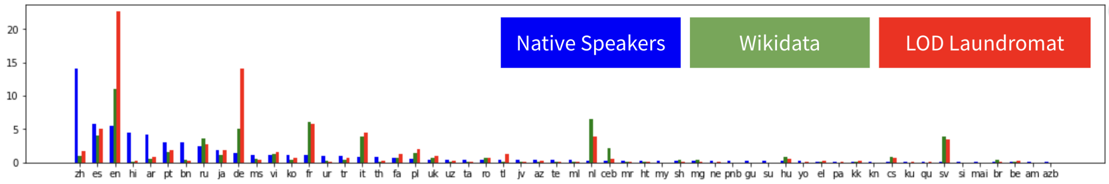


FIGURE 5.16: Comparison of language coverage between native speakers in the world, Wikidata, and the web of data (LOD Laundromat, removing labels without language tag).

## 5.20 Comparison to the Web of Data

We applied the framework described in Chapter 3 to both the web of data in the form of the LOD Laundromat dataset in Chapter 4 and Wikidata in this chapter. To show the applicability of the framework for comparison between datasets, we here describe the differences between the results for the web of data and Wikidata. Table 5.7 provides an overview of the results for the core metrics for the web of data and Wikidata, Figure 5.16 shows the comparison of share of languages between native speakers, Wikidata, and the web of data (LOD Laundromat). The web of data is much larger in size than Wikidata, and it is also more varied in its labelling quality. Entities are labelled less frequently, and there is a tendency to label entities in only one language. We attribute these factors to the fact that the laundromat dataset we use to represent the web of data has a number of different data sources. For a user trying to make the decision which data source to use for which use case, this framework gives an insight into the different factors for the human accessibility (i.e., labels and languages) of different linked data sources. In Chapter 6 we automate the selection between different data sources using our framework.

| Metric | Web of Data | Wikidata |
|---|---:|---:|
| Dataset size | 127M entities | 26M entities |
| Natural language URI | mix | opaque |
| Labelling properties | mix (`rdfs:label`) | `rdfs:label` |
| Entity label completeness | 5% | 100% |
| Class label completeness | 38% | 100% |
| Property label completeness | 81% | 100% |
| Unamiguity | 3.5% | - |
| Multilinguality (English) | 18% | 11% |
| Monolingual Islands | 83% | 58% |

TABLE 5.7: Comparison of values for each metric between the web of data (represented by the LOD Laundromat dataset) and Wikidata.

## 5.21 Discussion

We present the results of applying the framework of Chapter 3 to Wikidata. First, we gain an insight into the overall language distribution in labels on Wikidata. While Wikidata has a better label coverage and language distribution than the web of data at large (see Chapter 4), improvement is still needed to make the knowledge graph accessible across languages. Currently there is a lack of labels in some of the most widely spoken languages in the world.

We further apply the *Data Creator Metrics* to understand how the editors of Wikidata contribute to language distribution. Understanding label editing is an important topic, as it can help to understand where a community needs automated help to improve language coverage in a given knowledge graph. We show that the languages users know (as described in their BabelBox) correlate with the languages available on Wikidata in labels. We analyse the editing history of Wikidata in terms of the label-editing behaviour of three user groups: registered editors, bots, and anonymous editors. We find that bots edit by far the highest number of labels, but edit less across different languages compared to registered users. Anonymous users have not only a low edit count in general and per user, but also a lower number of edit languages. Active users do not necessarily cover more languages in their editing.

# Chapter 6

# Application of the Framework for Knowledge Graph Ranking

In this chapter we propose a use case of the framework presented in Chapter 3. We show how applications can benefit from information about languages and labels. In particular, we present the use case described in Section 1.1, multilingual question answering over knowledge graphs. Multilingual question answering systems enable communities to easily access knowledge in their preferred language.

We tackle the problem of providing a fine-grained representation of multilinguality in knowledge graphs, and using these descriptions to rank knowledge graphs based on a set of SPARQL queries whose answer should be presented in different languages. We describe multilinguality in terms of CLCs, which capture knowledge about languages at the level of classes based on the metrics proposed in Chapter 3. In a CLC, diverse metrics are used to provide a fine-grained representation of multilinguality. These metrics are based on our approach to measuring the labelling and the multilinguality of knowledge graphs as described in Chapter 3; they show a way to standardise the application of those metrics on knowledge graphs to make them comparable and applicable in real-world applications. In this chapter we propose a multilingual framework, called *LINGVO*, which captures the CLCs that describe existing knowledge graphs. Further, LINGVO exploits the knowledge captured in the CLC-based descriptions to rank the described knowledge graphs according to their satisfaction of the language requirements stated in a set of SPARQL queries.

We empirically study the expressive power of *Class-based Label Captures* (CLC) by conducting the evaluation of the queries of the state-of-the-art benchmark QALD-9 (Usbeck et al., 2018). The goal of the study is to determine whenever the CLC-based representation of multilinguality of existing knowledge graphs like DBpedia, YAGO, Wikidata, MusicBrainz, and LinkedMDB allows for accurately ranking these knowledge graphs according to language requirements associated with

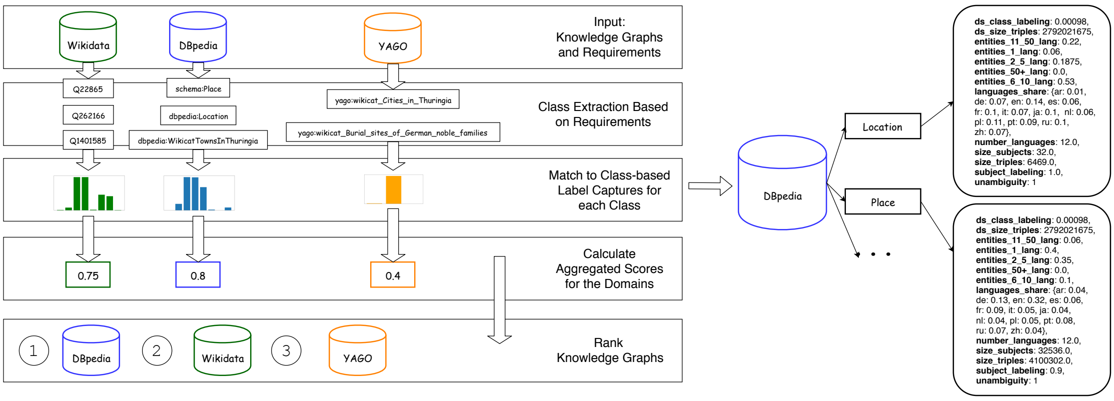


FIGURE 6.1: LINGVO approach. For each knowledge graph, classes for the required SPARQL query are extracted. Then, CLCs are matched to each extracted class with respect to the language provided in the required SPARQL queries. The respective scores of the CLCs are aggregated to finally rank the knowledge graphs. In this example, DBpedia is ranked highest and therefore recommended.

the benchmark queries. Capturing multilinguality at class and property level yields promising results, as we show in the comparison with the crowdsourced gold standard. The gold standard provides the best answer to each question of the benchmark between the knowledge graphs from a human perspective in three languages: English, Spanish, and Hindi.

Our contribution includes: (1) a fine-grained description model, CLC, capturing the multilinguality of knowledge graphs based on the framework described in Chapter 3; (2) a knowledge graph ranking approach, LINGVO, that exploits the CLC descriptions to rank the knowledge graphs that best satisfy the language requirements based on a set of SPARQL queries; and (3) a multilingual empirical evaluation including three languages (English, Spanish, and Hindi) based on the state-of-the-art benchmark and a new crowdsourced gold standard for the benchmark.

## 6.1 Problem statement

In this section, we define the problem of ranking knowledge graphs based on labelling and multilinguality for a set of queries, and propose the LINGVO approach. Let $U, B, L$ be an infinite disjoint sets of URIs, blank nodes, and literals, respectively. A triple $(s, p, o) \in (U \cup B) \times U \times (U \cup B \cup L)$ is denominated as an RDF triple, where $s$ is the subject, $p$ is the predicate, and $o$ is the object of the triple. An RDF graph is a set of RDF triples.

**Definition 6.1** (SPARQL Expression and SELECT Query, Schmidt et al. (2010))**.** Let $V$ be a set of variables disjoint from $U \cup B \cup L$. A SPARQL expression is built recursively as follows: (1) A triple pattern $t \in (U \cup V) \times (U \cup V) \times (L \cup U \cup V)$ is an expression. (2) If $Q_1$, $Q_2$ are expressions and $R$ is a filter condition, then $Q_1$ $FILTER$ $R$,

$Q_1$ $UNION$ $Q_2$, $Q_1$ $OPT$ $Q_2$, and $Q_1$ $AND$ $Q_2$ are expressions. Let $Q$ be a SPARQL expression and $S \subset V$ is a finite set of variables. A SPARQL `SELECT` query is an expression of the form $SELECT_S(Q)$

The evaluation of SPARQL queries over an RDF dataset is based on mappings. A mapping is a partial function $\mu : V \rightarrow ((U \cup B \cup L) \ \times \ \jmath)$ from a subset of variables to pair of RDF term and a language, e.g., EN, DE, or ES; $\eta$ is the set of all languages in the universe. The domain of a mapping $\mu, dom(\mu)$, is the subset of $V$ for which $\mu$ is defined. Two mappings $\mu_1, \mu_2$ are *compatible*, written $\mu_1 \sim_l \mu_2$, iff $\mu_a|_2 = \mu_b|_2 = l^1$, and $\mu_1(x) = \mu_2(x)$ for all $x \in dom(\mu_1) \cap dom(\mu_2)$, i.e., $\mu_1(x)|_1 \equiv \mu_2(x)|_1$ and $\mu_1(x)|_2 \equiv \mu_2(x)|_2$. Furthermore, $vars(t)$ denotes all variables in triple pattern t, and $\mu_l(t)$ is a triple pattern obtained when replacing all $x \in dom(\mu) \cap vars(t)$ in $t$ by $\mu(x)$.

**Definition 6.2** (SPARQL Algebra, Schmidt et al. (2010))**.** Let $\Omega_a, \Omega_b$ be mapping sets, $R$ denotes a filter condition, $S \subset V$ a finite set of variables, and $l$ a language. The expression of SPARQL algebraic operations are defined as follows:

$$\Omega_a \bowtie_l \Omega_b \ := \{\mu_a \cup_l \mu_b | \mu_a \in \Omega_a, \mu_b \in \Omega_b \colon \mu_a \sim_l \mu_b\}$$

$$\Omega_a \cup_l \Omega_b \ := \ \{\mu \mid \mu \in \Omega_a \ or \ \mu \in \Omega_b \wedge \mu|_2 = l\}$$

$$\Omega_a \setminus_l \Omega_b \ := \ \{\mu_a \in \Omega_a \mid for \ all \ \mu_b \in \Omega_b \colon \mu_a \not\sim_l \mu_b\}$$

$$\Omega_a \ ]\!\!\bowtie \ \Omega_b \ := \ (\Omega_a \bowtie_l \Omega_b) \cup (\Omega_a \setminus_l \Omega_b)$$

$$\begin{aligned} \pi_{S_l}(\Omega_a) \quad &:= \{\mu_1 | \exists \mu_2 \colon \mu_1 \cup \mu_2 \in \Omega_a \wedge dom(\mu_1) \subseteq S \wedge \\ &\quad dom(\mu_2) \cap S = \emptyset\} \end{aligned}$$

$$\sigma_{R_l}(\Omega_a) \quad := \{\mu \in \Omega_a \mid \mu \vDash R\}$$

**Definition 6.3** (SPARQL Semantics, Schmidt et al. (2010))**.** Let $D$ be an RDF knowledge graph, $t$ a triple pattern, and $Q_1, Q_2$ SPARQL expressions, $R$ a filter condition, $S \in V$ a finite set of variables, and $l$ a language. The expression $[[\cdot]]_D^l$ denotes the evaluation of an input SPARQL query over a RDF knowledge graph $D$ respecting a language $l$:

$$[[t]]_D^l \qquad := \{\mu \mid dom(\mu) \ = vars(t) \text{ and } \mu_l(t) \in D\}$$

$$[[Q_1 \ AND \ Q2]]_D^l := [[Q1]]_D^l \bowtie_l [[Q2]]_D^l$$

$$[[Q_1 \ OPT \ Q2]]_D^l := [[Q1]]_D^l \ ]\!\!\bowtie \ [[Q2]]_D^l$$

$$[[Q_1 \ UNION \ Q2]]_D^l := [[Q1]]_D^l \cup_l [[Q2]]_D^l$$

$$[[Q_1 \ FILTER \ R]]_D^l := \sigma_R([[Q1]]_D^l)$$

$$[[SELECT_S(Q_1)]]_D^l := \pi_S([[Q1]]_D^l)$$

[1] $|_1$ and $|_2$ denote the first and second element of the tuple, respectively, where the first element is an RDF term and the second a language.

**Definition 6.4** (Ideal KG)**.** An *Ideal KG*, $\overline{IK}$, is an RDF knowledge graph that contains, for all the entities in the universe, triples that associate the entities with existing languages. That is, for all the resources $s$ in the knowledge graph there exists (s, rdfs:label $o'$) such that $o'$ is a literal, annotated with all existing languages.

**Distance Measure.** Given a knowledge graph $KG$, a SPARQL query $q$ and a language $l$, the distance between answers of $q$ against $KG$ with respect to answers from the ideal of $KG$, $\overline{IK}$ is as follows:

$$d^{q}_{KG} := \frac{|[[q]]^{l}_{\overline{IK}} \setminus [[q]]^{l}_{KG}|}{|[[q]]^{l}_{\overline{IK}}|}$$

The aggregated distance between the answers of a knowledge graph $KG$ and an ideal knowledge graph $\overline{IK}$ for a given set of SPARQL queries $Q$, $d^{Q}_{KG}$ corresponds to

$$d^{Q}_{KG} := f(d^{q_i}_{KG}) \colon \forall q_i \in Q$$

where $f(.)$ is an aggregation function.

**Problem Statement** Given a set $K = \{KG_1, KG_2, \ldots, KG_n\}$ of knowledge graphs, and a set $Q = \{q_1, q_2, \ldots, q_n\}$ of annotated queries, where $q_i$ is a tuple $q_i = (sq_i, l_i)$ such that $sq_i$ refers to a SPARQL SELECT query and $l_i$ a language. The problem of capturing multilinguality corresponds to the problem of finding a set $\overline{K} = \{(KG_i, score_i) | KG_i \in K\}$ where $score_i$ is the ranking score of $KG_i$ with respect to other knowledge graphs in $K$ from the ideal knowledge graph, $\overline{IK}$. Measuring the distance between $KG_i$ and $\overline{IK}$ results in the distance score $d^{Q}_{KG}$. The knowledge graphs in $K$ are ranked with respect to their distance scores, and a ranking score is assigned, so that $d^{Q}_{KG_q} \preceq d^{Q}_{KG_k} \equiv score_q \geq score_k$.

**Proposed Solution** We propose LINGVO, a knowledge-driven framework able to describe knowledge graphs in terms of multilinguality, and to rank knowledge graphs according to SPARQL queries and multilingual restrictions. LINGVO considers class-based descriptions of data sources - an abstract description of entities that belong to same semantic type and their characteristics - to find a ranked list of knowledge graphs for a certain set of SPARQL queries. Such source descriptions are defined as CLC. While we could create CLCs for the datasets analysed in Chapter 4 and 5, we opt to describe the results to the reader as we have no need to compare them automatically. However, as with Chapter 5, we use Wikidata (see Section 6.3) in our comparison of knowledge graphs for question answering. Figure 6.1 depicts the components of the LINGVO approach, which receives a set of knowledge graphs and set of SPARQL SELECT queries. First, for the given queries, CLCs are extracted by matching the set of properties in CLCs with query patterns in the queries. Then, for each knowledge graph it calculates the aggregated score from each CLC. Finally, based on aggregated scores, LINGVO generates a ranked list of knowledge graphs, where the top of the list corresponds to the knowledge graph that best answers the queries given the language restriction.

## 6.2 Capturing Knowledge in LINGVO

LINGVO ranks knowledge graphs based on their multilingual properties at class level. We introduce CLCs, abstract descriptions of the entities in a knowledge graph. CLCs represent multilinguality based on dimensions described in subsection 6.2.2. Thus, each CLC states multilinguality in terms of metrics tailored towards the multilingual description of the labels of each entity in a knowledge graph.

### 6.2.1 Representing Labelling and Multilinguality of Knowledge Graphs

A knowledge graph contains a set of classes $C_{KG} = \{c_1, c_2, \ldots, c_n\}$. Each class has different characteristics, such as the number of languages or coverage of labels of its instances. We divide the knowledge graph based on the classes and extract for each $c_i$ in $C_{KG}$ a set of scores based on the dimensions in Section 6.2.2. We define CLCs as a source description, analogous to RDF-MTs (Endris et al., 2019), focusing on the labelling and multilinguality of each of the classes of a knowledge graph.

**Definition 6.5** (**Class-based Label Capture (CLC)**)**.** A CLC is a 4-tuple=<KG, C, DTP, CM>, where:

- KG – is a knowledge graph *G*;
- C – is an RDF class such that the triple pattern (?s rdf:type C) is true in *G*;
- DTP – is a set of tuples (p, PM) such that p is a labelling property with domain C and range `xsd:string`, the triple patterns (?s p ?o) and (?s rdf:type C) are true in *G*, and PM is a set of property level metric scores;
- CM – is a set of class-level metric scores

An example of the representation of a knowledge graph as CLCs can be seen in Figure 6.1 on the right. In the example, two classes in the DBpedia knowledge graph are represented as CLCs on `rdfs:label` property, `Location` and `Place`. The class-level metric scores, CM, are calculated over `Location` and `Place`, and the property level

| Metric | Application |
|---|---|
| Size | Dataset size (triples) |
| | CLC size (triples) |
| | CLC size (entities) |
| Class labelling | Dataset |
| Entity labelling | |
| Unambiguity | |
| Multilinguality | Number of languages |
| | Share of labels in given language |
| Monolingual Islands | |

TABLE 6.1: Metrics defined in Chapter 3. Metrics that are applied over the whole dataset are denoted with *dataset*, all other metrics are applied per CLC.

metric scores, DTP, are calculated based on the property `rdfs:label`. For each class of the knowledge graph, the scores of DTP and CM are calculated over all instances of a class, so that we can define the set of all metrics in a CLC as $M = \{m_1, m_2, \dots m_j\}$.

### 6.2.2 Framework for Labelling and Multilinguality

We measure multilinguality using the framework introduced in Chapter 3. As shown in Figure 3.1, the metrics used in this chapter are: *Size*, *Entity Labelling*, *Class Labelling*, *Unambiguity*, *Multilinguality*, and *Monolingual Islands*. The metrics and their application are detailed in Figure 6.1. For a definition of each metric, refer to Chapter 3. All metrics are applied across the entire dataset as well as on the CLC level, except class labelling, which is only applied on the entire dataset. To apply the metrics on the CLC level, we define all entities and relationships that are part of the CLC as a subset of the knowledge graph, so that we can treat one CLC as a dataset to apply the framework to. In the case of question answering systems, Diefenbach et al. (2018) observe in their survey of QA systems that those systems only consider labels identified with `rdfs:label` as the labelling property. Therefore, we consider only `rdfs:label` as labelling property in this chapter to gain an insight in the usage of the standard labelling property.

### 6.2.3 Ranking Knowledge Graphs by Queries and Languages

Given a set of knowledge graphs $K = \{KG_1, \dots, KG_n\}$, and SPARQL queries with language restrictions $Q = \{(sq_1, l_1), \dots, (sq_m, l_m)\}$, the LINGVO approach first matches each query to the set of classes in $KG_i$ such that $C^{sq_k}_{KG_i} = \{c_1^{KG_i,l_k}, c_2^{KG_i,l_k}, \dots, c_n^{KG_i,l_k}\}$. For example, in our motivating example in Figure 1.3 in Chapter 1, $Q$ contains three queries (all of them on "*where Bach was born*") with three different languages – German, English, and Spanish. The queries will be mapped to their respective classes as in Figure 6.1, i.e., the classes that describe *Germany*, so that for DBpedia for the Spanish query, $C^{sq_1}_{DBpedia}$ = $\{$`dbo:Location`, `dbo:Place`$\}$. For each class in $C^{sq_k}_{KG_i}$, the respective CLC is mapped with respect to the language $l_k$ of $sq_k$, i.e., $clc_i^{KG_i} \mapsto c_i^{KG_i,l_k}$. Thus, for our example, we retrieve the information about labelling and languages for the class `dbo:Location` for Spanish in the form of a CLC, which is then added to the set $CLC_{DBpedia}$. The set $CLC_{KG_i}$ contains all CLCs mapped to the classes extracted from $Q$ for $KG_i$, such that $CLC_{KG_i} = \{clc_1, clc_2, \ldots, clc_n\}$, where $clc_i$ contains information on the language of their respective $l_k$ of $KG_i$. We aggregate the values for all CLCs of a knowledge graph, $KG_i$, and rank based on the aggregated values. The aggregation of metrics in a CLC is performed as follows:

$$S_{clci} = \frac{\sum_{i=0}^{|M|} s(m_i)}{|M|}$$

where $s(m_i)$ is the score of metric $s(m_i) \in CLC_{ci}$ and $m_i \in M$. $S_{clc_{KG_i}}$ is the aggregation of the scores of the metrics associated with $CLCs$ from $Q$. The aggregation score in our example of `dbo:Location` for Spanish is 0.3, considering all the metrics. The ranking score of a knowledge graph is as follows:

$$R_{KG_j} = \frac{\sum_{i=0}^{|CLC_{KG_j}|} S_{clc_i}}{|CLC_{KG_j}|}$$

where $CLC_{KG_j}$ is a set classes that matched for the queries and language restrictions in $Q$. In Figure 6.1, DBpedia has a ranking score of 0.8 and is ranked first, compared to the ranking scores of Wikidata and YAGO.

## 6.3 Datasets

We conducted our evaluation over five widely used knowledge graphs. We selected three large cross-domain knowledge graphs, DBpedia, Wikidata, and YAGO, as they are widely used for QA (Diefenbach et al., 2018), and two domain-specific knowledge graphs, MusicBrainz and LinkedMDB. **DBpedia** is created by extracting information from Wikipedia in an automated manner and is available in a large number of languages (Lehmann et al., 2015). **Wikidata** is a collaborative knowledge graph, which is inherently multilingual and covers a large number of languages in terms of labels (for more details, see Chapter 5). **YAGO3** is extracted from Wikipedia and other structured resources and provides multilingual data (Mahdisoltani et al., 2015). **MusicBrainz** stores data about music, such as artists and their songs. Like Wikidata, MusicBrainz data is user-contributed (Swartz, 2002). **LinkedMDB** represents structured information about movies by combining information from different web sources on the topic (Hassanzadeh and Consens, 2009). To test our approach on a set of SPARQL queries, we adapted the QALD-9 dataset (Usbeck et al., 2018), which was designed for the 2018 question answering challenge over DBpedia. Natural language questions in different languages are expressed in the respective SPARQL query. As we want to test our approach for labels, we only select question that return URIs. Our benchmark consists of 289 questions out of the original 408 questions. The requirement queries in the benchmark are defined over three languages, English, Spanish, and Hindi, which are covered by QALD. English is the language mostly used on the web, Spanish is a widely spoken language and often covered online[2], and Hindi is one of the most widely spoken languages in the world but under-resourced online[3]. Since the SPARQL queries in QALD are defined for the DBpedia ontology, to make the different knowledge graphs comparable, we translate the queries into the ontologies of each benchmark knowledge graphs manually, i.e., into Wikidata, MusicBrainz, LinkedMDB, and YAGO ontologies. We skip queries that cannot be translated to the other ontologies, e.g., *Which museum in New York has the most visitors?* cannot be expressed in MusicBrainz's ontology and, therefore, this query is not included for this knowledge graph. The total number of queries translated for each knowledge graph can be found in Table 6.3.

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| Donald Trump | Trump | Donald Trump | Barack Obama |
| | | Barack Obama | |
| | | George Washington | |

TABLE 6.2: English example of answers displayed to annotators for the question *Who is the president of the United States of America?*. The annotator has to select between the four knowledge graphs the one that fits the question best. In this example, an annotator should select knowledge graph (1). For more details, see Appendix B.

## 6.4 Gold Standard

We create a gold standard to measure the correctness of our approach and the baselines, i.e., how well the approach can rank knowledge graphs based on a given query compared to the human annotations. Since we aim at human readability, we crowdsource the ideal ranking of answers to the QALD questions. We created three crowdsourcing experiments on Figure Eight[4] - one for each of the languages Hindi, Spanish, and English. The guidelines for the crowd annotating the data can be found in Appendix B. For each question, we retrieve the labels of answers in each language

[2] Spanish is the fourth language in terms of website content (3.7%) according to https://w3techs.com/technologies/overview/content_language, retrieved 9. June 2021.

[3] 0.1% of the websites are in Hindi, according to https://w3techs.com/technologies/overview/content_language, retrieved 9. June 2021.

[4] https://www.figure-eight.com/

from the benchmark knowledge graphs. If only one knowledge graph answers the question, that knowledge graph ranks best automatically. For example, for Hindi, only YAGO and Wikidata contain a large amount of Hindi, while MusicBrainz has only one entity tagged in Hindi, and LinkedMDB and DBpedia do not contain Hindi data. Questions that can be answered by more than one knowledge graph are selected for evaluation by the crowd. They are asked to choose the answer that they see as the best fit. Table 6.2 displays the possible answers for the question *Who is the president of the United States of America?*. Annotators see the question and have to select between the answers most appropriate for the question (answer 1 in the example). If they are not sure which is the correct answer, they have the option to select the checkbox labelled *X not sure about the factually correct answer*.

We selected a total of 46, 203, and 237 questions and their answers for Hindi, Spanish, and English, respectively. We assigned annotators that are able to speak the respective language, are on level 2 on Figure Eight[5], and passed the eight test questions created by the researchers. For each language, we received judgements from 3 annotators, who were paid $0.02 per HIT. The correct answers were selected based on majority voting. Agreement was high; on average the annotations had a confidence score of 0.87, 0.86, and 0.84 for Hindi, Spanish, and English, respectively. For the ranking, the ranking score of a knowledge graph is calculated based on the number of questions it can answer best as seen by the crowd over the given queries.

## 6.5 Baselines

We use three baselines for our approach, i.e., **CosN**, **Cos**, and **noCLC**. The first baseline (**CosN**) executes the queries over each knowledge graph and collects the number of answers. It exploits the fact that QALD does not provide labels but URIs in the answer set, so that we know the number of expected answers per question. CosN represents the domain as a vector of the number of answers for each query in the domain as $VA^{KG_i}\langle a_{q_1}^{KG_i}, a_{q_2}^{KG_i}, \ldots, a_{q_n}^{KG_i}\rangle$ and the number of answers from QALD for queries $q_i$ in $Q$ as $VA^{QALD} = \langle a_{q_1}^{QALD}, a_{q_2}^{QALD}, \ldots, a_{q_n}^{QALD}\rangle$. We measure the similarity between this vector and the actual number of answers per domain for each knowledge graph using cosine similarity, i.e., the score of a knowledge graph, $KG_i$, is calculated as $score_{cosN}^{KG_i} = cos(VA^{KG_i}, VA^{QALD})$.

The second baseline, **Cos**, executes the queries in the benchmark and treats the answer to a query as a binary value (returns any label or no label) in a vector. Then, it computes the similarity to the ideal vector, where all queries return a label. Let $(V^{KG_i} = \langle b_{q_1}^{KG_i}, b_{q_2}^{KG_i}, \ldots, b_{q_n}^{KG_i}\rangle)$ be the vector of binary values whether a query of $Q$ has an

[5] Levels on Figure Eight indicate experience, there are three levels. Appen, who took over Figure Eight, detail the levels here: https://success.appen.com/hc/en-us/articles/203219195-Guide-To-Contributors-Channels-Page, retrieved 7. May 2021.

| KG | # CLCs | Avg # CLCs | Max # CLCs | # Queries |
|---|---|---|---|---|
| DBpedia | 112,888 | 811.72 | 100,000 | 285 |
| YAGO | 13,642 | 77.39 | 1,972 | 169 |
| Wikidata | 874 | 4.49 | 268 | 270 |
| LinkedMDB | 4 | 0.07 | 2 | 20 |
| MusicBrainz | 3 | 0.01 | 2 | 4 |

TABLE 6.3: CLCs for each KG covered by QALD questions and number of QALD questions translated to the respective ontology. DBpedia has the largest numbers of classes and average number of classes per QALD questions, as well as the largest maximum number of classes for one QALD question.

answer when executed over knowledge graph, $KG_i$, so that $b_j^{KG_i} \in [0,1]$. Let $V^I = \langle b_{q_1}^I, b_{q_2}^I, \ldots, b_{q_n}^I \rangle$ be the ideal vector of binary answers to the queries in $Q$ so that $b_{q_j}^I = 1$. We calculate the cosine similarity between the two vectors such that the score for a $KG_i$ is calculated as $score_{cos}^{KG_i} = cos(V^{KG_i}, V^I)$.

The third baseline, **noCLC**, applies the dimensions of Section 6.2.2 over the entire knowledge graph to produce a ranking. Unlike the first two baselines, **noCLC** does not execute the queries over the knowledge graph, but computes scores by treating each knowledge graph as one domain. We create a CLC in which we define the entire knowledge graph as one class to measure the effectiveness of separating the knowledge graph in multiple classes.

For all baselines, we calculate the average score for a knowledge graph over the domain, by summing the scores over all queries in the domain. The same approach as for LINGVO is used to rank the knowledge graphs. We consider these approaches as a baseline because they rank datasets in a naive way. For instance, the first two approaches execute all the queries, and the last approach considers the whole knowledge graph as a single domain, ignoring the fact that cross-domain knowledge graphs have different level of coverage in labelling and languages.

## 6.6 Experiments

We empirically study the effectiveness of LINGVO in capturing the multilinguality of knowledge graphs. The goal is to show that the application of CLCs is a solid approach for ranking knowledge graphs, based on their multilingual properties compared to the human judgements. Below are the details of the experimental setup and results.

**Research Questions** We hypothesise that our approach can capture the multilinguality of the knowledge graphs, so that the distance between the resulting ranking and the ideal ranking is minimal. We aim at answering the following research question during the evaluation of our approach based on the overall research question **RQ2 How can knowledge about languages in a knowledge graph be applied to the task of ranking them for question answering?** introduced in Chapter 1.

**RQ2.1** Are CLCs able to capture multilingual knowledge?

**RQ2.2** Given a set of multilingual queries, can LINGVO precisely identify the knowledge graphs that can answer those queries in the best way?

**RQ2.3** Is the knowledge about a domain represented in the knowledge graph affecting the multilinguality of the knowledge graph?

**Evaluation Metrics** We evaluate the ranked lists three-fold using **Kendall's tau**, **Spearman's rho**, and **RankDCG** rank correlation coefficient. The Kendall and Spearman correlation is high between two rankings if they have a similar rank, i.e., a score of 1 for two identical lists and $-1$ for fully different lists. To take the actual scores for the ranking into account, we also apply a version of *normalized discounted cumulative gain*, RankDCG, introduced by Katerenchuk and Rosenberg (2016). A larger value for RankDCG indicates similar values in the ranking.

**Implementation** Our approach is implemented in Python 2.7. We set up a Virtuoso endpoint for each of the five knowledge graphs used in the evaluation. The code of LINGVO is available at `https://github.com/luciekaffee/LINGVO`, the code for the benchmark is available at `https://github.com/luciekaffee/Benchmark-Queries`.

### 6.6.1 Random Domains

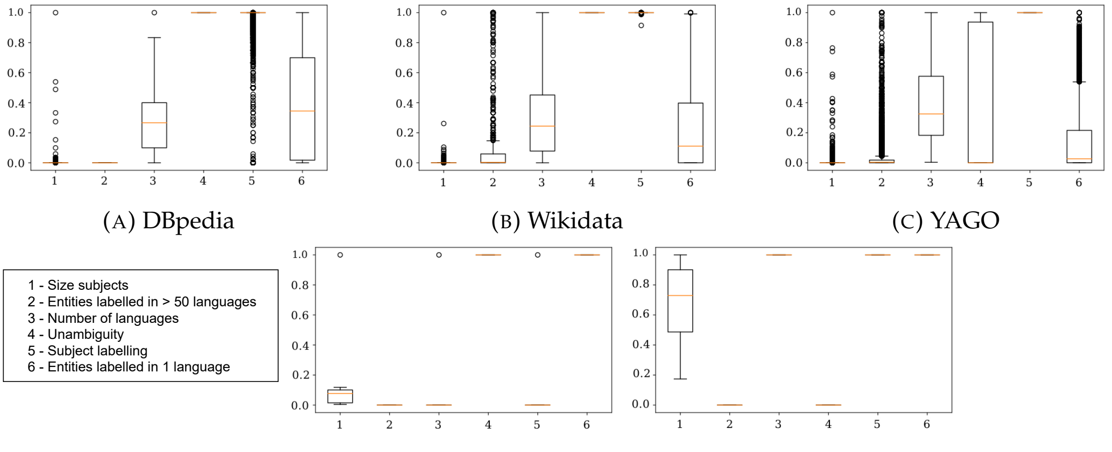
In this experiment, we investigate ranking the knowledge graphs based on given multilingual queries in an automated manner.

We create five domains each consisting of 20 randomly selected queries from the benchmark. Each query in a domain is assigned a language randomly - English, Spanish, or Hindi. The knowledge graphs are then ranked using the baselines and the LINGVO approach for the five domains. We measure the ranking correlation between the computed rankings with the gold standard ranking based on the RankDCG, Spearman-Rho, and Kendall-Tau evaluation metrics.

### 6.6.2 Gold-standard Domains

We evaluate the impact of the domains on the performance of the ranking approach. We show that our approach performs over a selection of random domains. For this experiment, we investigate how our approach performs over domains that are contextual, where all queries are in a human-selected domain.

We manually categorise the benchmark queries into 17 domains, one domain per question. From those 17 domains, we select 5 domains that have over 20 queries or represent the domains of the domain-specific knowledge graphs: `movie` (36 queries, domain of MusicBrainz), `music` (19 queries, domain of LinkedMDB), `location` (82 queries), `politics` (35 queries), and `company` (21 queries). For each query of the domain, we randomly select a language between English, Spanish, and Hindi for the multilingual setting.



(A) DBpedia

(B) Wikidata

(C) YAGO

(D) MusicBrainz

(E) LinkedMDB

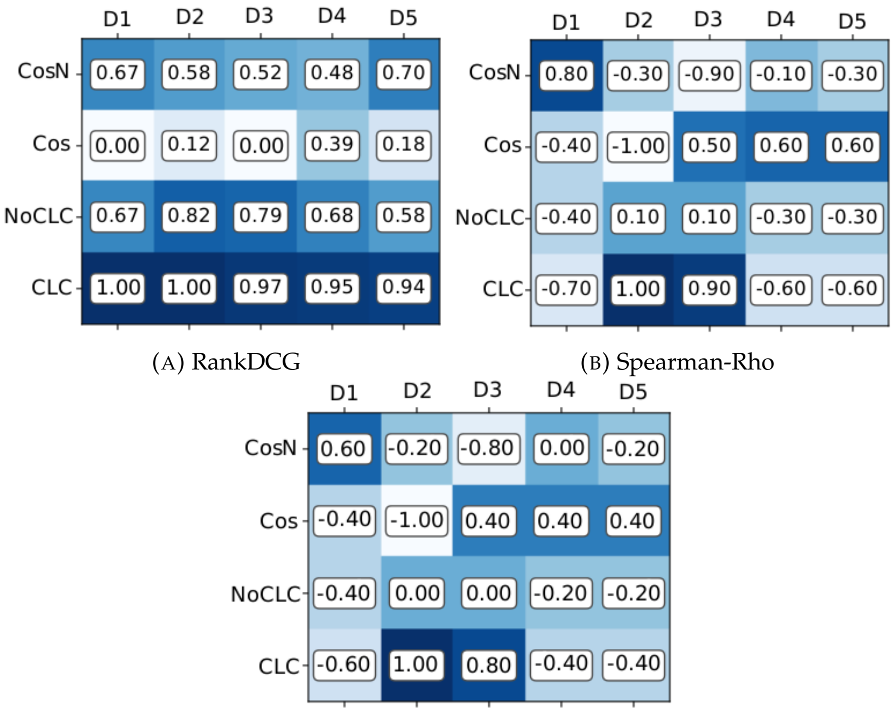
FIGURE 6.2: Boxplots of classes used for QALD. Scores are between 0 and 1; a higher variety in the scores between different classes can be observed with a higher number of classes overall.

## 6.7 Results

LINGVO captures knowledge about multilinguality of the knowledge graphs on class level. The QALD benchmark queries cover a large number of classes, e.g., up to 100, 000 classes as answer types for one query in DBpedia. The number of classes that match the queries for each knowledge graph can be found in Table 6.3. Wikidata has fewer total and average number of classes for the 270 queries that could be expressed in the Wikidata ontology than YAGO and DBpedia. YAGO and DBpedia often have a higher number of answers per question, and a larger number of classes per answer. The high variance in number of classes covered indicates a variation of coverage between topics in the knowledge graphs. Further, we show in Figure 6.2 that the scores between different classes vary strongly. The three knowledge graphs with a higher number of CLCs (DBpedia, Wikidata, and YAGO) have a higher variance in the scores between the different classes. Expressing information at class level captures those variances in language knowledge in the classes of a knowledge graph. In the following, we show how we exploit this knowledge about variances expressed in CLCs for ranking knowledge graphs.

### 6.7.1 Random Domains

Figure 6.3 presents the results of ranking knowledge graphs over five randomly defined domain queries and their correlation to the gold standard ranking. There is a high variance between the scores for the different domains. For example, the correlation score for **CosN** for **Spearman Rho** is the highest; however, the baseline performs worse overall, whereas the **Cos** baselines produces low scores for all domains with respect to the gold standard. The LINGVO ranking, **CLC**, scores higher in many domains for the **Spearman-Rho** (Figure 6.3b) and **Kendall-Tau** (Figure 6.3c) metrics. As described previously, RankDCG (Figure 6.3a) is the more expressive metric, as it evaluates the ranking in more detail. For this metric, the LINGVO ranking produces the best correlation scores over all domains, because interchangeable ranks



(A) RankDCG

(B) Spearman-Rho

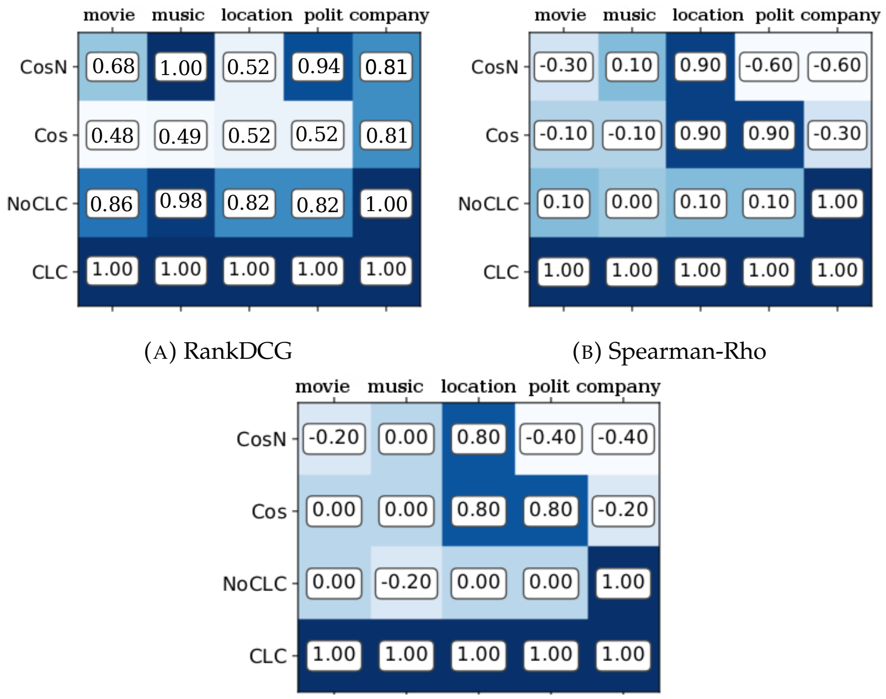
(C) Kendall-Tau

FIGURE 6.3: Random domains ($D1 - D5$). Results for correlation between the gold standard ranking to computed ranking. Each domain consists of 20 randomly selected queries from the QALD dataset. Three baseline ranking approaches, **CosN**, **Cos**, and **NoCLC** and the LINGVO approach, i.e., **CLC**. The addition of CLCs leads to higher scores in the ranking over all domains.

in the gold standard can be taken into account. I.e., if two knowledge graphs rank the same (both are second best), we can consider their exact ranking.

### 6.7.2 Gold-standard Domains

Figure 6.4 shows the result of ranking knowledge graphs for the selected domains. Our approach scores the best results compared to all baselines. Since the domains are created by selection queries from the benchmark queries that are related to a specific context, the performance of the LINGVO approach in capturing the multilinguality of knowledge graphs was superior to the randomly generated domains (in Section 6.7.1). The naive baselines (Cos and CosN) perform slightly better over the selected domains compared to random domains. Our ranking performs better than the baselines as it considers the domains in the form of classes. Further, this experiment setup yields better results than the random results because the selected domains are related compared to the randomly created ones. This indicates that the selected domains have similar properties in terms of labels and languages, showing that domains affect the multilinguality of a knowledge graph (RQ2.3).



(A) RankDCG

(B) Spearman-Rho

(C) Kendall-Tau

FIGURE 6.4: Selected domains. Results for correlation between the gold standard ranking to three baseline ranking approaches, **CosN**, **Cos**, and **NoCLC** and the LINGVO approach, i.e., **CLC**. The domains are contextual. The addition of CLCs leads to higher scores in the ranking over all domains.

## 6.8 Discussion

We show that our approach to extracting meta information at class level about languages and labelling in a knowledge graph based on CLCs can achieve promising results for capturing knowledge about multilinguality and labels in a way that can be reused for applications, particularly for the ranking of knowledge graphs for question answering.

In the experimental study using randomly selected queries for the domains, the LINGVO approach outperforms the baselines. Among the baselines, NoCLC performs very well, too. NoCLC applies the metrics over the entire knowledge graph. It shows an improvement over the naive approach, indicating that working only on metadata rather than the actual queries is a good direction toward creating an ideal ranking. The results show that our metrics capture the variance of knowledge graphs between classes. The CLCs capture multilinguality in a way that lets us reuse the framework introduced in Chapter 3 in applications, so that we are able to rank knowledge graphs based on their multilingual features (RQ2.1 and RQ2.2).

In the experimental study with domains manually selected based on the queries' topics, we show that our approach performs better in a setting with a limited number of topic areas, and conclude that adapting to a domain is beneficial when evaluating and ranking knowledge graphs. The NoCLC baseline is static and does not adjust to the requirements of the domain. For all languages, it ranks DBpedia highest, followed by Wikidata, YAGO, LinkedMDB, and MusicBrainz. This ranking scores high for most settings, as it takes into account the overall coverage. However, for many tasks, the crowd favoured Wikidata answers over DBpedia.

Thus, scores for this baselines are not competitive in different settings. Especially for the queries that are restricted to Hindi, DBpedia scores low because it does not provide Hindi content. This shows that it is necessary to consider domains when looking at the multilinguality of a knowledge graph, especially for applications where a single label, or labels of a single domain, are required.

# Chapter 7

# Transliteration and Translation of Knowledge Graph Labels

In the previous chapters demonstrated that there is a lack of multilingual labelling on the web of data. Achieving adequate multilingual coverage for knowledge graphs is crucial for enabelling users of different languages to access the data.

The main way to refer to an entity in a knowledge graph is by using its official name, or *label*, and use the `rdfs:label` property with an associated language tag. For example, in the triple `wikidata:Q312 rdfs:label ‘‘Apple Inc.’’@en`, the latter part “`@en`” indicates the language code that allows the assignment of a single label in multiple languages to an entity. Entities may have additional and different surface forms in natural language, known as *aliases*, which are important for downstream NLP tasks. For example, the company `IBM` is commonly known as “Big Blue” and therefore both “IBM” and “Big Blue” are valid aliases of `IBM`. Entities can have at most one label per language, as well as multiple aliases.

We focus in this chapter on translation and transliteration from English to Chinese. Chinese is the most widely spoken language in the world, but its coverage in knowledge graphs is lacking, as we showed in Section 5.12. In the company domain, for example, Wikidata contains over 145k company entities with English labels, but only about 5k of those companies have a label in simplified Chinese. This gap in cross-lingual information is difficult to address, as manually translating labels is prohibitively expensive.

In this chapter we explore transliteration and translation of entity labels and aliases by generating Chinese aliases from English entity labels. Entities’ labels in a knowledge graph are a mix of transliterated and translated items. Transliteration transfers a word from one script to another, while translation transfers the meaning of a word from one to another language. In contrast to people’s names, which are always transliterated (Merhav and Ash, 2018), knowledge graph entities have no indication of whether to use transliteration or translation to to transfer them from one language to another We therefore explore two approaches to selecting between transliteration and translation: one using rule mining, and the second using knowledge graph embeddings. We follow the intuition that referential information in the knowledge graph can be an indicator for lexical information. For example, the company *Toyota* has the *named after* property, linking its founder. People’s names are transliterated, giving an

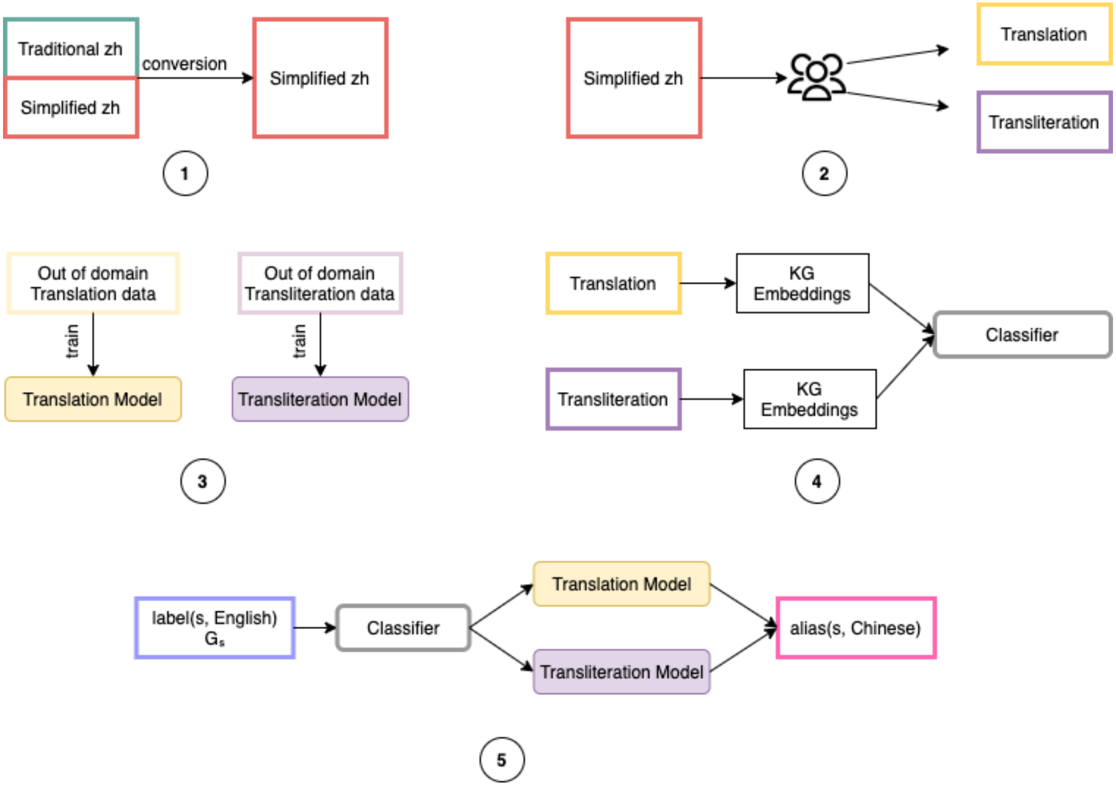


FIGURE 7.1: Overview of the contributions in this chapter. (1) Conversion from traditional to simplified Chinese (zh). (2) Crowdsourcing to differentiate between translation and transliteration in the entities. (3) Training two models, one for transliteration and one for translation, on out of domain data. (4) Training the classifier on the entities classified as transliteration or translation by the crowd. (4) Overview of the final pipeline.

indicator that the company name is to be transliterated, too. To generate training data for the automatic classification as transliteration or translation, we created a crowdsourcing task in which annotators annotate label-alias pairs as transliterated, translated, or a mix or transliteration and translation. Since manually translating all labels in a knowledge graph is prohibitively expensive, and datasets for automatic transliteration and translation are rare, we explore the translation and transliteration of entity labels and aliases using out-of-domain data for the tasks of transliteration and translation.

We test our approach on the company domain in Wikidata. The company domain has many applications in the areas of enterprise and finance where there is a focus on market intelligence or stock market data (Gschwind et al., 2019). We focus on company entities as they present a useful microcosm of the overall challenges of knowledge graph entity translation, such as the mix of translation and transliteration in cross-lingual labelling. In the following study, we explore the creation of company aliases in simplified Chinese by translation or transliteration of English labels. For example, beginning with the company label *Google*, we generate the Chinese alias 谷歌, based on information in the knowledge graph. While previous work on knowledge graph translation by Moussallem et al. (2019) has focused on translating knowledge graphs from English to German, we here choose to work with English and Chinese because they are two of the most widely spoken languages in the world, and because they use different scripts.

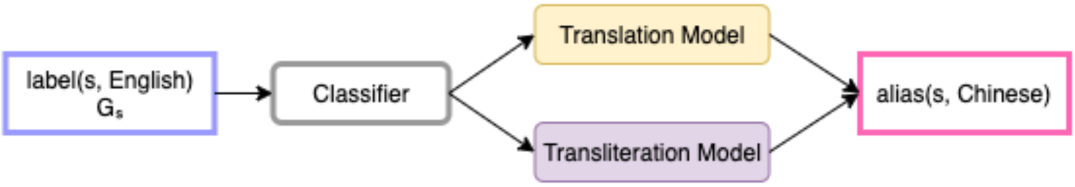


FIGURE 7.2: Approach: given an entity $s$, its English label, and its respective triples ($G_s$), we classify it as either translation or transliteration. With this knowledge, we can run it through a model which is specialised in either translation or transliteration, and trained specifically for that task. The resulting Chinese-language alias will be checked for correctness.

## 7.1 Problem Statement

We formulate the problem as follows: A knowledge graph $G$ is represented as triples, while $G_s$ for an entity $s$ represents the subset of all triples $(s, p, o)$, where $p$ is the predicate and $o$ the object of the relationship ($G_s \subset G$). Given a language $l$ and an entity $s \in G$, we denote $label^{s,l}$ as the entity's label, while $A^{s,L} = \{alias_1^{s,l}, \ldots, alias_n^{s,l}\}$ is the list of all possible $n$ aliases of $s$ for the language $l$.

Given a source language $l_{src}$ and a target language $l_{tgt}$, $label^{s,l_{src}}$ (the $l_{src}$ label of entity $s$), we want to generate an $alias^{s,l_{tgt}} \in A^{s,l_{tgt}}$. In our case we focus on source language English and target language Simplified Chinese, while all entities of focus are of type company.

In our work, we analyse company (label, alias) pairs to learn how they can be mapped. We differentiate between translation and transliteration of company labels and automate the classification as transliteration or translation, training a model for transliteration and one for translation.

## 7.2 Approach

Given an entity $s$, its English label, and its respective triples ($G_s$), we want to generate an $alias^{s,l_{tgt}}$ in simplified Chinese. We approach this problem by proposing a pipeline as described in Figure 7.2. An entity label is automatically classified as either transliteration or translation. The experiments for the classifier are described in Section 7.4. We create a crowdsourcing task to collect human annotations for company label-alias pairs, described in Section 7.3.

We create two different models: one for translation and one for transliteration, which we test on the company (label, alias) pairs annotated by the crowd. Due to the limited training data in the company domain, we train both models on out-of-domain data and report the results. We describe the training data in Section 7.5, the model setup and evaluation in Section 7.6, and the design of the transliteration and translation model in Section 7.7.2 and Section 7.7.3 respectively.

The results for our overall approach can be found in Section 7.9.

## 7.3 Crowdsourcing Task

In the company domain, we observe four types of relations between $label^{s,L_{src}}$ and $alias^{s,L_{tgt}}$: (1) Translation of $alias^{s,L_{tgt}}$ from $label^{s,L_{src}}$ - is the transfer of a concept from one language into another. For example, the English *Apple* is translated as 苹果. (2) Transliteration of $alias^{s,L_{tgt}}$ from $label^{s,L_{src}}$ - is the transfer of the same word from one script to another on a phonetic basis. For example, the English *Google* is transliterated to 谷歌 in Chinese. (3) Partial translation/transliterations of $alias^{s,L_{tgt}}$ from $label^{s,L_{src}}$ (4) *Not aligned* appears when one cannot align $alias^{s,L_{tgt}}$ from $label^{s,L_{src}}$ as translation or transliteration. For example, the English *IBM* is not aligned with 国际商业机器 in Chinese which means *International Business Machine*. Note that an English label of an entity can have multiple Chinese aliases.

**Crowdsourced Data Annotation** We create a data annotation task in which we ask Chinese native speakers to select from "transliteration", "translation", "partial translation", and "not aligned" given an English company label and a Chinese alias for the same entity. For the partial transliteration category, annotators mark parts that are translated or transliterated and add English translations for each.

The annotated dataset contains 3,056 pairs of English and Chinese labels and aliases from Wikidata. Each pair is annotated by three annotators. If there is disagreement between the annotators, we select the correct annotation using majority voting.

The results of the crowd annotating (English label, Chinese alias) pairs can be found in Table 7.3. Overall, the biggest share of companies are partially translated or transliterated (66.8%). In this category fall also companies, where one part is translated and the rest is not aligned. Krippendorff's alpha for the annotations is 0.55, increasing to 0.76 when only considering translations and transliterations, indicating higher disagreement when detecting partial and not aligned data. We run the study iteratively, removing annotators who had a low agreement with either the gold standard of 30 samples or the other annotators. The goal of our experiments is to understand whether we can improve the quality of the Chinese alias generation from English labels by dividing transliterations and translations. Hence, we focus on the subset which was marked as either translated or transliterated.

**Partial Data** While the following approach only considers company labels and aliases that are fully translated or transliterated, the biggest proportion of company aliases were annotated as partial. These include also partial alignments, e.g., *Apple* in English but *Apple Inc.* in Chinese, where only *Apple* can be aligned. We leave it to future work to include these partial company aliases, but want to give an overview of this subset of the data here. We investigate the partial data in terms of repeated patterns in the data. It could be used to build, for example, a dictionary for human translators to refer back to.

of the 1,965 (label, alias) pairs annotated as partial, 1,411 are annotated with the same alignments in the fragments of translation or transliteration between English and Chinese. We extract patterns by using the aligned English-Chinese fragments. We extract 118 patterns that are unique (English, Chinese) pairs. Overall, each pattern is repeated 4.98 times, with 1.81 different Chinese words used, on average, for one English fragment. When using majority

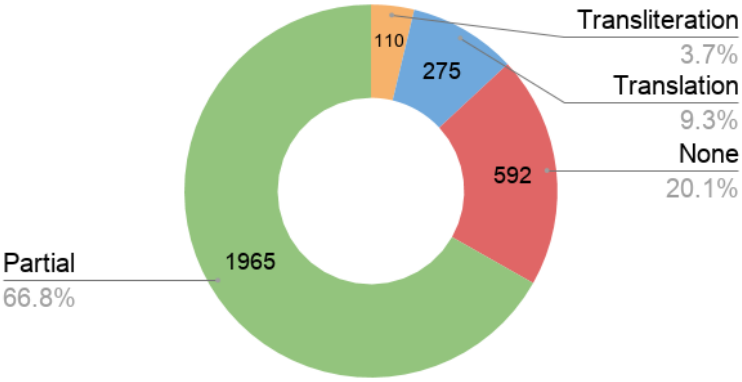


FIGURE 7.3: Crowdsourcing results. Number of (English label, Chinese alias) pairs categorised as Translation, Transliteration, Partial or None, and their percentage.

| | Accuracy | F-score | Precision |
|---|---|---|---|
| AMIE | 0.50 | 0.00 | 0.00 |
| fastText | **0.76** | **0.48** | **0.64** |
| BigGrap | 0.68 | 0.42 | 0.47 |

TABLE 7.1: Classification of the entities as translation or transliteration, using knowledge graph embeddings-based classifiers.

voting, we can extract 182 patterns. These patterns mostly contain company types (such as *co., ltd.*) and company fields (such as *gas* or *coal*). The most common patterns used are "expressway" (11), "holdings limited" (10), "mining" (6), and "hong kong" (6).

## 7.4 Automatic Transliteration/Translation Classification

For the entities' classification as candidates for transliteration or translation, we leave out partial data and focus on classifying only into two classes, namely, transliteration and translation. We investigate the possibility of deducing whether an entity would be transliterated or translated from its relationships and properties, such as location, company size, and so on. We introduce two classifiers: rule mining and knowledge graph embeddings. Based on the annotations of the human annotators, we evaluate the classifiers using Accuracy, F-Score, and Precision. Because we classify based on entities and their triples, if we have multiple (label, alias) pairs for the same entity, we only consider one classification. For example, if an entity $e$ has one label $l$ in English and two corresponding aliases $a_1$ and $a_2$, and one is a transliteration of $l$ and one is a translation of $l$, we pick one class (translation or transliteration) at random. We consider these edge cases, as we only consider exact alignments between labels and aliases, and the same entity having a valid transliteration and translation as alias of the English label only occurs twice in our dataset.

### 7.4.1 Rule Mining

Rule mining is an approach to knowledge base completion. Given a set of triples, patterns are extracted that are formulated into rules for the knowledge graph. These rules can be applied to other triples to extend the coverage of the knowledge graph. A commonly used framework for rule mining is provided with AMIE (Galárraga et al., 2013) and AMIE+ (Galárraga et al., 2015). For example, given the following three triples:

1. `WWE – stock exchange – NY Stock Exchange .`
2. `WWE – replaces – World Wrestling Federation .`
3. `World Wrestling Federation – stock exchange – NY Stock Exchange .`

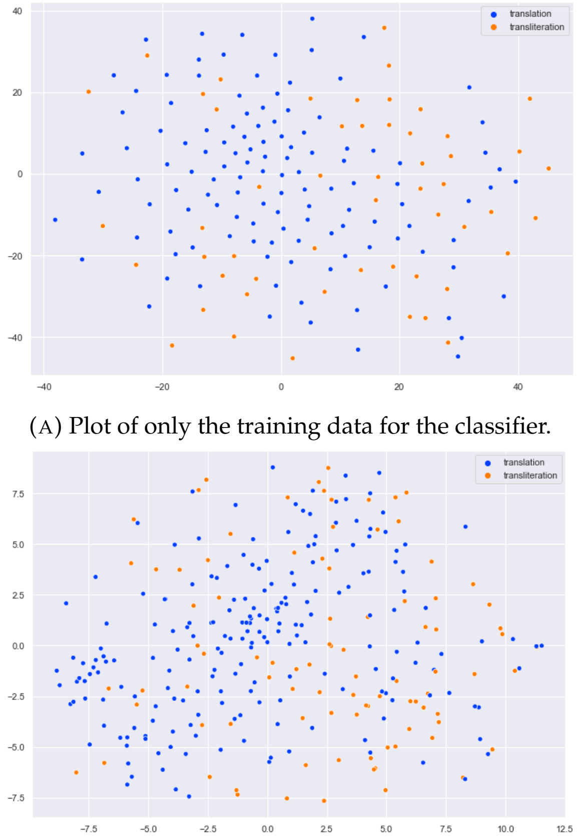
and if one has multiple such triple types, a rule could be extracted that states if company $c_1$ replaces company $c_2$ and $c_1$ is traded on stock exchange $s$, then $c_2$ must also be traded on stock exchange $s$.

Using this method for classification, we first split all entities between translation and transliteration based on the crowd annotations. Then, for each entity, we extract their respective triples from Wikidata. In our training set, there are 62 Chinese-English pairs for transliteration, for which we extract $1,184$ triples, and 140 Chinese-English pairs for translation, for which we extract a total of $1,536$ triples.

We apply AMIE on the entities and their triples categorised as transliteration, and on the entities and their triples categorised as translations separately, extracting two set of rules. From those rules, we extract the properties in each set of rules and for each of the entities in the test set. We test if transliterations or translations are more likely to have one set of properties. We extract 3 rules for transliteration, and 25 rules for translation. The high divergence in number of rules can be explained by the much higher number of entities available for translation. We then extract the properties for each set of rules. There are 18 properties in the rules for transliteration compared to 41 properties in the rules for translation. When classifying each company name in the test set into either transliteration or translation based on their use of triples, all entities are classified as translations, as there are more translations than transliterations properties. Therefore, this classifier is not able to give usable results and is discarded.

### 7.4.2 Knowledge Graph Embeddings

Knowledge graph completion and link prediction are in the recent literature approached using knowledge graph embeddings, in which an entity is represented in a vector (Cai et al., 2018). We reuse this technology because it can be used to grasp a lot of information about an entity as input for a machine learning model. We trained a binary random forest classifier on the knowledge graph embeddings for transliteration and translation entities. We leave experimenting with different machine learning models for the classifier for future work. The classifier predicts whether an entity is to be translated or transliterated based on the entity's embedding, i.e., the vector. We hypothesise that its triples can give us an indication as to whether a company label is translated or transliterated. This meta information shapes the



(A) Plot of only the training data for the classifier.

(B) Plot of all entities, including training, test and dev data.

FIGURE 7.4: TSNE scatter plot of entities used for the classification; blue for translation, orange for transliteration as annotated by the crowd.

embedding of an entity - in our case, a company. We test two approaches to knowledge graph embedding: fastText-based embeddings, and pretrained Wikidata embeddings by BigGraph (Lerer et al., 2019). fastText has been previously used for knowledge graph translation in the work of Moussallem et al. (2019); Yang et al. (2019), but it is text-based and thus not optimised for knowledge graphs. We therefore focus our experiments on an embeddings technology that was designed specifically to capture knowledge graph information.

As we can see in Table 7.1, all accuracy scores are below 0.8. fastText and BigGraph perform very similarly, with slightly better scores for fastText. We opt to use BigGraph for the experiments in Section 7.9, as BigGraph encodes the triples rather than having the triples as text, and we believe it is a more appropriate approach for the task. The classifiers are able to capture the ratio between translations and transliterations, as they are similar to the gold standard's distribution (about 60% fewer transliterations). We attribute the overall low scores to the fact that knowledge graph embeddings encode information that can not be leveraged for classification as transliteration and translation. This is also seen in the plotting of entities using T-SNE in Figure 7.4 (Maaten and Hinton, 2008). While there is a cluster of entities of which the aliases are translated, there is no clear trend, which complicates the task of our classifier.

| Dataset | # of pairs |
|---|---:|
| **Person names** | 61, 309 |
| **CEDICT** | 117, 598 |
| **Pinyin Characters** | 24, 452 |
| **Pinyin Dataset** | 145, 953 |
| **Total transliteration dataset** | 190, 062 |

TABLE 7.2: Size of the datasets used in the transliteration experiments (transliteration data and pinyin data). Duplicates are removed in the total transliteration dataset.

## 7.5 Out-of-domain Training Data

Company aliases' translation and transliteration data is sparse. As detailed above, there is a lack of, e.g., company labels in simplified Chinese in Wikidata. Even if there are labels, labels do not need to be exactly aligned, e.g., the difference between *Apple* and *Apple Inc.* as labels for the same company. Creating this data is very costly, as it requires manual work in the scope needed for state of the art neural models. Therefore, we utilise out-of-domain data to enable us to transliterate or translate an English company label into Chinese. Below, we present the transliteration and translation datasets respectively, based on existing transliteration and translation datasets.

### 7.5.1 Transliteration Data

As we want to create a model that focuses on the transliteration of company labels, we create a dataset containing only transliterations, i.e., transferring a word from one script to another. Peoples' names are typically transliterated. E.g., the name *Ada Lovelace* is transliterated to 爱达·勒芙蕾丝 in Chinese. For the people names dataset, we reuse the dataset published by Merhav and Ash (2018). They published a dataset of 61,309 Chinese-English people name pairs based on Wikidata labels, with first and last names separated, i.e., one word per line.

### 7.5.2 Translation Data

For the translation datasets, we leverage available datasets commonly used for machine translation. The Fourth Conference on Machine Translation (WMT19) shared task publishes datasets every year, which we leverage. In particular, we make use of the News Commentary dataset (320k pairs) as well as the Wikititles dataset. Wikititles increase the performance of our model, as Wikipedia titles have a similar format to the company alias and labels in terms of token length, while news comments are long sentences.

### 7.5.3 Pinyin Data

We also test pinyin datasets in our training data. Pinyin is the official romanisation system for standard Chinese. The pinyin datasets are derived from two sources: an alignment of pinyin and Latin Characters for English[1] (called *pinyin characters dataset* from hereon) and *CEDICT*[2], inspired by the work of Li et al. (2018), a freely available English-Chinese dictionary including pinyin representation for each word which we leverage. We merge the CEDICT and pinyin characters datasets to one (the pinyin dataset). One of the limitations of the pinyin dataset is that pinyin alignments are not necessarily words transliterated from Chinese to English and vice versa, but they are a standardised way of romanising of the Chinese characters based on their pronunciation. Further, we encounter duplicates in the dataset. For example, there are multiple representations of *levi*. After merging the CEDICT and pinyin characters datasets, we remove those duplicates by matching all English words that have the same Chinese representation. If there are multiple Chinese representations of the same English word, we pick the first occurrence of the word, as we do not have a way to identify the best translation automatically. Without duplicates, the dataset consists of $145,953$ English-Chinese pairs.

For the experiments in Section 7.7, we test two datasets: the person names dataset, and the three datasources (person names, pinyin, and CEDICT) together in one dataset. The assumption to test is whether more data is beneficial to the learning of the model even if the data is noisy. However, following the work of Belinkov and Bisk (2018), who explore the fragility of NMT systems w.r.t. noisy data, we expect the pure name dataset to perform better, as pinyin introduces noise into the English to Chinese setup.

## 7.6 Evaluation and Models

We evaluate the results of the language models using standard evaluation metrics for machine translation, i.e., BLEU. Given the short tokens of our company domain, we report the 1-gram BLEU scores additionally to the 4-gram BLEU scores. Based on the work of Li and Specia (2019), who contributed to the WMT19 challenge with a model for Chinese-English news translation, we also look at the character error rate (CER). We further evaluate results with word error rate (WER) as per Merhav and Ash (2018), however, this metric's applicability to our problem is limited due to the low number of words in the company domain. On testing, for one label, we have multiple aliases to be checked against, e.g., *Apple* can be represented as 苹果公司 and 苹果. We do not aggregate these (label, alias) pairs, but treat them as different pairs. This can lead to lower final scores.

We implement the translation and transliteration models using JoeyNMT (Kreutzer et al., 2019). We choose a transformer model for the transliteration model, as Merhav and Ash (2018) show the transformer performs better than other models. The values of parameters follow the standard parameters of JoeyNMT's transformer setup and are only changed to adapt to the character-level model: 3 layers, 4 transformer heads, dropout rate is 0.1, label smoothing rate is 0.0, Adam optimizer, learning rate is 0.001 with NOAM decay, batch size 50. We follow the same implementation for the translation model, but use a word-level transformer using BPE (byte pair encoding) in the encoding as it is more appropriate for the translation task and increase the batch size to 100.

[1] https://github.com/hotoo/pinyin/blob/master/data/dict-zi-web.js
[2] https://www.mdbg.net/chinese/dictionary?page=cc-cedict

| | converted data | simplified data |
|---|---|---|
| **people** | 16.70 | 23.54 |
| **simplified** | 27.59 | 30.87 |
| **converted** | **29.58** | **30.89** |

TABLE 7.3: Experiment comparing training on the original people names dataset, the dataset with only simplified Chinese and the dataset converted to simplified Chinese, reported with BLEU-4. Overall the model converted to simplified Chinese performs best on both the dataset converted to simplified Chinese and the dataset only simplified Chinese.

## 7.7 Transliteration and Translation Model Setup

In the following, we explore the different setups for the translation and transliteration models as described in Section 7.6. We test how differentiating between simplified and traditional Chinese impacts the models performance (Section 7.7.1), the different setups for the transliteration model (Section 7.7.2) and for the translation model (Section 7.7.3).

### 7.7.1 Simplified Chinese

In the datasets we use for training (people names for transliteration and WMT dataset for translation), and for testing (company aliases from Wikidata), there is a mix of traditional and simplified characters. Simplified and traditional Chinese have two different sets of characters, which are used in different Chinese speaking regions. As our work focuses on simplified Chinese, this brings disadvantages as any language model has to predict into two different sets of characters without indication which character set is used in which case. For example, when training over all people names, the model would output 沃爾沃, which is *Volvo* in traditional Chinese, while the reference would state 沃尔沃, *Volvo* in simplified Chinese. We deterministically identify strings that have simplified Chinese characters. In the set of companies that was annotated as transliteration, 53.33% have simplified Chinese characters. In the people names dataset, 63.05% of names have simplified Chinese characters. In the WMT19 training split, only 68.31% of lines have simplified Chinese characters. This leads to a noisy dataset that (1) has a much bigger vocabulary size due to the combination of traditional and simplified characters and (2) more noisy data as the network has to not only detect how to translate a word but also which character set to use.

There are two options for dealing with this problem: (1) remove all non-simplified Chinese character pairs, which leads us to a smaller dataset (2) deterministically convert traditional to simplified Chinese characters. We create both datasets based on the original dataset: `simplified` removes all words, which do not have simplified characters. `converted` converts all words to simplified Chinese by the deterministically transferring characters from traditional to simplified Chinese.

We train two new transliteration models, one only on people names in simplified Chinese, one on the converted Chinese dataset, and compare them with the original results. As the numbers of words in the test datasets differ due to removing non-simplified words in the simplified dataset, we report the results here as BLEU-4 scores. Removing words with non-simplified

| | dev | test |
|---|---|---|
| **all-data** | 16.73 | 0.12 |
| **people only** | 22.48 | 0.16 |

TABLE 7.4: Experiment comparing running over the dataset including pinyin data and the person names dataset only. Training and validation on the original datasets, test on company names, evaluated with BLEU score.

Chinese characters improves results by 31% compared to training on the full dataset, as can be seen in Table 7.3; however, it decreases the dataset size. The deterministic conversion of characters from traditional to simplified Chinese improves the performance of the model further. The model using data converted to simplified Chinese outperforms all other setups. It achieves a 77% increase in BLEU-4 score when testing on the people names dataset that has been converted to simplified Chinese, compared to training on the original dataset.

Further, it achieves a 31% increase when testing on the people names dataset in which all non-simplified Chinese words have been removed.

### 7.7.2 Transliteration Model

To test our implementation of the transliteration model, we recreate the baseline of Merhav and Ash (2018) using our model setup described above. As the original paper provides a dataset for Chinese but does not include results for Chinese, we first recreated the languages they do report on with our model. As our goal is to translate from English to a target language; we only reproduce the experiments that use English as a source language. We achieve the same WER scores as the Merhav and Ash (2018) paper for Arabic (0.45), Hebrew (0.44), Katakana (0.52), and Russian (0.35), showing that our model implementation is competitive. While they provide the dataset, they do not report on the scores for Chinese. Using our character-level transformer model, we get a WER of 0.77 for Chinese - a 38.7% decrease in performance compared to the result for Katakana. We attribute this margin to the complexity of the language and the mix of simplified and traditional Chinese as described in Section 7.7.1.

**Pinyin and people names** To diversify the training data, we test the addition of pinyin data to the names dataset (see Section 7.5.3) for training. We find that the addition of Pinyin to the dataset seems to introduce noise, as can be seen in Table 7.4. As discussed previously, it can be assumed that this is due to the very different nature of the data. While pinyin can be helpful in other tasks in the field of machine translation (see, e.g., Liu et al. (2019)), the additional information cannot be leveraged by our model setup. Since the evaluation scores decrease with the addition of Pinyin datasets, we therefore focus on experimenting with the person names only.

### 7.7.3 Translation Model

For the translation model, we use a world-level transformer model using BPE in the preprocessing. The Chinese data was tokenised using jieba, a tokeniser for Chinese, following

| | Data / Model | **Translit. Model** | **Translat. Model** |
|---|---|---|---|
| BLEU | **Crowd Translit.** | 6.5/0.0 | **12.1/0.0** |
| | **Crowd Translat.** | 0.0/0.0 | **24.9/6.45** |
| | **All Companies** | 0.2/0.0 | **14.1/6.53** |
| CER | **Crowd Translit.** | **0.72** | 0.90 |
| | **Crowd Translat.** | 0.99 | **0.71** |
| | **All Companies** | 0.97 | **0.89** |
| WER | **Crowd Translit.** | **0.93** | 1.07 |
| | **Crowd Translat.** | **1.0** | 1.09 |
| | **All Companies** | **0.98** | 1.08 |

TABLE 7.5: Results for out-of-domain data, i.e., train on people names/WMT and test on the companies annotated by the crowd, BLEU-1/BLEU-4 (top), CER (middle), WER (bottom) scores.

work of Li and Specia (2019). BPE was used on the English-Chinese pairs. Training the model on WMT'19 news commentary and Wikititles performs best with a BLEU-4 score of 12.47. We tested with older versions of WMT datasets news commentary with and without Wikititles, with worse results.

## 7.8 Transliteration and Translation Based on Gold Standard

In Table 7.5 we show the results in terms of BLEU score, CER, and WER for the company test dataset, annotated as transliteration and translation respectively. For both training and test, we use the dataset converted to simplified Chinese as described in Section 7.7.1. We find that training on one domain and applying the data to another domain impacts the models' performance and therefore yields relatively low BLEU scores overall. This is due to the very different nature of the training and testing data. The translation model is only trained on long sentences, with the exception of the wikititles. The people names corpus does not have white spaces, as it contains only one word per line, differing strongly from company labels and aliases. In English, company entities have an average of 2.6 words per label. While in terms of BLEU score (higher is better), the translation model performs best over the gold standard for translation and transliteration, we find differences in the CER and WER scores (lower is better). The transliteration model has a better CER than the translation model on the transliterations, with a 25% increase, and the translation model has a better CER on the translations, with a 39.4% increase. This is promising in terms of differentiating between translation and transliteration. Further, the transliteration model increases by 34.7% and the translation model by 25.4% on their respective tasks compared to their performance over all company (label, alias) pairs without differentiating translation and translation (*All Companies* in Table 7.5).

| | Data / Model | **Translit. Model** | **Translat. Model** |
|---|---|---|---|
| BLEU | **Translit.** | **7.1/0.0** | 13.5/0.0 |
| | **Translat.** | 0.5/00 | **22.5/5.0** |
| CER | **Translit.** | **0.81** | **0.81** |
| | **Translat.** | 0.99 | **0.76** |
| WER | **Translit.** | **0.93** | **1.11** |
| | **Translat.** | 1.0 | 1.17 |

TABLE 7.6: Results for aliases generated after automated classification as transliteration or translation, BLEU-1/BLEU-4 (top), CER (middle), WER (bottom) scores. The translation and transliteration models perform better on the entities classified as translation and transliteration respectively, while both models perform better with the classification than the companies overall.

## 7.9 Transliteration and Translation Based on Automated Classification

In the previous sections, we explored each step of generating a Chinese alias from an English label. In Figure 7.2, we describe the whole approach. We explored the classifier in Section 7.4, in which we use a knowledge graph embeddings-based classifier to differentiate between transliteration and translation. Future work should investigate how to improve features and learn which triples indicate such differences. We show in Section 7.7, that the training on out-of-domain data is possible, but the short tokens of company labels and aliases and the one-to-many mapping of labels to aliases indicate that there is room for improvement.

These limitations in each step of our approach are evident also when we generate Chinese aliases from labels using the knowledge graph embeddings-based classifier. Table 7.6 shows that the overall scores are lower than the results over the gold standard as annotated by the crowd in Table 7.5. This is mainly a consequence of the classification step, in which transliteration or translation is decided based on the knowledge graph embeddings-based classifier. As we observed in the previous experiments in Section 7.8, the translation model can to an extent handle transliterations; the transliteration model, however, has no way of dealing with translations. This is another factor in favour of the translation model overall. We show in Table 7.6 that, using the knowledge graph embeddings-based classifier, the transliteration model outperforms the translation model by 42% in terms of BLEU-1 score on entities' labels classified as transliterations, while the translation model has a 66% higher BLEU-1 score on translations than the transliteration model. The translation model performs well overall, presumably due to the presence of transliterations in the translation training dataset. I.e., CER scores are equal for the transliterations for both translation and transliteration model.

## 7.10 Discussion

We showed the lack of multilingual knowledge graph data in Chapters 4 and 5. In this chapter we have explored the generation of Chinese aliases from English labels by differentiating between transliteration and translation in the company domain on Wikidata, with a view to developing practical solutions to this problem of data inequality between languages in multi-language knowledge graphs. We find that our approach can be a starting point for further exploration of the topic, and list below the contributions and results we contribute to each part of the pipeline. In the next chapter, Chapter 8, we show a use case for using multilingual labels for article creation on Wikipedia. Approaches to generating articles in lower resourced languages improve with a greater number of multilingual labels being available; therefore, the creation of new labels is of importance.

**Translations and transliterations of company labels and aliases.** Company aliases can be translated, transliterated, or a mix of both. For example, "Bloomberg Limited Partnership" in Chinese is "彭博 有限合夥企", where *Bloomberg* is transliterated and *Limited Partnership* is translated. We designed a crowdsourcing task to understand the distributions of translations, transliterations, and partial translations/transliterations of company labels/aliases and report on the results in Section 7.3. Further, we provide an outlook on company labels and aliases that are partially translated/transliterated and investigate patterns we find in the data that can be used in future work to facilitate the translation and transliteration process. We use this information for classification in Section 7.4.

**Translation and transliteration using out-of-domain data.** The number of companies in a knowledge graph is limited, and translation of companies can be very domain specific, making existing translation data in this domain sparse. We explore the translation and transliteration of company labels using out-of-domain data, which has significantly different properties from those of company labels and aliases, and show that while there is a gap between the training and test datasets, we can use out-of-domain to create new aliases in the company domain.

**Differentiating translation and transliteration.** Previous work on knowledge graph label generation has focused on either transliteration (Merhav and Ash, 2018) or translation (Moussallem et al., 2019). We combine previous work by studying the classification of an entity as transliteration or translation before generating a Chinese alias. We explore automatically classifying (label, alias) pairs as transliteration or translation in Section 7.4. As we have contextual information about an entity in the form of its triples (such as locations, relationships, properties, etc.), we want to understand whether enriching a classification with this information yields improved translation and transliteration results. We test employing rule mining-based classifier and knowledge graph embeddings as a base for the classification (Wang et al., 2017). We show that both fastText and BigGraph can be used to classify entities for translation or transliteration. Further work will be needed to explore whether certain properties of the entities have a larger impact on the classification task. When testing the translation and transliteration models based on the classification on knowledge graph embeddings in Section 7.9, the transliteration model has a 42% higher BLEU-1 score than the translation model on transliterations, and the translation model has a 66% higher BLEU-1 score than the transliteration model on translations. Both models perform better in their respective tasks than over all company labels without classification as transliteration or translation: 25% and 39.5% in terms of character error rate (CER) for the transliteration and the translation model respectively.

**Simplified versus traditional Chinese.** Chinese is the most spoken language in the world and research in the machine translation community reflects this, e.g., through the integration of Chinese in the shared task of the conference on machine translation (WMT) (Barrault et al., 2019). However, a differentiation between simplified and traditional Chinese is usually not made. This mix of two character sets impacts the performance of our task. We show in Section 7.7.1 that deterministic conversion of training and test data to simplified Chinese in the people domain for the transliteration model can increase the BLEU-4 score by up to 77%.

# Chapter 8

# ArticlePlaceholder from Wikidata for Wikipedia

In Chapters 4 and 5 we show that the distribution of multilingual information in knowledge graphs is more diverse than the distribution of website languages. In Chapter 7 we show a way of generating more aliases in lower-resourced languages. Wikipedia has a lack of multilingual information, whereas such information is abundant in Wikidata. Wikipedia is available in 301 languages, but its content is unevenly distributed (Hecht and Gergle, 2010). Language versions with less coverage face multiple challenges: fewer editors means less quality control, making that particular Wikipedia less attractive for readers in that language, which in turn makes it more difficult to recruit new editors from among the readers.

In this chapter we therefore propose a way of integrating the multilingual information of Wikidata into Wikipedia, as a use case for multilingual knowledge graph labels. We detail the language coverage of Wikidata in Chapter 5, showing that information on a large scale is available across languages. This, along with its connection to Wikipedia (see Section 2.6.2), makes it an ideal candidate for making more information available to readers of Wikipedia.

The ArticlePlaceholder implements this idea. Currently used in 14 Wikipedias (see Section 8.1.1), the ArticlePlaceholder takes advantage of Wikidata's multilingual capabilities to increase the coverage of under-resourced Wikipedias (Kaffee, 2016). When someone looks for a topic that is not yet covered by Wikipedia in their language, the ArticlePlaceholder tries to match the topic with an entity in Wikidata. If successful, it then redirects the search to an automatically generated *placeholder page* that displays the relevant information, for example the name of the entity and its main properties, in their language.

In Kaffee et al. (2018b) we propose propose an iteration of the ArticlePlaceholder to facilitate the representation of the data on the placeholder page. The original version of the tool pulled the raw data from Wikidata (available as triples with labels in different languages) and displayed it in tabular form (see Figure 8.1 in Section 8.1). In the current version, we use Natural Language Generation (NLG) techniques to automatically produce one summary sentence from the triples instead. Presenting structured data as text rather than tables helps people uninitiated in the relevant technologies to make sense of it (Vougiouklis et al., 2018).

This is particularly useful in contexts where one cannot make any assumptions about the levels of data literacy of the audience, as it is the case for a large share of Wikipedia readers.

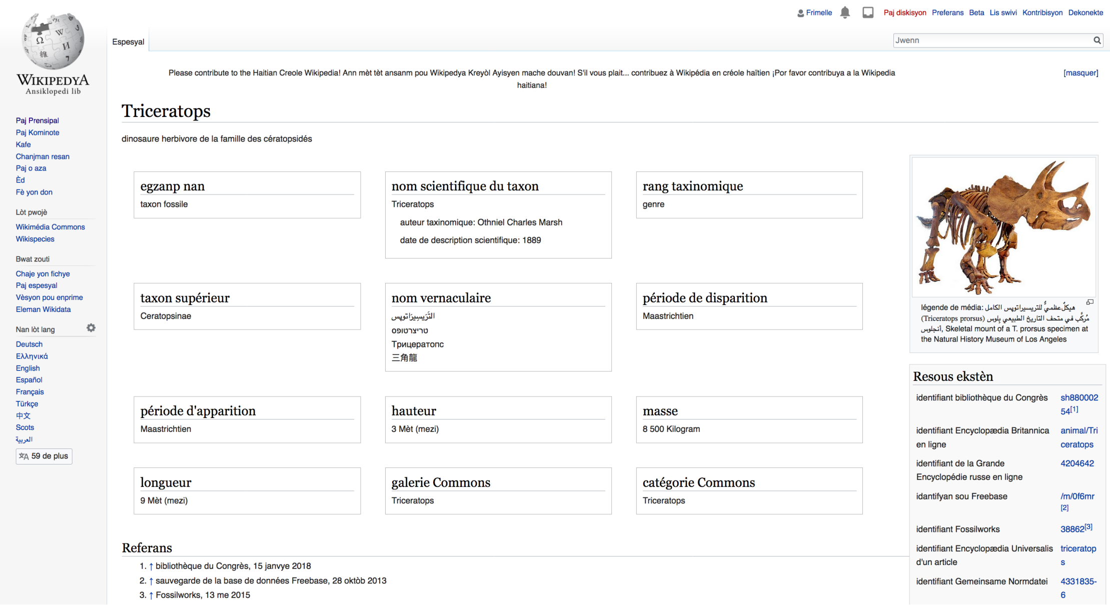
In the studies presented in this chapter, we pursue the following research questions based on the overall research question **RQ4 How do Wikipedia editors perceive automatically generated Wikipedia summaries?** introduced in Chapter 1. We showed in our previous study in Kaffee et al. (2018b) that we can train a neural network to generate text from triples in a multilingual setting. In this work we focus on how Wikipedia readers and editors perceive and use those sentences, and therefore limit the description of the natural language generation model to referencing the work in Kaffee et al. (2018b).

**RQ4.1 What is the quality of neural generated text from triples in a multilingual setting?** To answer this question, we undertook a quantitative study with participants from two different Wikipedia language communities (Arabic and Esperanto), who were asked to assess, from a reader's perspective, whether the text is fluent and appropriate for Wikipedia. In previous work (Kaffee et al., 2018b), we showed that the performance of the model used for the generation of sentences is competitive in terms of automated metrics, compared to a set of baselines, across languages and domains. The study to answer this research question is described in Section 8.3.

**RQ4.2 How do editors perceive the generated text on the ArticlePlaceholder page?** To add depth to the quantitative findings of the first study, we undertook a second, mixed-methods study within six Wikipedia language communities (Arabic, Swedish, Hebrew, Persian, Indonesian, and Ukrainian). We carried out semi-structured interviews in which we asked editors to comment on their experience with reading the summaries generated through our approach, and we identified common themes in their answers. Among other things, we were interested to understand how editors perceive text that is created by the artificial intelligence (AI) algorithm rather than being manually crafted, and how they deal with so-called <rare> tokens in the sentences. These tokens represent realisations of infrequent entities in the text, which data-driven approaches generally struggle to verbalise (Luong et al., 2015). The study to answer this research question is described in Section 8.4.

**RQ4.3 How do editors use the generated sentence in their work?** As part of the second study, we also asked participants to edit the placeholder page, starting from the automatically generated text or removing it completely. We assessed text reuse both quantitatively, using a string-matching metric, and qualitatively through the interviews. Just like in *RQ*4.2, we were also interested to understand whether summaries with <rare> tokens, which point to limitations in the algorithm, would be used when editing, and how the editors would work around the tokens. The study to answer this research question is described in Section 8.4.

The evaluation helps us build a better understanding of the tools and experience we need to help nurture under-served Wikipedias. Our quantitative analysis of the reading experience showed that participants rank the summary sentences close to the expected quality standards in Wikipedia, and are likely to consider them as part of Wikipedia. This was confirmed by the interviews with editors, which suggested that people believe the summaries to come from a Wikimedia-internal source. According to the editors, the new format of the ArticlePlaceholder



Please contribute to the Haitian Creole Wikipedia! Ann mèt tèt ansanm pou Wikipedya Kreyòl Ayisyen mache douvan! S'il vous plait... contribuez à Wikipédia en créole haïtien ¡Por favor contribuya a la Wikipedia haitiana!

Triceratops

dinosaure herbivore de la famille des cératopsidés

egzanp nan
taxon fossile

nom scientifique du taxon
Triceratops
auteur taxinomique: Othniel Charles Marsh
date de description scientifique: 1889

rang taxinomique
genre

taxon supérieur
Ceratopsinae

nom vernaculaire
التَّرَيسِيرَاتوبس
טריצרטופס
Трицератопс
三角龍

période de disparition
Maastrichtien

période d'apparition
Maastrichtien

hauteur
3 Mèt (mezi)

masse
8 500 Kilogram

longueur
9 Mèt (mezi)

galerie Commons
Triceratops

catégorie Commons
Triceratops

légende de média: هيكلٌ عظميٌّ للتريسيراتوبس الكامل (Triceratops prorsus) مُركَّب في متحف التاريخ الطبيعي بلوس أنجلوس, Skeletal mount of a T. prorsus specimen at the Natural History Museum of Los Angeles

Resous ekstèn

| | |
|---|---|
| identifiant bibliothèque du Congrès | sh88000254[1] |
| identifiant Encyclopædia Britannica en ligne | animal/Triceratops |
| identifiant de la Grande Encyclopédie russe en ligne | 4204642 |
| idantifyan sou Freebase | /m/0f6mr [2] |
| identifiant Fossilworks | 38862[3] |
| identifiant Encyclopædia Universalis d'un article | triceratops |
| identifiant Gemeinsame Normdatei | 4331835-6 |

Referans

1. ↑ bibliothèque du Congrès, 15 janvye 2018
2. ↑ sauvegarde de la base de données Freebase, 28 oktòb 2013
3. ↑ Fossilworks, 13 me 2015

FIGURE 8.1: Example page of the ArticlePlaceholder as deployed now on 14 Wikipedias. This example contains information from Wikidata on Triceratops in Haitian-Creole.

enhances the reading experience: people tend to look for specific bits of information when accessing a Wikipedia page, and the compact nature of the generated text supports that. In addition, the text seems to be a useful starting point for further editing, and editors reuse a large portion of it even when it includes <rare> tokens.

We believe the two studies (described in Section 8.3 and Section 8.4 respectively) could also help advance the state of the art in two other areas: together, they propose a user-centred methodology to evaluate NLG, which complements automatic approaches based on standard metrics and baselines, the norm in most papers; at the same time, they also shed light on the emerging area of human-AI interaction in the context of NLG. While the editors worked their way around the <rare> tokens both during reading and writing, they did not check the text for correctness, nor query where the text came from or what the tokens meant. This suggests that we need more research into how to communicate the provenance of content in Wikipedia, especially in the context of automatic content generation and deep fakes (Isola et al., 2017), as well as algorithmic transparency.

## 8.1 Bootstrapping empty Wikipedia articles

The overall aim of our system is to give editors access to information that is not yet covered in Wikipedia, but is available in the relevant language in Wikidata. The system is built on the ArticlePlaceholder that displays Wikidata triples dynamically on different language Wikipedias. In this chapter we extend the ArticlePlaceholder with an NLG component that generates an introductory sentence on each ArticlePlaceholder page in the target language from Wikidata triples.

| Page stats | Esperanto | Arabic | English | Wikidata |
|---|---|---|---|---|
| Articles | 241,901 | 541,166 | 5,483,928 | 37,703,807 |
| Average number of edits/page | 11.48 | 8.94 | 21.11 | 14.66 |
| Active users | 2,849 | 7,818 | 129,237 | 17,583 |
| Vocabulary size | 1.5M | 2.2M | 2M | – |

TABLE 8.1: Page statistics and number of unique words (vocabulary size) of Esperanto, Arabic and English Wikipedias in comparison with Wikidata. Retrieved 27 September 2017. Active users are registered users that have performed an action in the last 30 days.

### 8.1.1 ArticlePlaceholder

As discussed earlier, some Wikipedias suffer from a lack of content, which means fewer readers, and in turn, fewer potential editors. The idea of the ArticlePlaceholder is to use Wikidata, which contains about 55 million entities (by comparison, the English Wikipedia covers around 5 million topics), often in different languages, to bootstrap articles in language versions lacking content.

ArticlePlaceholders are pages on Wikipedia that are dynamically drawn from Wikidata triples. When the information in Wikidata changes, the ArticlePlaceholder pages are automatically updated. In the original release, the pages display the triples in a tabular way, purposely not reusing the design of a standard Wikipedia page to make the reader aware that the page was automatically generated and requires further attention. An example of the interface can be seen in Figure 8.1.

The Article Placeholder is deployed on 14 Wikipedias with a median of 69,623.5 articles, between 253,539 (Esperanto) and 7,464 (Northern Sami).

### 8.1.2 Text generation

We use a data-driven approach that allows us to extend the ArticlePlaceholder pages with a short description of the article's topic. We reuse the encoder-decoder architecture introduced in previous work of ours in Vougiouklis et al. (2018), which was focused on a closed-domain text generative task for English. Details about the model setup can be found in Kaffee et al. (2018b), including the *property placeholders*, which aim at addressing the problem of out-of-vocabulary words, based on the work of Vougiouklis et al. (2018). In case a rare entity in the text is not matched to any of the input triples, its realisation is replaced by the special `<rare>` token. An overview of the model is displayed in Figure 8.2.

In order to train and evaluate our system, we created a dataset for text generation from knowledge base triples in two languages based on the work of ElSahar et al. (2018). We used two language versions of Wikipedia which differ in terms of size (see Table 8.1) and language support in Wikidata (see Chapter 5). The dataset aligns Wikidata triples about an item with the first sentence of the Wikipedia article about that entity. More details about the dataset, training, testing, and evaluation of the model can be found in Kaffee et al. (2018a).

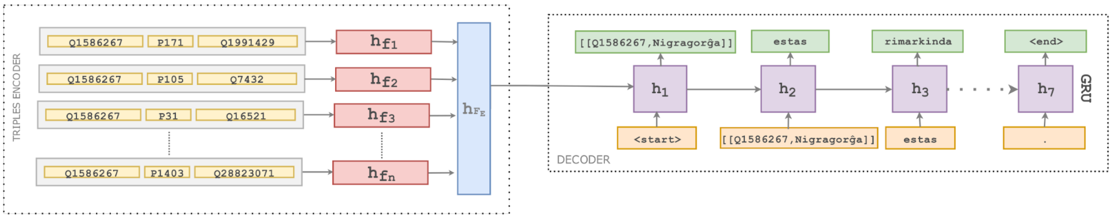


FIGURE 8.2: Representation of the neural network architecture. The triple encoder computes a vector representation for each one of the three input triples from the ArticlePlaceholder, $h_{f_1}$, $h_{f_2}$ and $h_{f_3}$. Subsequently, the decoder is initialised using the concatenation of the three vectors, $[h_{f_1}; h_{f_2}; h_{f_3}]$. The purple boxes represent the tokens of the generated text. Each snippet starts and ends with special tokens: start-of-summary `<start>` and end-of-summary `<end>`. Example is in Esperanto.

| | Data | Method | Participants |
|---|---|---|---|
| RQ4.1 | Survey answers | Judgement-based evaluation, quantitative | Readers of two Wikipedias |
| RQ4.2 | Interview answers | Task-based evaluation, qualitative (thematic analysis) | Editors of six Wikipedias |
| RQ4.3 | Interview answers and text reuse metrics | Task-based evaluation, qualitative (thematic analysis) and quantitative | Editors of six Wikipedias |

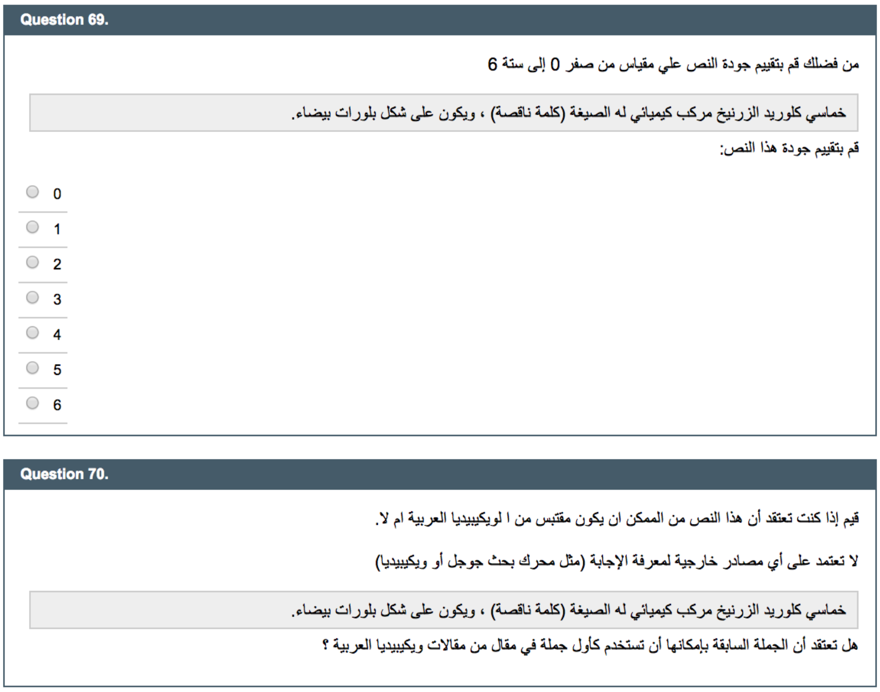
TABLE 8.2: Evaluation methodology

Using the same language Wikipedia to train makes it possible for the network to pick up the community-specific language and possibly also their different approach to a topic, debiasing the model compared to pure translation from an English Wikipedia article, which preserves the English Wikipedia way of phrasing and view-point.

## 8.2 Methods for Answering the Research Questions

We followed a mixed-methods approach to investigate the three questions discussed in the introduction (Table 8.2). To answer *RQ*4.1, we used a judgement-based quantitative evaluation with readers in two language Wikipedias who are native (or fluent, in the case of Esperanto) speakers of the language[1]. We showed them text generated through our approach, as well as genuine Wikipedia sentences and news snippets of similar length, and asked them to rate their fluency and appropriateness for Wikipedia on a scale. To answer *RQ*4.2 and *RQ*4.3, we carried out an interview study with editors of six Wikipedias, with qualitative and quantitative components. We instructed editors to complete a reading and an editing task and to describe their experiences along a series of questions. We used thematic analysis to identify common themes in the answers. For the editing task, we also used a quantitative element in the form of a text reuse metric, which is described below.

[1] The raw data of the quantitative evaluation experiments can be found here: `https://github.com/pvougiou/Mind-the-Language-Gap/tree/master/crowdevaluation`.



**Question 69.**

من فضلك قم بتقييم جودة النص علي مقياس من صفر 0 إلى ستة 6

خماسي كلوريد الزرنيخ مركب كيميائي له الصيغة (كلمة ناقصة) ، ويكون على شكل بلورات بيضاء.

قم بتقييم جودة هذا النص:

○ 0
○ 1
○ 2
○ 3
○ 4
○ 5
○ 6

**Question 70.**

قيم إذا كنت تعتقد أن هذا النص من الممكن ان يكون مقتبس من ا لويكيبيديا العربية ام لا.

لا تعتمد على أي مصادر خارجية لمعرفة الإجابة (مثل محرك بحث جوجل أو ويكيبيديا)

خماسي كلوريد الزرنيخ مركب كيميائي له الصيغة (كلمة ناقصة) ، ويكون على شكل بلورات بيضاء.

هل تعتقد أن الجملة السابقة بإمكانها أن تستخدم كأول جملة في مقال من مقالات ويكيبيديا العربية ؟

FIGURE 8.3: Example of a question to readers about quality and appropriateness for Wikipedia of the generated summaries in Arabic. They see this page after the instructions are displayed. First, the user is asked to evaluate the quality from 0 to 6 (Question 69), then they are asked whether the sentence could be part of Wikipedia (Question 70). The sentence to evaluate has a grey background. English translation in Figure 8.4

## 8.3 RQ4.1 - Judgement-based evaluation

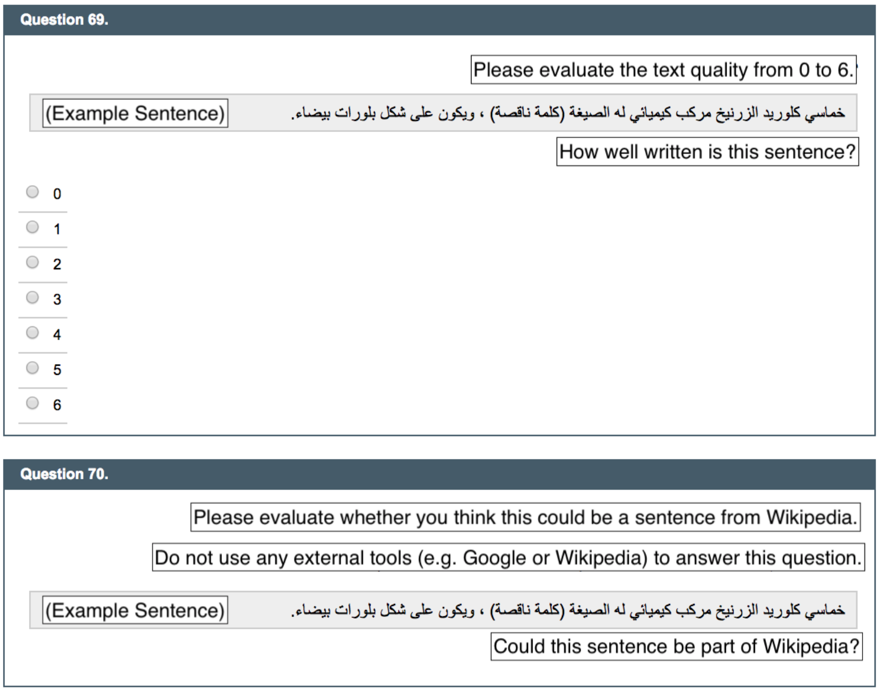
We defined quality in terms of text fluency and appropriateness, where fluency refers to how understandable and grammatically correct a text is, and appropriateness captures the extent to which the text 'feels like' Wikipedia content. We asked two sets of participants from two different language Wikipedias to assess the same summary sentences on a scale according to these two metrics.

Participants were asked to fill out a survey combining fluency and appropriateness questions. An example question can be found in Figure 8.3.

### 8.3.1 Recruitment

Our study targets any speaker of Arabic and Esperanto who reads that particular Wikipedia, independent of their contributions to Wikipedia. We wanted to reach fluent speakers of each language who use Wikipedia and are familiar with it even if they do not edit it frequently. For Arabic, we reached out to Arabic-speaking researchers from research groups working on Wikipedia-related topics. For Esperanto, as there are fewer speakers and they are harder to reach, we promoted the study on social media such as Twitter and Reddit[2] using the

[2] https://www.reddit.com/r/Esperanto/comments/75rytb/help_in_a_study_using_ai_to_create_esperanto/



**Question 69.**

Please evaluate the text quality from 0 to 6.

(Example Sentence) خماسي كلوريد الزرنيخ مركب كيميائي له الصيغة (كلمة ناقصة) ، ويكون على شكل بلورات بيضاء.

How well written is this sentence?

○ 0
○ 1
○ 2
○ 3
○ 4
○ 5
○ 6

**Question 70.**

Please evaluate whether you think this could be a sentence from Wikipedia.

Do not use any external tools (e.g. Google or Wikipedia) to answer this question.

(Example Sentence) خماسي كلوريد الزرنيخ مركب كيميائي له الصيغة (كلمة ناقصة) ، ويكون على شكل بلورات بيضاء.

Could this sentence be part of Wikipedia?

FIGURE 8.4: English translation of Figure 8.3

researchers' accounts. The survey instructions and announcements were translated into Arabic and Esperanto.[3] The survey was open for 15 days in September 2017.

| | | #P | #S | #P: S>50% | Avg #S/P | Median #S/P | All Ann. |
|---|---|---|---|---|---|---|---|
| Arab. | Fluency | 27 | 60 | 5 | 15.03 | 5 | 406 |
| | Appropriateness | 27 | 60 | 5 | 14.78 | 4 | 399 |
| Esper. | Fluency | 27 | 60 | 3 | 8.7 | 1 | 235 |
| | Appropriateness | 27 | 60 | 3 | 8.63 | 1 | 233 |

TABLE 8.3: Judgement-based evaluation: total number of participants (*P*), total number of sentences (*S*), number of participants who evaluated at least 50% of the sentences, average and mean number of sentences evaluated per participant, and number of total annotations

### 8.3.2 Participants

We recruited a total of 54 participants (see Table 8.3). Coincidentally, 27 of them were from each language community.

#### 8.3.2.1 Ethics

The research was approved by the Ethics Committee of the University of Southampton under ERGO Number 30452.

[3] https://github.com/luciekaffee/Announcements

### 8.3.3 Data

For both languages, we created a corpus consisting of 60 summaries of which 30 are generated through our approach, 15 are from news, and 15 from Wikipedia sentences used to train the neural network model. For news in Esperanto, we chose introductory sentences to articles in the Esperanto version of Le Monde Diplomatique[4]. For news in Arabic, we did the same using the RSS feed of BBC Arabic[5].

### 8.3.4 Metrics

Each participant was asked to assess the *fluency* of 60 sentences on a scale from 0 to 6 as follows:

- **(6)** No grammatical flaws and the content can be understood with ease
- **(5)** Comprehensible and grammatically correct summary that reads a bit artificial
- **(4)** Comprehensible summary with minor grammatical errors
- **(3)** Understandable, but has grammatical issues
- **(2)** Barely understandable summary with significant grammatical errors
- **(1)** Incomprehensible summary, but a general theme can be understood
- **(0)** Incomprehensible summary

For each sentence, we calculated the mean fluency given by all participants and then averaged over all summaries of each category.

To assess the appropriateness, participants were asked to assess whether the displayed sentence could be part of a Wikipedia article. We tested whether a reader can tell from just one sentence whether a text is appropriate for Wikipedia, using the news sentences as a baseline. This gave us an insight into whether the text produced by the neural network "feels" like Wikipedia text. Participants were asked not to use any external tools for this task and had to give a binary answer. Similarly to fluency, average appropriateness is calculated by averaging the corresponding scores of each summary across all annotators.

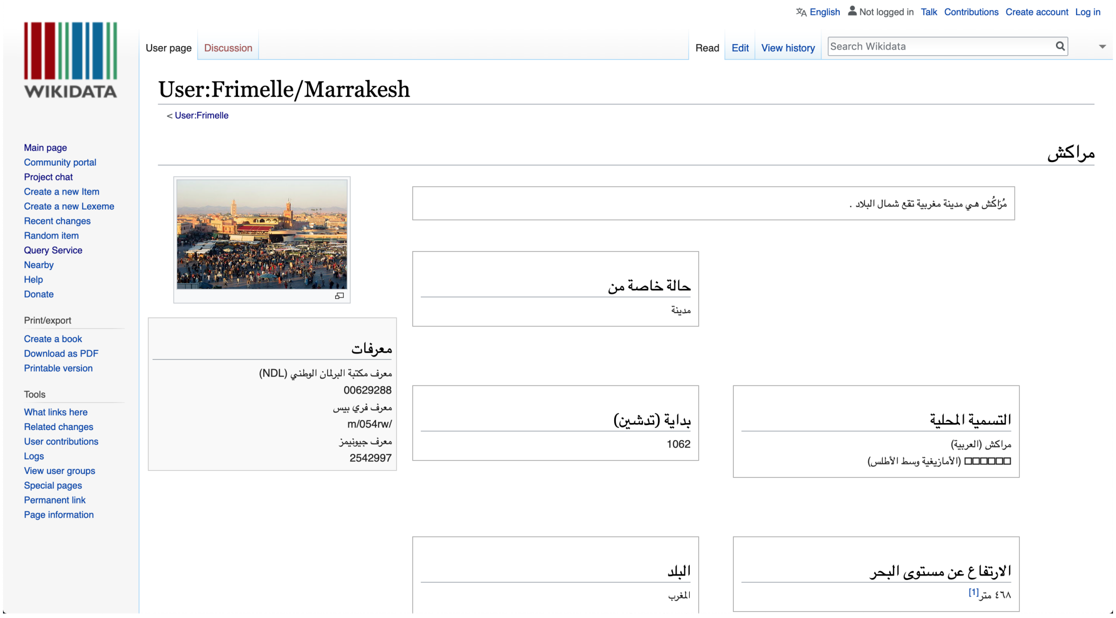
### 8.3.5 Results: RQ4.1 - Judgement-based evaluation

#### 8.3.5.1 Fluency

As shown in Table 8.5, overall, the quality of the generated text is high (4.7 points out of 6 in average in Arabic and 4.5 in Esperanto). In Arabic, 63.3% of the summaries scored as much as 5 in fluency on average. In Esperanto, which had the smaller training corpus, the participants nevertheless gave as many as half of the snippets an average of 5, with 33% of them reaching a

[4] `http://eo.mondediplo.com/`, accessed on the 28th of September, 2017

[5] `http://feeds.bbci.co.uk/arabic/middleeast/rss.xml`, accessed on the 28th of September, 2017



User:Frimelle/Marrakesh

< User:Frimelle

مراكش

مُرّاكُش هي مدينة مغربية تقع شمال البلاد .

حالة خاصة من

مدينة

معرفات

معرف مكتبة البرلمان الوطني (NDL)
00629288
معرف فري بيس
m/054rw/
معرف جيونيمز
2542997

بداية (تدشين)

1062

التسمية المحلية

مراكش (العربية)
□□□□□□ (الأمازيغية وسط الأطلس)

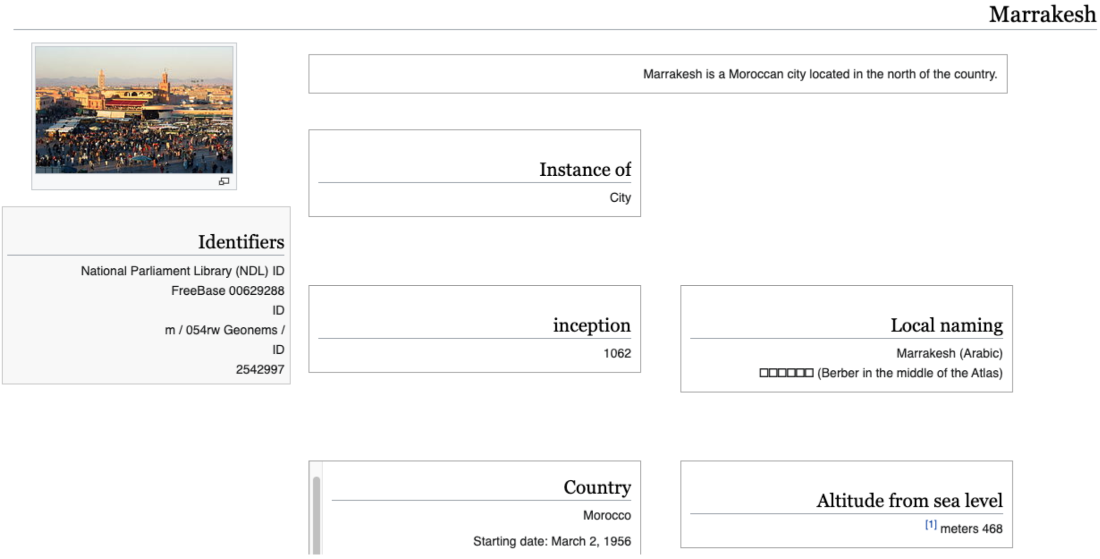
البلد

المغرب

الارتفاع عن مستوى البحر

٤٦٨ متر[1]

FIGURE 8.5: Screenshot of the reading task. The page is stored as a subpage of the author's user page on Wikidata, therefore the layout copies the original layout of any Wikipedia. The layout of the information displayed mirrors the ArticlePlaceholder setup. The participants sees the sentence to evaluate alongside information included from the Wikidata triples (such as the image and statements) in their native language (Arabic in this example).


Marrakesh

Marrakesh is a Moroccan city located in the north of the country.

Instance of

City

Identifiers

National Parliament Library (NDL) ID
FreeBase 00629288
ID
m / 054rw Geonems /
ID
2542997

inception

1062

Local naming

Marrakesh (Arabic)
□□□□□□ (Berber in the middle of the Atlas)

Country

Morocco
Starting date: March 2, 1956

Altitude from sea level

[1] meters 468

FIGURE 8.6: Screenshot of the reading task as in Figure 8.5, translated to English.

| Language | # Articles | Active Editors | Native Speakers |
|---|---:|---:|---:|
| Arabic | 541,166 | 5,398 | 280 |
| Swedish | 3,763,584 | 2,669 | 8.7 |
| Hebrew | 231,815 | 2,752 | 8 |
| Persian | 643,635 | 4,409 | 45 |
| Indonesian | 440,948 | 2,462 | 42.8 |
| Ukrainian | 830,941 | 2,755 | 30 |

TABLE 8.4: Number of Wikipedia articles; active editors on Wikipedia (editors that performed at least one edit in the last 30 days); and number of native speakers in Million for each language.

| | | Fluency | | Appropriateness | |
|---|---|---|---|---|---|
| | | Mean | SD | Part of Wikipedia | |
| Arabic | Ours | 4.7 | 1.2 | 77% | 🟪 |
| | Wikipedia | 4.6 | 0.9 | 74% | 🟩 |
| | News | 5.3 | 0.4 | 35% | 🟦 |
| Esper. | Ours | 4.5 | 1.5 | 69% | 🟪 |
| | Wikipedia | 4.9 | 1.2 | 84% | 🟩 |
| | News | 4.2 | 1.2 | 52% | 🟦 |

TABLE 8.5: Results for fluency and appropriateness.

6 by all participants. We concluded that, in most cases, the text we generated was very understandable and grammatically correct. In addition, the results were perceived to match the quality of writing in Wikipedia and news reporting.

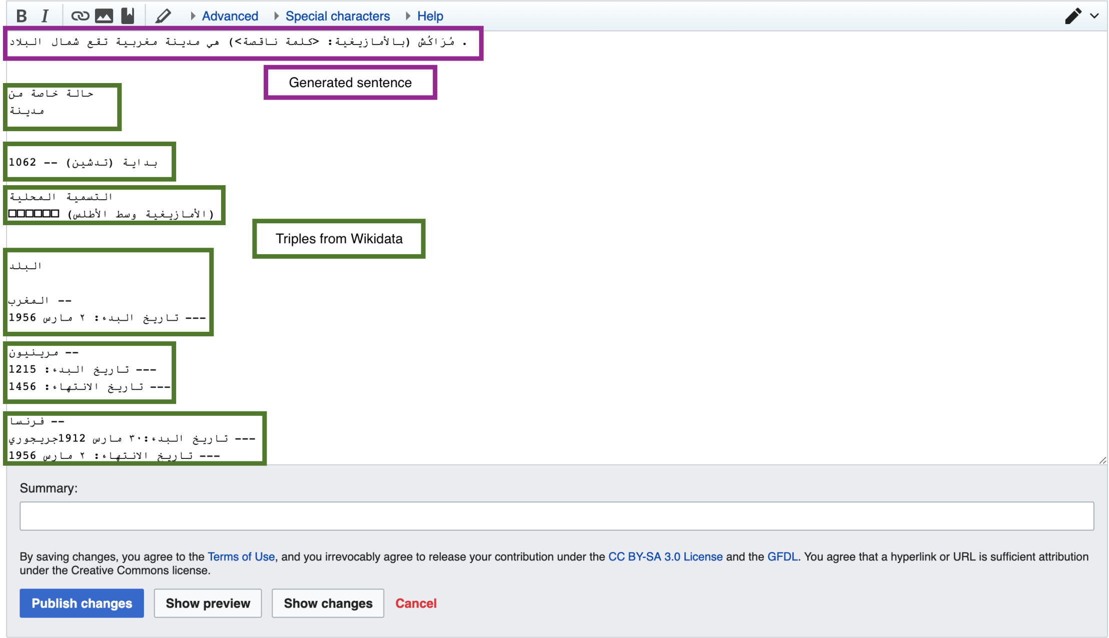
#### 8.3.5.2 Appropriateness

The results for appropriateness are summarised in Table 8.5. A majority of the snippets were considered to be part of Wikipedia (77% for Arabic and 69% in Esperanto). The participants confirmed that they seemed to identify a certain style and manner of writing with Wikipedia. By comparison, only 35% of the Arabic news snippets and 52% of the Esperanto ones could have passed as Wikipedia content in the study. Genuine Wikipedia text was recognised as such (in 77% and 84% of the cases, respectively).

Our model was able to generate text that is not only accurate from a writing point of view, but that, in a high number of cases, felt like Wikipedia and could blend in with other Wikipedia content.

## 8.4 RQ4.2 and RQ4.3 - Task-based evaluation

We ran a series of semi-structured interviews with editors of six Wikipedias to get an in-depth understanding of their experience with reading and using the automatically generated text.



مُرَاكُش (بالأمازيغية: <كلمة ناقصة>) هي مدينة مغربية تقع شمال البلاد .

حالة خاصة من
مدينة

بداية (تدشين) -- 1062

التسمية المحلية
(الأمازيغية وسط الأطلس) □□□□□□

البلد

-- المغرب
--- تاريخ البدء: ٢ مارس 1956

-- مرينيون
--- تاريخ البدء: 1215
--- تاريخ الانتهاء: 1456

-- فرنسا
--- تاريخ البدء: ٣٠ مارس 1912جريجوري
--- تاريخ الانتهاء: ٢ مارس 1956

By saving changes, you agree to the Terms of Use, and you irrevocably agree to release your contribution under the CC BY-SA 3.0 License and the GFDL. You agree that a hyperlink or URL is sufficient attribution under the Creative Commons license.

FIGURE 8.7: Screenshot of the editing task, with annotations. The page is stored on a subpage of the author's user page on Wikidata, so the layout is equivalent to the current MediaWiki installations on Wikipedia. The participants see the sentence (in green) that they saw before in the reading task, and the triples from Wikidata in their native language (Arabic in this example, in purple). The triples are manually added to the page by the researchers for easier interaction with the data by the editor. The data is the same data as in the reading task (Figure 8.5).

Each interview started with general questions about the participant's experience with Wikipedia and Wikidata, and their understanding of different aspects of these projects. The participants were then asked to open and read an ArticlePlaceholder page that included text generated through the NLG algorithm, as shown in Figure 8.5. Finally, participants were asked to edit the content of a page that contained the same information as the one they had to read earlier, but was displayed as plain text in the Wikipedia edit field. The editing field can be seen in Figure 8.7.

### 8.4.1 Recruitment

The goal was to recruit a set of editors from different language backgrounds, in order to gain a wide ranging understanding of different language communities. We reached out to editors of different Wikipedia editor mailing lists[6] and tweeted a call for participants using the lead author's account[7].

We were in contact with 18 editors from different Wikipedia communities. We allowed all editors to participate, but had to exclude editors who edit only on English Wikipedia (as it is outside our use-case) and editors who did not speak a sufficient level of English to participate in the interview.

[6] Mailing lists contacted: wikiar-l@lists.wikimedia.org (Arabic), wikieo-l@lists.wikimedia.org (Esperanto), wikifa-l@lists.wikimedia.org (Persian), Wikidata mailing list, Wikimedia Research mailing list

[7] https://twitter.com/frimelle/status/1031953683263750144

| Language | Sentence displayed to participants | # Participants | Participant |
|---|---|---|---|
| Arabic (ar) | مُرَاكُش (بالأمازيغية: <rare>) هي مدينة مغربية تقع شمال البلاد. | 4 | P1, P4, P8, P9 |
| Swedish (sv) | Marrakech (arabiska <rare>, Berberspråk <rare>) är en stad i sydvästra Marocko, vid foten av <rare>. | 2 | P5, P6 |
| Hebrew (he) | מרקש (בערבית: <rare>) היא עיר מדברית בדרום מערב מרוקו למרגלות הרי <rare>. | 1 | P3 |
| Persian (fa) | شهر مراکش (به بربری: <rare>) یکی از شهرهای کشور مراکش و مرکز استان مراکش <rare> است. | 1 | P2 |
| Indonesian (id) | Marrakesh (Arab: <rare>) ialah kota di barat daya Maroko di kaki <rare>. | 1 | P7 |
| Ukrainian (uk) | Марракеш (араб. <rare>) — важливе імперське місто в Марокко, розташованого біля підніжжя гір <rare>. | 1 | P10 |

TABLE 8.6: Overview of sentences and number of participants in each language

### 8.4.2 Participants

Our sample consists of 10 Wikipedia editors of different lower-resourced languages (measured by their number of articles compared to English Wikipedia). We originally conducted interviews with 11 editors from seven different language communities, but had to remove one interview with an editor of the Breton Wikipedia, as we were not able to generate the text for the reading and editing tasks because of a lack of training data.

Among the participants, 4 were from the Arabic Wikipedia and participated in the judgement-based evaluation from *RQ*4.1. The remaining 6 were from other smaller Wikipedia language communities: Persian, Indonesian, Hebrew, Ukrainian (one per language), and Swedish (two participants).

While Swedish is officially the third largest Wikipedia in terms of number of articles,[8] most of the articles are not manually edited. In 2013, the Swedish Wikipedia passed one million articles, thanks to a bot called *lsjbot*, which at that point in time had created almost half of the articles on Swedish Wikipedia[9]. Such bot-generated articles are commonly limited, both in terms of information content and length. The high activity of a single bot is also reflected in the small number of active editors compared to the large number of articles (see Table 8.4).

The participants were experienced Wikipedia editors, with average tenures of 9.3 years in Wikipedia (between 3 and 14 years, median 10). All of them have contributed to at least one language besides their main language, and to the English Wikipedia. Further, 4 editors worked in at least two other languages beside their main language, while 2 editors were active in as many as 4 other languages. All participants knew about Wikidata, but had varying levels of experience with the project. 4 participants have been active on Wikidata for over 3 years (with 2 editors being involved since the start of the project in 2013), 5 editors had some experience with editing Wikidata, and one editor had never edited Wikidata, but knew the project.

[8] https://meta.wikimedia.org/wiki/List_of_Wikipedias

[9] https://blog.wikimedia.org/2013/06/17/swedish-wikipedia-1-million-articles/

Table 8.6 displays the sentence used for the interviews in different languages. The Arabic sentence is generated by the network based on Wikidata triples, while the other sentences are synthetically created as described below.

### 8.4.3 Ethics

The research was approved by the Ethics Committee of the University of Southampton under ERGO Number 44971 and written consent was obtained from each participant ahead of the interviews.

### 8.4.4 Data

For Arabic, we reused a summary sentence from the *RQ*4.1 evaluation. For the other language communities, we emulated the sentences the network would produce. First, we picked an entity to test with the editors. Then, we looked at the output produced by the network for the same concept in Arabic and Esperanto to understand possible mistakes and tokens produced by the network. We chose the city of *Marrakesh* as the concept editors would work on. Marrakesh is a good starting point, as it is a topic possibly highly relevant to readers and is widely covered, but falls into the category of topics that are potentially under-represented in Wikipedia due to its geographic location (Graham et al., 2014). An article about Marrakesh exists in 93 language editions (as of September 2020), including the ones in this study.

We collected the introductory sentences for Marrakesh in the editors' languages from Wikipedia. These are the sentences the network would be trained on and tries to reproduce. We ran the keyword matcher that was used for the preparation of the dataset on the original Wikipedia sentences. It marked the words that the network would pick up or that would be replaced by property placeholders. Therefore, these words could not be removed.

As we were particularly interested in how editors would interact with the missing word tokens the network can produce, we removed up to two words in each sentence: the word for the concept in its native language (e.g., *Morocco* in Arabic for non-Arabic sentences), as we saw that the network does the same, and the word for a concept that is not connected to the main entity of the sentence on Wikidata (e.g., *Atlas Mountains*, which is not linked to Marrakesh). An overview of all sentences used in the interviews can be found in Table 8.6.

### 8.4.5 Task

The interview started with an explanation that the researcher would observe the reading and editing of the participant in their language Wikipedia. Until both reading and editing were finished, the participant did not know about the provenance of the text. To start the interview, we asked demographic questions about the participants' contributions to Wikipedia and Wikidata, and to test their knowledge on the existing ArticlePlaceholder. Before reading, they were introduced to the idea of displaying content from Wikidata on Wikipedia. Then, they saw the mocked page of the ArticlePlaceholder, as can be seen in Figure 8.5 in Arabic. Each participant saw the page in their respective language. As the interviews were remote, the interviewer asked them to share the screen so they could point out details with the mouse cursor. Questions were asked to let them describe their impression of the page while they were looking at the page. Then, they were asked to open a new page, which can be seen in Figure 8.7. Again, this page would contain information in their language. They were asked to edit the page and describe what they are doing at the same time freely. We asked them not to edit a whole page but only to write two to three sentences as the introduction to a Wikipedia article on the topic with as much of the information given as needed. After the editing was finished, they were asked questions about their experience. (For the interview guideline, see Appendix C.) The interview followed the methodology of a semi-structured interview in which all participants were asked the same questions. Only then, at the end of the interview, was the provenance of the sentences revealed. Given this new information, we asked them how they thought the automatically generated sentence would impact their editing. Finally, we left time to discuss any questions the participants might want to raise. The interviews were scheduled to last between 20 minutes to one hour.

### 8.4.6 Analysing the interviews

We interviewed a total of 11 editors of seven different language Wikipedias. The interviews took place in September 2018. We used thematic analysis to evaluate the results of the interviews. The interviews were coded by two researchers independently, in the form of inductive coding based on the research questions. After comparing and merging all themes, both researchers independently applied these common themes to the text again.

### 8.4.7 Editors' reuse metric

Editors were asked to complete a writing task. We assessed how they used the automatically generated summary sentences in their work by measuring the amount of text reuse. We based the assessment on the editors' resultant summaries after the interviews were finished.

To quantify the amount of reuse of the generated summaries we use the Greedy String-Tiling (GST) algorithm (Wise, 1996). GST is a substring matching algorithm that computes the degree of reuse or copy from a source text and a dependent one. GST is able to deal with cases when a whole block is transposed, unlike other algorithms such as the Levenshtein distance, which calculates it as a sequence of single insertions or deletions rather than a single block move. Adler and de Alfaro (2007) introduce the concept of *Edit Distance* in the context of vandalism detection on Wikipedia. They measure the trustworthiness of a piece of text by measuring how much it has been changed over time. However, their algorithm punishes the copy of the text, as they measure every edit against the original text. In contrast, we want to measure how much of the text is reused. Therefore, GST is appropriate for the task at hand.

Given a generated summary $S = s_1, s_2, ..$ and an edited one $D = d_1, d_2, ..$, each consisting of a sequence of tokens, GST identifies a set of disjoint longest sequences of tokens in the edited text that exist in the source text (called ***tiles***) $T = \{t_1, t_2, ..\}$. It is expected that there will be common stop words appearing in both the source and the edited text. However, we were rather interested in knowing how much of the real structure of the generated summary is being copied. Thus, we set minimum match length factor $mml = 3$ when calculating the tiles, s.t. $\forall t_i \in T : t_i \subseteq S \land t_i \subseteq D \land |t_i| \geq mml$ and $\forall t_i, t_j \in T | i \neq j : t_i \cap t_j = \emptyset$. This means that copied sequences of single or double words will not count in the calculation of reuse. We calculated a reuse score *gstscore* by counting the lengths of the detected tiles, and normalised by the length of the generated summary.

$$gstscore(S, D) = \frac{\sum_{t_i \in T} |t_i|}{|S|} \tag{8.1}$$

We classified each of the edits into three groups according to the *gstscore* as proposed by Clough et al. (2002): 1) **Wholly Derived (WD)**: the summary sentence has been fully reused in the composition of the editor's text ($gstscore \geq 0.66$); 2) **Partially Derived (PD)**: the summary sentence has been partially used ($0.66 > gstscore \geq 0.33$); 3) **Non Derived (ND)**: the text has been changed completely ($0.33 > gstscore$).

### 8.4.8 Results: RQ4.2 - Task-based evaluation

As part of our interview study, we asked editors to read an ArticlePlaceholder page with included NLG text and asked them to comment on a series of issues. We grouped their answers into several general themes around: their use of the snippets; their opinions on text provenance; the ideal length of the text; the importance of the text for the ArticlePlaceholder; and limitations of the algorithm.

**Use** The participants appreciated the summary sentences:

> ”I think it would be a great opportunity for general Wikipedia readers to help improve their experience, while reading Wikipedia.” (P7)

Some of them noted that the summary sentence on the ArticlePlaceholder page gave them a useful overview and quick introduction to the topic of the page, particularly for people trained in one language or non-English speakers:

> ”I think that if I saw such an article in Ukrainian, I would probably then go to English anyway, because I know English, but I think it would be a huge help for those who don't.” (P10)

**Provenance** As part of the reading task, we asked the editors what they believed was the provenance of the information displayed on the page. This gave us more context to the promising fluency and appropriateness scores achieved in the quantitative study. The editors made general comments about the page and tended to assume that the generated sentence was from other Wikipedia language versions:

> [The generated sentence was] ”taken from Wikipedia, from Wikipedia projects in different languages.” (P1)

Editors more familiar with Wikidata suggested the information might be derived from Wikidata's descriptions:

> "it should be possible to be imported from Wikidata" (P9)

Only one editor could spot a difference in the generated sentence (text) and regular Wikidata triples:

> "I think it's taken from the outside sources, the text, the first text here, anything else I don't think it has been taken from anywhere else, as far as I can tell." (P2)

Overall, the answers supported our assumption that NLG, trained on Wikidata labelled triples, could be naturally added to Wikipedia pages without changing the reading experience. At the same time, the task revealed questions around algorithmic complexity and capturing provenance. Both are relevant to ensuring transparency and accountability and helping flag quality issues.

**Length** We were interested in understanding how we could iterate over the NLG capabilities of our system to produce text of appropriate length. While the model generated just one sentence, the editors thought it to be a helpful starting point:

> "Actually I would feel pretty good learning the basics. What I saw is the basic information of the city so it will be fine, almost like a stub." (P4)

While generating larger pieces of text could arguably be more useful, reducing the need for manual editing even further, the fact that the placeholder page contained just one sentence made it clear to the editors that the page still requires work. In this context, another editor referred to a *'magic threshold'* for an automatically generated text to be useful (see also Section 8.4.9). Their expectations for an NLG output were clear:

> "So the definition has to be concise, a little bit not very long, very complex, to understand the topic, is it the right topic you're looking for or" (P1)

We noted that whatever the length of the snippet, it needs to match reading practice. Editors tend to skim and scan articles rather than reading them in detail:

> "[...] most of the time I don't read the whole article, it's just some specific, for instance a news piece or some detail about I don't know, a program in languages or something like that and after that, I just try to do something with the knowledge that I learned, in order for me to acquire it and remember it" (P6) "I'm getting more and more convinced that I just skim" (P1) "I should also mention that very often, I don't read the whole article and very often I just search for a particular fact" (P3) "I can't say that I read a lot or reading articles from the beginning to the end, mostly it's getting, reading through the topic, "Oh, what this weird word means," or something." (P10)

When engaging with content, people commonly go straight to the part of the page that contains what they need. If they are after an introduction to a topic, having a summary at the top of the page, for example in the form of an automatically generated summary sentence, could make a real difference in matching their information needs.

**Importance** When reading the ArticlePlaceholder page, people looked first at our text:

> ″The introduction of this line, that's the first thing I see.″ (P4)

This is their way to confirm that they landed on the right page and if the topic matches what they were looking for:

> ″Yeah, it does help because that's how you know if you're on the right article and not a synonym or some other article.″ (P1)

This makes the text critical for engagement with the ArticlePlaceholder page, where most of the information is expressed as Wikidata triples. Natural language can add context to structured data representations:

> ″Well that first line was, it's really important because I would say that it [the ArticlePlaceholder page] doesn't really make a lot of sense [without it] ... it's just like a soup of words, like there should be one string of words next to each other so this all ties in the thing together. This is the most important thing I would say.″ (P8)

<`rare`> **Tags** To understand how people react to a fault in the NLG algorithm, we chose to leave the <`rare`> tags in the summary sentences the participants saw during the reading task. As mentioned earlier, such tags refer to entities in the triples that the algorithm was unsure about and could not verbalise. We did not explain the meaning of the tokens to the participants beforehand and none of the editors mentioned them during the interviews. We believe this was mainly because they were not familiar with what the tokens meant and not because they were not able to spot errors overall. For example, in the case of Arabic, the participants pointed to a Wikidata property with an incorrect label, which our NLG algorithm reused. They also picked up on a missing label in the native language for a city. However, the <`rare`> tokens were not noticed in any of the 10 reading tasks until explicitly mentioned by the interviewer. The name of the city Marrakesh in one of the native languages (Tamazight) was realised using the <`rare`> token (see the Arabic sentence in Table 8.6). One editor explained that the fact that they are not familiar with this language (Tamazight) and can therefore not evaluate the correctness of the statement is the main reason that they oversaw the token:

> ″the language is specific, it says that this is a language that is spoken mainly in Morocco and Algeria, the language, I don't even know the symbols for the alphabets [...] I don't know if this is correct, I don't know the language itself so for me, it will go unnoticed. But if somebody from that area who knows anything about this language, I think they might think twice.″ (P8)

| Category | Examples | # |
|---|---|---|
| WD | Марракеш (араб. <rare>) — важливе імперське місто в Марокко, розташованого біля підніжжя гір <rare>.<br>Марракеш (араб. مراكش) — важливе імперське місто в Марокко, розташованого біля підніжжя гір. | 8 |
| PD | Marrakech (arabiska <rare>, Berberspråk <rare>) är en stad i sydvästra Marocko, vid foten av <rare>.<br>Marrakech (arabiska: مراكش, tamazight: ⵎⵕⵕⴰⴽⵛ) är en stad i Marocko med 928 850 invånare (2014). | 2 |
| ND | مُرَاكُش (بالأمازيغية: <rare>) هي مدينة مغربية تقع شمال البلاد .<br>مُرَاكُش مدينة سياحية مشهورة. | 0 |

TABLE 8.7: Number of snippets in each category of reuse. A generated snippet (top) and its edited version (bottom). Solid lines represent reused tiles, while dashed lines represent overlapping sub-sequences not contributing to the *gstscore*. The first two examples are created in the studies, the last one (ND) is from a previous experiment. A list of all created sentences and their translations to English can be found in Appendix D.

### 8.4.9 Results: RQ4.3 - Task-based evaluation

Our third research questions focused on how people work with automatically generated text. The overall aim of adding NLG to the ArticlePlaceholder is to help Wikipedia editors bootstrap missing articles without disrupting their editing practices.

As noted earlier, we carried out a task-based evaluation in which the participants were presented with an ArticlePlaceholder page that included the summary sentence and triples from Wikidata relevant to the topic. We carried out a quantitative analysis of the editing activities via the GST score, as well as a qualitative, thematic analysis of the interviews, in which the participants explained how they changed the text. In the following we will first present the GST scores, and then discuss the themes that emerged from the interviews.

**Reusing the text** As shown in Table 8.7, the snippets are heavily used and all participants reused them at least partially. 8 of them were wholly derived ($WD$) and the other 2 were partially derived ($PD$) from the text we provided, which means an average GST score of 0.77. These results are in line with a previous editing study of ours, described in Kaffee et al. (2018a), with a sample of 54 editors from two language communities, 79% and 93% of the snippets were either wholly ($WD$) or partially ($PD$) reused.

We manually inspected all edits and compared them to the 'originals' - as explained in Section 8.2, we had 10 participants from 6 language communities, who edited 6 language versions of the same article. In the 8 cases where editors reused more of the text, they tended to copy it with minimal modifications, as illustrated in sequences $A$ and $B$ in Table 8.7. Three editors did not change the summary sentence at all (including the special token), but only added to it based on the triples shown on the ArticlePlaceholder page.

One of the common things that hampers the full reusability are <rare> tokens. This can lead editors to rewrite the sentence completely, as in the PD example in Table 8.7.

**Editing experience** While GST scores gave us an idea about the extent to which the automatically generated text is reused by the participants, the interviews helped us understand how they experienced the text. Overall, the summary sentences were widely praised as helpful, especially for newcomers:

> ”Especially for new editors, they can be a good starting help: ”I think it would be good at least to make it easier for me as a reader to build on the page if I’m a new volunteer or a first time editor, it would make adding to the entry more appealing (...) I think adding a new article is a barrier. For me I started only very few articles, I always built on other contribution. So I think adding a new page is a barrier that Wikidata can remove. I think that would be the main difference.” (P4)

All participants were experienced editors. Just like in the reading task, they thought having a shorter text to start editing had advantages:

> ”It wasn’t too distracting because this was so short. If it was longer, it would be (...) There is this magical threshold up to which you think that it would be easier to write from scratch, it wasn’t here there.” (P10)

The length of the text is also related to the ability of the editors to work around and fix errors, such as the `<rare>` tokens discussed earlier. When editing, participants were able to grasp what information was missing and revise the text accordingly:

> ”So I have this first sentence, which is a pretty good sentence and then this is missing in Hebrew and well, since it’s missing and I do have this name here, I guess I could quickly copy it here so now it’s not missing any longer.” (P3)

The same participant checked the Wikidata triples listed on the same ArticlePlaceholder page to find the missing information, which was not available there either, and then looked it up on a different language version of Wikipedia. They commented:

> ”This first sentence at the top, was it was written, it was great except the pieces of information were missing, I could quite easily find them, I opened the different Wikipedia article and I pasted them, that was really nice” (P3).

Other participants mentioned a similar approach, though some decided to delete the entire snippet because of the `<rare>` token and start from scratch. However, the text they added turned out to be very close to what the algorithm generated.

> ”I know it, I have it here, I have the name in Arabic, so I can just copy and paste it here.” (P10)
> ”[deleted the whole sentence] mainly because of the missing tokens, otherwise it would have been fine.” (P5)

One participant commented at length on the presence of the tokens:

> ”I didn’t know what rare [is], I thought it was some kind of tag used in machine learning because I’ve seen other tags before but it didn’t make any sense because I didn’t know what use it had, how can I use it, what’s the use of it and I would say it would be distracting, if it’s like in other parts of the page here. So that would require you to understand first what rare does there, what is it for and that would take away the interest I guess, or the attention span so it would be better just to, I don’t know if it’s for instance, if the word is not, it’s rare, this part right here which is, it shouldn’t be there, it should be more, it would be better if it’s like the input box or something.” (P1)

Overall, the editing task and the follow-up interviews showed that the summary sentences were a useful starting point for editing the page. Missing information, presented in the form of <rare> markup, did not hinder participants from editing and did not make them consider the snippets less useful. Although they were unsure about what the tokens meant, they intuitively replaced them with the information they felt was missing, either by consulting the Wikidata triples that had not been summarised in the text, or by trying to find that information elsewhere on Wikipedia.

## 8.5 Discussion

Our first research question focuses on how well an NLG algorithm can generate summaries from the Wikipedia reader’s perspective. In most of the cases, the text is considered to be from the Wikimedia environment. Readers do not clearly differentiate between the generated summary and an original Wikipedia sentence. While this indicates the high quality of the generated textual content, it is problematic with respect to trust in Wikipedia. Trust in Wikipedia and how humans evaluate trustworthiness of a certain article has been investigated using both quantitative and qualitative methods. Adler et al. (2008) and Adler and de Alfaro (2007) develop a quantitative framework based on Wikipedia’s history. Lucassen and Schraagen (2010) use a qualitative methodology to code readers’ opinions on the aspects that indicate the trustworthiness of an article. However, none of these approaches take the automatic creation of text by non-human algorithms into account. A high quality Wikipedia summary, which is not distinguishable from a human-generated one, can be a double-edged sword. While conducting the interviews, we realised that the Arabic generated summary has a factual mistake. We showed in previous work (Vougiouklis et al., 2018) that those factual mistakes occur relatively seldom; however, they are a known drawback of neural text generation. In our case, the Arabic sentence stated that Marrakesh was located in the north, while it is actually in the centre of the country. One of the participants lives in Marrakesh. Curiously, they did not realise this mistake, even while translating the sentence to the interviewee:

> ”Yes, so I think, so we have here country, Moroccan city in the north, I would say established and in year (. . . )” (P1)

As we did not introduce the sentence as automatically generated, we assume this acceptance of the error by the editors was due to both the trust Wikipedians have in their platform, and the quality of the sentence:

> ”This sentence was so well written that I didn’t even bother to verify if it’s actually a desert city.” (P3)

So as to not misinform readers and editors and eventually damage this trust, future work will have to investigate ways of dealing with such unfactual statements. On a technical level, that will mean exploring ways to excluding sentences with factual mistakes, called *hallucinations*. These hallucinations can be found across different models, including the most recent self-attentive NLG architectures (Koncel-Kedziorski et al., 2019). On an HCI level, investigating ways of communicating the problems of neural text generation to the reader will be needed. One possible solution to this problem could be visualisations, as these can influence the trust given to a particular article (Kittur et al., 2008). One example of this technology is WikiDashboard (Pirolli et al., 2009). A similar tool could be used for the provenance of text.

Supporting the previous results from the readers, editors also seem to have a positive perception of the summaries. It is the first thing they read when they arrive at a page and it helps them to quickly verify that the page is about the topic they are looking for.

When creating the summaries, we assumed their relative brevity might be a point for improvement from an editors’ perspective. After all, as research suggests that the length of an article indicates its quality – basically, the longer, the better (Blumenstock, 2008). From the interviews with editors, we found that they mostly skim and scan articles when reading them. This seems to be the more natural way of browsing the information in an article, and is supported by the short summary that gives an overview of the topic.

All editors we worked with are part of the multilingual Wikipedia community, editing in at least two Wikipedias. Hale (2014, 2015) highlights that users of this community are particularly active compared to their monolingual counterparts and confident in editing across different languages. However, taking potential newcomers into account, editors suggest that the ArticlePlaceholder might be helpful to lower the barrier for starting to edit. Recruiting more editors has been a long-standing objective, with initiatives such as the Tea House (Morgan et al., 2013) aiming at welcoming and encouraging new editors; Wikipedia Adventure employs a similar approach using a tool with gamification features (Narayan et al., 2017). The ArticlePlaceholder, and in particular the provided summaries in natural language, could have an impact on how, and how early in their connection with Wikipedia, people start editing.

In comparison to Wikipedia Adventure, the readers are exposed to the ArticlePlaceholder pages. This could overcome their reservations about editing by offering a more natural place to start.

Lastly, we asked the research question of how editors use the textual summaries in their workflow. In general, we show that the text is highly reused. One of the editors mentions a *magic threshold* that makes the summary acceptable as a starting point for editing. This seems similar to post-editing in machine translation (or rather, monolingual machine translation (Hu et al., 2011), where a user only speaks the target or source language). Translators have been found to oppose machine translation and post-editing, as they perceive it as more time consuming and restrictive with respect to their freedom in the translation (e.g., sentence structure) (Lagoudaki, 2009). Nevertheless, it has been shown that a different interface can not only lead to reduced input time and enhanced quality, but also convinces a user to believe in the improved quality of the machine translation (Green et al., 2013). This underlines the importance of the right integration with machine-generated sentences, as we aim for in the ArticlePlaceholder.

The core of this work is to understand the perception of a community, such as Wikipedians, of the integration of a state-of-the-art machine learning technique for NLG in their platform. We show that such an integration can work and be supported. This finding aligns with other projects already deployed on Wikipedia. For instance, bots (short for robots) that monitor Wikipedia have become a trusted tool for vandalism fighting (Geiger and Ribes, 2010) - so much so, that they are even empowered to revert edits made by humans if they believe them to be malicious. The cooperative work between humans and machines on Wikipedia has also been theorised in machine translation. Alegria et al. (2013) argue for the integration of machine translation into Wikipedia that learns from the post-editing of the editors. Such a human-in-the-loop approach is also applicable to our NLG work, where an algorithm could learn from the humans' contributions.

There is a need for investigating this direction further, since NLG algorithms will not achieve the same quality as humans. Especially in a low-resource setting like the one observed in this work, human support is needed. However, automated tools can be a great way of allocating the limited human resources to the tasks where they are mostly needed. Post-editing the summaries can serve a purely data-driven approach such as ours with additional data that can be used to further improve the quality of the automatically generated content. To make such an approach feasible for the ArticlePlaceholder, we need an interface that encourages the editing of the summaries. The less effort this editing requires, the more we can ensure an easy collaboration between human and machine.

## 8.6 Limitations

We interviewed 11 editors having different levels of Wikipedia experience. As all editors are already Wikipedia editors, the conclusions we can draw for new editors are limited. We focus on experienced editors, as we expect them to be the first editors to adapt the ArticlePlaceholder in their workflow. Typically, new editors will follow the existing guidelines and the standards set by the experienced editors; therefore, the focus on experienced editors will give us an understanding of how editors overall will accept and interact with the new tool. Further, it is difficult to sample from new editors, as there is a variety of factors that can make a contributor develop into a long-term editor or not (Kuznetsov, 2006).

The distribution of languages favours Arabic, as this community was most responsive. This can be assumed to be due to previous collaborations. While we cover different languages, we cover only a small part of the different language communities that Wikipedia covers in total. Most studies of Wikipedia editors currently focus on English Wikipedia (Panciera et al., 2009). Even the few studies that observe multiple language Wikipedia editors do not include the span of insights from different languages that we provide in this study. In our study we treat the different editors as members of a unified community of Wikipedia underserved languages. This is supported by the fact that their answers and themes were consistent across different languages. Therefore, adding more editors of the same languages would not have brought a benefit.

We aimed our evaluation at two populations: readers and editors. While the main focus was on the editors and their interaction with the new information, we wanted to include the readers' perspective. In the readers evaluation we focus on the quality of text (in terms of fluency and appropriateness), as this will be the most influential factor for their experience on Wikipedia. Readers, while usually overlooked, are an important part of the Wikipedia community (Lemmerich et al., 2019). Together, these two groups form the Wikipedia community, with new editors being recruited from the existing pool of readers, and readers making their own essential contributions to the platform, as shown by Antin and Cheshire (2010).

The sentences the editors worked on are synthesised, i.e., not automatically generated but created by the authors of this study. An exception is the Arabic sentence, which was generated by the approach described in Section 8.2. While the synthesised sentences were not created by natural language generation, they were created and discussed with other researchers in the field. We therefore focused on the most common problem in text generative tasks similar to ours: the `<rare>` tokens. Other problems of neural language generation, such as factually wrong sentences or hallucinations (Rohrbach et al., 2018), were excluded from the study, as they are not a common problem for short summaries such as ours. The extent of hallucinations on single sentence generative scenarios has also been explored by Chisholm et al. (2017), who have shown a precision of 93% in the generation of English biographies. However, we believe this topic should be explored further in future work given that, as we show in our study, factual mistakes caused by hallucinations can be easily missed by editors. Further, our study set-up was focusing on the integration in the ArticlePlaceholder, i.e., the quality of the sentences, rather than their factual correctness. We leave it to future work to apply the mixed-methods approach of evaluating natural language generation proposed in this paper to full generated articles.

# Chapter 9

# Conclusions

In this thesis, we have created a set of studies to assess, improve, and apply the language coverage of knowledge graphs. First, we propose a framework to measure multilinguality and labelling in knowledge graphs. We apply this framework to a representation of the web of data, and to Wikidata. We show there is a lack of broad language coverage in labels of knowledge graphs. Wikidata shows a more diverse coverage of languages, drawn from the knowledge of its multilingual editing community. We then use the framework as a base to rank knowledge graphs for question answering. Motivated by the lack of multilingual knowledge in knowledge graphs, we explore an approach to the generation of multilingual aliases for Wikidata from English. Further, we use the multilingual labels to generate introductory sentences for Wikipedia.

We can conclude that while the web of data still lacks multilingual information, it is a good starting point for multilingual applications, as the structured information facilitates multilingual access for humans and machines alike. For example, ArticlePlaceholder shows that we can improve Wikipedia's readers' experience by reusing multilingual knowledge graph data.

## 9.1 Contributions and Results

In Section 1, we introduced set of research questions, which we outline again here. We want to answer the primary question: *How can we support multilingual access to knowledge graphs for speakers of low-resourced languages?* We explore this research question from four different perspectives. We detail the research questions and the chapters addressing each research question below.

RQ1 To get an insight into the availability of language information and labels on the web of data at large and in Wikidata in specific, we start our studies with the question *RQ1: What is the state of knowledge graphs with regard to labels and multilinguality?*
Chapters 3, 4, and 5 explored different aspects of this research question.

RQ2 Gaining an insight into the state of labels and multilinguality in knowledge graphs lets us reuse this knowledge for applications. We test this assumption in the domain of question answering, posing the research question *RQ2: How can knowledge about languages in a knowledge graph be applied to the task of ranking them for question answering?*
In Chapter 6, we addressed this research question.

RQ3 Not only question answering systems but also a large number of other applications are dependent on multilingual labels in knowledge graphs. However, with a lack of labels currently observed in knowledge graphs, these applications also have a limited usability. Therefore, we explore *RQ3: How does differentiating between translation and transliteration impact the generation of new knowledge graph labels and aliases?*
Chapter 7 shows an approach to addressing this research question.

RQ4 Finally, we explored another use case of multilingual labels and making knowledge graph information more accessible to Wikipedia readers through the creation of ArticlePlaceholders on Wikipedia. In this context, we pose the research question *RQ4: How do Wikipedia editors perceive automatically generated Wikipedia summaries?*
In Chapter 8, we addressed this research question.

Below, we detail the results of our studies relating to each research question.

### 9.1.1 RQ1 What is the state of knowledge graphs with regard to labels and multilinguality?

In Chapter 3, we introduced a framework to measure label and language information in knowledge graphs. Then the framework was applied to two datasets. First, in Chapter 4, we observed the language distribution of the web of data at large based on the LOD Laundromat dataset. Then, in Chapter 5, we focused on Wikidata.

**Web of Data** In Chapter 3, we proposed a framework to measure the coverage of languages in knowledge graphs, and in Chapter 4 applied it to the LOD Laundromat dataset, a representation of the web of data. The most widely used labelling property is `rdfs:label`, 1,384% more used than the second most frequent labelling property (`foaf:name`). This is promising, as it supports easy automated access to labels and decreases the need for ontology alignment between different datasets and their labelling properties.

We found a lack of entity labelling across the LOD Laundromat dataset – only 5.42% of the subjects are labelled. However, for the properties, which are much smaller in number but highly reused across the dataset, 80.9% are labelled. Even if datasets cover multiple languages, the five most used languages across the knowledge graph cover over 50% of labels, and entities are likely to be labelled only in one language (83.2%), which limits access.

**Wikidata** To gain a better understanding of the coverage of Wikidata's language communities, we analysed data on natural language labels from Wikidata in Chapter 5 based on the framework introduced in Chapter 3.

There is still much room for improvement on the current state; as with most of the web, Wikidata's knowledge is mostly available in a few languages, while most languages have close to no coverage. Even languages spoken by large parts of the world's population are not necessarily well covered. The languages with the most coverage are similar to those with the greatest presence on Wikipedia, which we assume to be due to two factors: imports of Wikipedia article titles, and an overlap of communities.

A few promising observations can be made, however: languages do not necessarily have to be spoken by many people to achieve a higher level of completeness; suitable tools can greatly accelerate the process; and there are more comprehensive translations for data that is used more, as shown with properties in Wikidata.

The hybrid approach of Wikidata, of humans and bots editing the knowledge graph side by side, supports collaborative work towards the completion of the knowledge graph. The different roles of bots and humans complement each other, as we outline in our work. The results of this work can be a starting point for a variety of tools to support the editors, e.g., by suggesting edits to editors based on the knowledge of what bots typically do not do, and, analogously, by suggesting the creation of bots for typical bot tasks in labels.

### 9.1.2 RQ2 How can knowledge about languages in a knowledge graph be applied to the task of ranking them for question answering?

In Chapter 6, we addressed the problem of capturing knowledge about the multilinguality of existing knowledge graphs. Our approach showed an application domain for the framework introduced in Chapter 3. We proposed LINGVO, a framework able to compare and rank knowledge graphs based on multilingual knowledge at class level. LINGVO provides computation methods to extract and store knowledge about the classes in a knowledge graph (i.e., Class-based Label Captures (CLCs)) in terms of labels and languages based on our framework. We empirically showed that ranking on class-level leads to precise results, testing over five widely used knowledge graphs: Wikidata, DBpedia, YAGO, MusicBrainz, and LinkedMDB. Moreover, the ranking of these knowledge graphs is particularly effective when it is performed with respect to a contextual domain, e.g., movies or people. These results support the statement that capturing knowledge about multilinguality paves the way for the development of the new generation of multilingual applications.

### 9.1.3 RQ3 How does differentiating between translation and transliteration impact the generation of new knowledge graph labels and aliases?

In Chapter 7, we explored the impact of differentiating between transliteration and translation when generating a Chinese alias from an English label in the company domain of a knowledge graph. We performed a crowdsourcing study annotating (English label, Chinese alias) pairs and showed that obtaining aliases in Chinese for these entities is challenging given that they can be transliterated, translated, or both. We then explored the usage of knowledge graph embeddings to classify an entity into whether it should be translated or transliterated, using a different model for each. Effective classification between cases before generating an alias improves results compared to using either a translation- or transliteration-based model on its own - an improvement of 34.7% for the transliteration model and 25.4% for the translation model, in terms of CER. When exploring the Chinese datasets, we found that models are expected to predict into two different character sets, i.e., traditional and simplified Chinese. Converting all Chinese training and test datasets can improve performance by 77.12% in terms of BLEU-4 score. Leveraging knowledge graph embeddings does have limitations for this task, and in future work we aim to explore training them on properties and relationships that are important for our task.

### 9.1.4 RQ4 How do Wikipedia editors perceive automatically generated Wikipedia summaries?

In Chapter 8, we conducted a quantitative study with members of the Arabic and Esperanto Wikipedia communities, and semi-structured interviews with members of six different Wikipedia communities, in order to understand the communities' understanding and acceptance of generated text in Wikipedia. To understand the impact of automatically generated text for a group such as the Wikipedia editors, we surveyed their perception of generated summaries in terms of fluency and appropriatness. To deepen this understanding, we conducted 10 semi-structured interviews with experienced Wikipedia editors from 6 different language communities and measured their reuse of the original summaries.

The addition of the summaries seems to be natural for readers: we showed that Wikipedia editors rank our text close to the expected quality standards of Wikipedia, and are likely to consider the generated text to be part of Wikipedia. The language the neural network produces integrates well with the existing content of Wikipedia, and readers appreciate the summary in the ArticlePlaceholder, as it is the most helpful element on the page to them.

We show that editors assumed the text was part of Wikipedia, and that the summary improves the ArticlePlaceholder page. In particular, the summary being short supported the editors' usual workflow of scanning the page for information needed. The missing word token, which we included to gain an understanding of how users interact with faultily produced text, did not hinder the reading experience nor the editing experience. Editors are likely to reuse a large portion of the generated summaries. Additionally, participants mentioned that the summary can be a good starting point for new editors.

## 9.2 Future Work

Digital technologies provide the opportunity to bridge gaps between communities speaking different languages; and legislation created by international organisations, such as the European Commission[1], paves the way for the further development of multilingual applications. In order to address the challenge of supporting multilinguality in digital applications, extensive quantities of knowledge demand to be captured. Knowledge graphs have become a popular formalism for representing entities and their properties using a graph

[1] https://ec.europa.eu/digital-single-market/en/blog/multilingualism-digital-age-barrier-or-opportunity

data model - they supply the expressive power to represent the knowledge required for supporting multilingual applications. In our work we show three main areas of research: the current state of language coverage of knowledge graphs, and how to assess this coverage; the increase of knowledge graph labels in lower-resourced languages; and applications for multilingual labels. We focus on suggestions for future work on the last two directions: increasing knowledge graph language coverage, and leveraging multilingual knowledge graph information for applications. The suggestions are based on needs identified in the course of this research, and projects that have a focus on supporting language communities with knowledge graphs.

### 9.2.1 Increasing knowledge graph language coverage

Assessing the state of the web of data with regard to label and language coverage, we show that there is an urgent need for a more diverse language coverage in order to support communities across languages. Future work will have to deepen the understanding of how to create more multilingual knowledge graph data. We studied two approaches: Wikidata editors creating multilingual data through manual or bot imports of labels (Chapter 5); and the translation and transliteration of knowledge graph labels in an automated manner (Chapter 7).

We show that contextual information in the form of knowledge graph embedding, as used in Chapter 7, is a starting point for the transliteration and translation of knowledge graph labels. Further exploration is needed into the potential use of linguistic information, for example, lexicographical data introduced to Wikidata in the form of *lexemes*.[2] This will require a larger corpus of the multilingual information in lexemes. On the status of lexemes in Wikidata as of 2020, Nielsen (2020) states:

> "The top language with most lexemes is Russian (101,137 lexemes), followed by English (38,122), Hebrew (28,278), Swedish (21,790), Basque (18,519), French (10,520) and Danish (4,565). Russian is also the language with more forms than any other language (1,236,456), followed by Basque (956,473), Hebrew (446,795), Swedish (148,980), Czech (77,747) and English (64,798). For senses, the languages from the top are Basque (20,272), English (12,911), Hebrew (3,845), Russian (2,292) and Danish (2,217)." (Nielsen, 2020)

How to improve the language coverage and usage of the lexemes is currently largely unexplored, opening a large research field for lexicographical data linked to Wikidata's entities.

Another approach to increasing the coverage of knowledge graph labels in lower-resourced languages could use relation extraction. This approach can make use of existing text in the target languages, and extract language based on the existing information in the knowledge graph. In particular, the close connection between Wikidata and Wikipedia could be leveraged in such an approach.

It is yet to be investigated how low-resourced languages could be supported to ensure their coverage in terms of knowledge graph labels. In languages with limited resources online, a

[2] https://www.wikidata.org/wiki/Wikidata:Lexicographical_data, retrieved 29. April 2021

high coverage in labels in structured data could change the possibilities for application drastically. It would enable a large number of applications to be made available to speakers of low-resourced languages in an automated manner. A solution that places due emphasis on both the involvement of the relevant communities and technological solutions needs to be explored in future work. It is essential to encourage the involvement of the relevant language communities in such work, not least because the available training data is limited, and gaps can only be bridged by manual annotations.

### 9.2.2 Leveraging multilingual knowledge graph information

**In Wikimedia projects** An aim of our research was to make multilingual information accessible to readers of Wikipedia with the ArticlePlaceholder introduced in Chapter 8. A new approach to making Wikipedia accessible across languages is *Abstract Wikipedia*, introduced by Vrandecic (2018). Abstract Wikipedia will be editable in a meta-language, which can then be translated across languages using linguistic information similar to lexemes. This project will require great numbers of user studies, starting with the question of how to make it accessible for all communities of Wikipedia.

Since Wikipedia is a tertiary resource[3] and as such should contain no original knowledge, accurate and ample references are crucial to the site's effectiveness. There is a general lack of adequate referencing across all language versions of Wikipedia and Wikidata. The set of articles on a given topic across different language Wikipedias will have different references, and different numbers of references.[4] In the *Scribe*[5] project, we explore using knowledge graph information, such as the link between different language Wikipedia articles and publisher metadata, to suggest new references to Wikipedia editors in different languages. The interaction of references between Wikidata and Wikipedia's different language versions is yet to be explored in more detail. Relation extraction could make it possible to reuse references from Wikipedia for Wikidata, while existing references from Wikidata could be suggested for Wikipedia articles. The availability of different language labels is crucial to ensuring such approaches will work for all Wikipedia language versions.

**Outside Wikimedia projects** A more equal distribution of knowledge graph data among different languages has many benefits. There are various applications of this in the field of natural language processing research that are yet to be explored. Especially for low-resourced languages, and initiatives such as *Masakhane*[6] focusing on African languages, more knowledge graph labels in a wider range of languages could change the way we develop applications in these languages (∀ et al., 2020).

With a larger number of multilingual labels in knowledge graphs, question answering systems could be developed that would be accessible in a large number of languages by default. Furthermore, any application relying on named entities can benefit from multilingual labels.

[3] https://en.wikipedia.org/wiki/Wikipedia:Wikipedia_is_a_tertiary_source, retrieved 29. April 2021

[4] https://misinfocon.com/scribes-reference-api-enables-users-to-access-wikipedia-references-b8f749bf60d1, retrieved 29. April 2021

[5] https://meta.wikimedia.org/wiki/Scribe, retrieved 29. April 2021

[6] https://www.masakhane.io/, retrieved 29. April 2021

For example, Google Maps transliterated names of places in 10 Indian languages (Tech Desk, New Delhi, 2021).

## 9.3 Final remarks

This thesis contributes to an area of research that has seen substantial growth in the interval between the first study we discuss here and the most recent one. This intensifying scholarly interest reflects the fact that facilitating access to information for people across languages and geographies is one of the key challenges of our time. UNESCO emphasises the need for access to information as part of the UN's sustainable development goals:

> "At a time of profound transformation, inequality and upheaval, UNESCO has redoubled its efforts to ensure freedom of expression, access to information and inclusive digital development worldwide. However, much more remains to be done."[7]

Historical language inequality has fed into a corresponding inequality on the web (and subsequently the web of data): information that all of humanity could benefit from remains the preserve of speakers of a small set of languages (DiMaggio et al., 2004; Heller and McElhinny, 2017). Our work aims to contribute to a more just future, in which speakers of currently lower-resourced languages are not prevented from accessing knowledge online. There is still a long way to go to support a broader range of languages. Nevertheless, recent developments in the field of natural language generation and machine translation show promise. Wikipedia and Wikidata will play an important role in these developments, showcasing the possibility of a large community working together towards a common goal of more equal distribution of knowledge.

[7] `https://en.unesco.org/ci-programme`, retrieved 20. June 2021

# Appendix A

# Labelling Properties in LOD Laundromat

| Labelling property | Usage |
|---|---:|
| <http://www.w3.org/2000/01/rdf-schema#label> | 246201989 |
| <http://xmlns.com/foaf/0.1/name> | 16590050 |
| <http://www.w3.org/2004/02/skos/core#prefLabel> | 12270826 |
| <http://www.loc.gov/mads/rdf/v1#authoritativeLabel> | 10064682 |
| <http://purl.org/dc/terms/title> | 6721999 |
| <http://purl.org/dc/elements/1.1/title> | 3497188 |
| <http://rdf.freebase.com/ns/type.object.name> | 3496386 |
| <http://lexvo.org/ontology#label> | 2204386 |
| <http://www.w3.org/2008/05/skos-xl#literalForm> | 1028504 |
| <http://sw.cyc.com/CycAnnotations_v1#label> | 516098 |
| <http://purl.org/rss/1.0/title> | 507321 |
| <http://www.livejournal.org/rss/lj/1.0/journaltitle> | 269947 |
| <http://usefulinc.com/ns/doap#name> | 219296 |
| <http://purl.org/goodrelations/v1#name> | 189025 |
| <http://www.w3.org/2006/03/wn/wn20/schema/lexicalForm> | 146842 |
| <http://www.geonames.org/ontology#officialName> | 96647 |
| <http://www.w3.org/2006/03/wn/wn20/schema/gloss> | 84591 |
| <http://www.w3.org/2004/02/skos/core#hiddenLabel> | 80628 |
| <http://rdf.freebase.com/ns/biology.organism_classification.scientific_name> | 77292 |
| <http://rdf.insee.fr/geo/nom> | 41230 |
| <http://ec.europa.eu/eurostat/ramon/ontologies/nace.rdf#name> | 25891 |
| <http://skipforward.net/skipforward/resource/seeder/skipinions/itemName> | 23359 |
| <http://purl.org/collections/nl/am/title> | 11596 |
| <http://www.w3.org/2008/05/skos#prefLabel> | 8133 |
| <http://schema.org/name> | 7107 |
| <http://www.w3.org/2006/03/wn/wn20/schema/senseLabel> | 6088 |
| <http://spatial.ucd.ie/lod/osn/property/valueLabel> | 4585 |
| <http://spatial.ucd.ie/lod/osn/property/keyLabel> | 4380 |
| <http://rdf.freebase.com/ns/organization.leadership.title> | 4369 |
| <http://rdf.freebase.com/ns/organization.organization_board_membership.title> | 2634 |
| <http://www.inter2geo.eu/2008/ontology/ontology.owl#defaultCommonName> | 2354 |
| <http://vitro.mannlib.cornell.edu/ns/vitro/public#filename> | 2178 |
| <http://purl.org/nxp/schema/v1/groupName> | 2147 |
| <http://www.fao.org/countryprofiles/geoinfo/geopolitical/resource/nameOfficial> | 2031 |
| <http://vitro.mannlib.cornell.edu/ns/vitro/0.7#modTime> | 1964 |
| <http://www.fao.org/countryprofiles/geoinfo/geopolitical/resource/nameList> | 1944 |

| | |
|---|---|
| <http://www.news-project.com/Ontology/2008/03/skos_redefined/core#definition> | 1772 |
| <http://www.fao.org/countryprofiles/geoinfo/geopolitical/resource/nameCurrency> | 1398 |
| <http://purl.org/dc/terms/identifier> | 1176 |
| <http://rdf.freebase.com/ns/base.newsevents.news_report.title> | 1127 |
| <http://prismstandard.org/namespaces/1.2/basic/location> | 1022 |
| <http://xmlns.com/foaf/0.1/accountName> | 1021 |
| <http://rdfs.org/sioc/ns#name> | 854 |
| <http://www.w3.org/2004/02/skos/coreprefLabel> | 770 |
| <http://xmlns.com/foaf/0.1/firstName> | 745 |
| <http://pleiades.stoa.org/places/vocab#nameAttested> | 720 |
| <http://wwwis.win.tue.nl/∼ppartout/Blu-IS/Ontologies/TV-Anytime/PhaseI/Classifications/ContentCS.owl#Name> | 685 |
| <http://www.w3.org/2000/10/swap/pim/contact#fullName> | 681 |
| <voc://nokia.com/MARS-2/MetaVocabulary/label> | 593 |
| <http://www.news-project.com/Ontology/2008/03/skos_redefined/core#prefLabel> | 591 |
| <http://xmlns.com/foaf/0.1/title> | 540 |
| <http://www.geonames.org/ontology#name> | 524 |
| <http://www.w3.org/2004/02/skos/core#displayablePrefLabel> | 514 |
| <http://xmlns.com/foaf/spec/name> | 514 |
| <http://www.inter2geo.eu/2008/ontology/GeoSkills#defaultCommonName> | 375 |
| <http://rdf.freebase.com/ns/base.business2.governership.title> | 364 |
| <http://dataportal.ucar.edu/schemas/esg.owl#hasResolution> | 346 |
| <http://creativecommons.org/ns#attributionName> | 303 |
| <http://www.w3.org/2006/vcard/ns#label> | 302 |
| <http://rdf.freebase.com/ns/base.foodrecipes.recipe_ingredient.recipe_name> | 296 |
| <http://ontologi.es/rail/vocab#tiploc> | 278 |
| <http://rdf.freebase.com/ns/business.company_name_change.new_name> | 265 |
| <http://www.holygoat.co.uk/owl/redwood/0.1/tags/name> | 233 |
| <http://rdf.freebase.com/ns/base.business2.leadership.title> | 203 |
| <http://downlode.org/rdf/iso-639/schema#name_fr> | 186 |
| <http://downlode.org/rdf/iso-639/schema#name_en> | 186 |
| <http://www.icm.jhu.edu/ontology/ep#hasName> | 183 |
| <http://www.weblab.isti.cnr.it/projects/QH/properties/label> | 182 |
| <http://telegraphis.net/ontology/money/money#name> | 178 |
| <http://projects.apache.org/ns/asfext#title> | 168 |
| <http://telegraphis.net/ontology/money/money#minorName> | 161 |
| <http://www.w3.org/2003/03/glossary-project/lang#name> | 152 |
| <http://4m.cs.hut.fi/ns/onto/lexicon/0.1#description> | 146 |
| <http://www.w3.org/2006/vcard/ns#country-name> | 145 |
| <http://www.w3.org/2001/vcard-rdf/3.0#Orgname> | 131 |
| <http://rdf.freebase.com/ns/location.location.usbg_name> | 127 |
| <http://purl.org/rss/1.0title> | 124 |
| <http://www.co-ode.org/ontologies/amino-acid/2006/05/18/amino-acid.owl#preferredName> | 110 |

TABLE A.1: Properties used across the LOD cloud for labelling, manually selected. The most used labelling property is rdfs:label.

# Appendix B

# Crowdsouring Guidelines for Annotation of the Knowledge Graph Ranking

## B.1 Overview

We want to select the best answers to a set of questions. Please read the question carefully and select the better answer. All answers are just keywords, not full sentences. There might be multiple keywords answering one question.

Please select the best answers to the best of your knowledge. If you are not sure about the correctness of the content, please select additionally the checkbox for (X not sure about the factually correct answer).

## B.2 Steps

You are displayed a question and two answers. Above the answers is the number of the answer. Select from the buttons below the answer based on the number, that answers the question best.

## B.3 Rules & Tips

The answers should be concise, not repeating the same word. The answers should be in Hindi, not in English. If the question asks for multiple answers (for example "list all movies"), more answers are preferred. But those answers should not be repeated, for example "Lion King, Cinderella" is preferred over "Lion King, Cinderella, Lion King (movie)".

## B.4 Examples

### B.4.1 Example (1): Too many ambiguous answers

Question: *Who is the president of the United States of America?*

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| Donald Trump | Trump | Donald Trump | Barack Obama |
|  |  | Barack Obama |  |
|  |  | George Washington |  |

TABLE B.1: Example 1 knowledge graph answers to select from

Answer 1 is factually correct and the answer contains the whole name. While Answer 2 is also correct, it is not as comprehensive as Answer 1. Answer 4 is outdated and therefore not correct, but since it is still a clear, valid answer it is still better than Answer 3. Answer 3 has too many ambiguous answers and can not answer the question correctly.

Therefore, the resulting ranking of this example would be (1, 2, 4, 3)

### B.4.2 Example (2): Repetition of the same answer

Question: *What is the capital of Germany?*

| 1 | 2 |
|---|---|
| Berlin, Germany | Berlin |
| Berlin |  |
| Berlin (city) |  |

TABLE B.2: Example 2 knowledge graph answers to select from

While both Answer 1 and Answer 2 contain the correct answer, Answer 1 contains the same content multiple times. This makes it not as good as the second one.

Therefore, the resulting ranking of this example would be (2, 1).

### B.4.3 Example (3): List of answers

Question: *Show me all movies with Rami Malek.*

| 1 | 2 | 3 |
|---|---|---|
| Papillon | Project X | Papillon |
| Bohemian Rhapsody | Papillon | Bohemian Rhapsody |
| Bohemian Rhapsody (movie) | Bohemian Rhapsody | The Voyage of Doctor Dolittle |
| | The Voyage of Doctor Dolittle | |

TABLE B.3: Example 3 knowledge graph answers to select from

Answer 1 repeats one of the answers (*Bohemian Rhapsody* and *Bohemian Rhapsody (movie)*). Answer 3 does not contain as many unique answers as Answer 2, therefore we can assume it is less comprehensive.

Therefore, the resulting ranking of this example would be (2, 3, 1).

# Appendix C

# Guidelines for the semi-structured interview

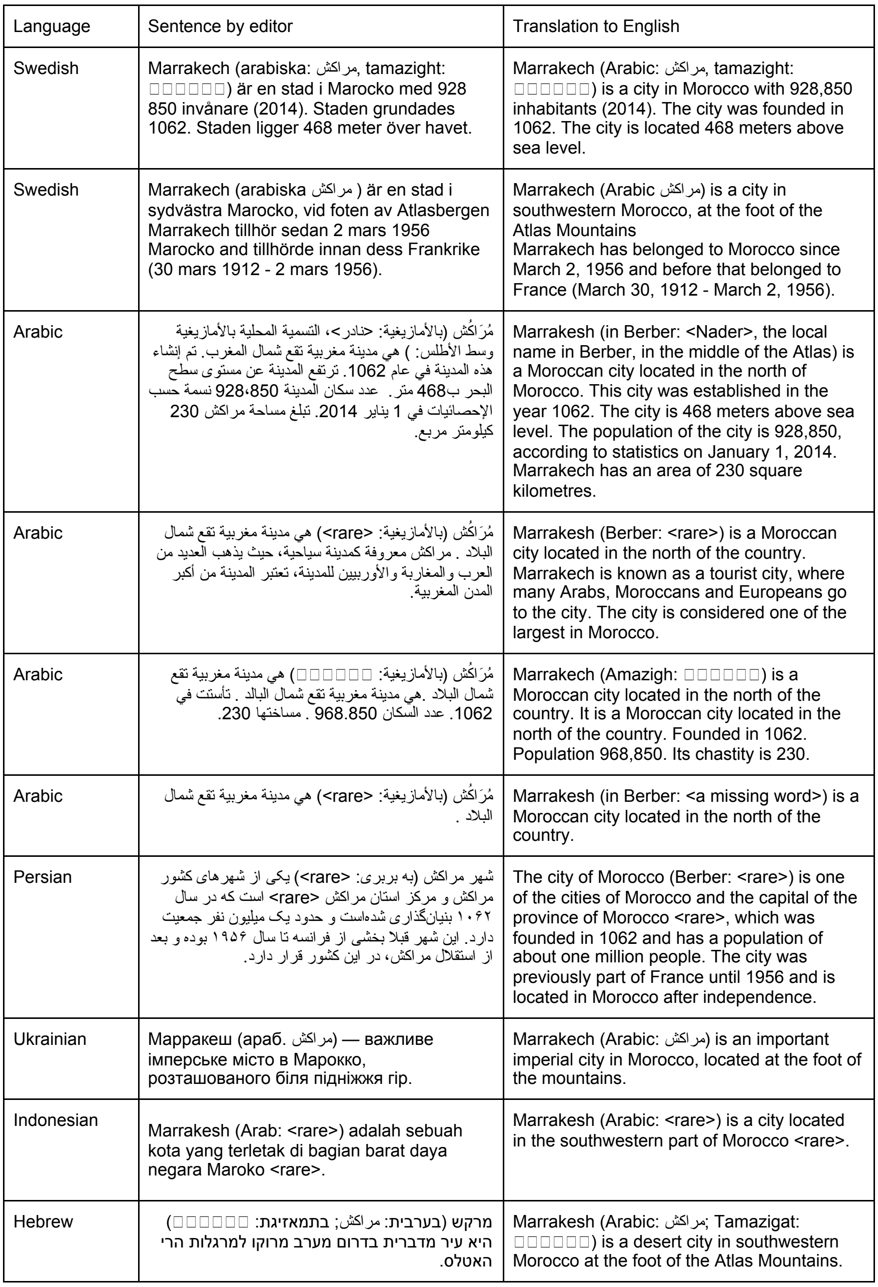
- Opening/Introduction
    - I am Lucie, a researcher at the University of Southampton and I work on this project as part of my PhD research, collaborating with Pavlos Vougiouklis and Elena Simperl.
    - Before I start about the content, I want to ask for your consent to participate in this study, according to the Ethics Committee of the University of Southampton. We will treat your data confidentiality and it will only be stored on the password-protected computer of the researchers. You have the option to withdraw, but we will have to ask you to do that up to 2 weeks after today.
    - We will use the results anonymised to provide insights into the editing of Wikipedia editors and publish the results of the study to a research venue. This experiment will observe your interaction with text and how you edit Wikipedia.
    - Do you agree to participate in this study?
- Demographic Questions
    - Do you read Wikipedia?
        - In which language do you usually read Wikipedia?
        - Do you search topics on Wikipedia or search engines (google)?
        - If you can't find a topic on Wikipedia, what do you do?
    - Do you edit Wikipedia?
        - What is your Wikimedia/Wikipedia username?
        - Which Wikipedia do you mainly contribute to?
        - How long have you contributed to Wikipedia?
        - When you edit, what topics do you choose?
            - Topics you are interested in or topics that you think that is needed?
            - How do you decide, what is interesting/needed?

        - How do you usually start editing?
            - Where do you look up information? (for this topic specifically, in general)
            - Do you draft points and then write text or write the text first?
    - Have you heard of Wikidata?
        - What is Wikidata?
        - What is the relationship between Wikidata and Wikipedia?
        - Have you edited it before?
        - How long are you contributing to Wikidata?

- Description of AP
    - This project is base on the ArticlePlaceholder.
    - The idea is if there is no information about a topic, a user can search on Wikipedia and still get the information on the topic, that is available on Wikidata.
    - We do not create stub articles, everything is displayed dynamically.
    - Have you heard about the ArticlePlaceholder?
- Reading the layout of the AP
    - Is the topic familiar to you? If so, how?
    - If you look at the page, what information do you look at first? (If you want you can point it out with your mouse)
    - What information do you look at after that? (In which order?)
    - What part do you think is taken directly from Wikidata and what is from other sources? Which sources?
    - What information is particularly helpful on this page?
    - What do you think of the text in addition to the rest of the information?
- After editing sentence
    - Where do you start in this case?
    - What information do you miss when editing?
    - Would you prefer to have more or less information given? What type of information?
    - Would you prefer a longer text, even if it has a similar amount of missing information as this one?
- Closing Questions
    - What we do in this project is to generate an introductory sentence from Wikidata triples. We train on this language Wikipedia.
    - What impact do you think this project can have for you as a reader?
    - Do you believe this project will have an impact on your editing?
    - What does it change?
    - Any questions from the interviewee?

# Appendix D

# Sentences editors created in the interviews and English translations.



| Language | Sentence by editor | Translation to English |
|---|---|---|
| Swedish | Marrakech (arabiska: مراكش, tamazight: □□□□□□) är en stad i Marocko med 928 850 invånare (2014). Staden grundades 1062. Staden ligger 468 meter över havet. | Marrakech (Arabic: مراكش, tamazight: □□□□□□) is a city in Morocco with 928,850 inhabitants (2014). The city was founded in 1062. The city is located 468 meters above sea level. |
| Swedish | Marrakech (arabiska مراكش ) är en stad i sydvästra Marocko, vid foten av Atlasbergen Marrakech tillhör sedan 2 mars 1956 Marocko and tillhörde innan dess Frankrike (30 mars 1912 - 2 mars 1956). | Marrakech (Arabic مراكش) is a city in southwestern Morocco, at the foot of the Atlas Mountains Marrakech has belonged to Morocco since March 2, 1956 and before that belonged to France (March 30, 1912 - March 2, 1956). |
| Arabic | مُرَاكُش (بالأمازيغية: <نادر>، التسمية المحلية بالأمازيغية وسط الأطلس: ) هي مدينة مغربية تقع شمال المغرب. تم إنشاء هذه المدينة في عام 1062. ترتفع المدينة عن مستوى سطح البحر ب468 متر. عدد سكان المدينة 928،850 نسمة حسب الإحصائيات في 1 يناير 2014. تبلغ مساحة مراكش 230 كيلومتر مربع. | Marrakesh (in Berber: <Nader>, the local name in Berber, in the middle of the Atlas) is a Moroccan city located in the north of Morocco. This city was established in the year 1062. The city is 468 meters above sea level. The population of the city is 928,850, according to statistics on January 1, 2014. Marrakesh has an area of 230 square kilometres. |
| Arabic | مُرَاكُش (بالأمازيغية: <rare>) هي مدينة مغربية تقع شمال البلاد . مراكش معروفة كمدينة سياحية، حيث يذهب العديد من العرب والمغاربة والأوربيين للمدينة، تعتبر المدينة من أكبر المدن المغربية. | Marrakesh (Berber: <rare>) is a Moroccan city located in the north of the country. Marrakech is known as a tourist city, where many Arabs, Moroccans and Europeans go to the city. The city is considered one of the largest in Morocco. |
| Arabic | مُرَاكُش (بالأمازيغية: □□□□□□) هي مدينة مغربية تقع شمال البلاد .هي مدينة مغربية تقع شمال البالد . تأسثت في 1062. عدد السكان 968.850 . مساختها 230. | Marrakech (Amazigh: □□□□□□) is a Moroccan city located in the north of the country. It is a Moroccan city located in the north of the country. Founded in 1062. Population 968,850. Its chastity is 230. |
| Arabic | مُرَاكُش (بالأمازيغية: <rare>) هي مدينة مغربية تقع شمال البلاد . | Marrakesh (in Berber: <a missing word>) is a Moroccan city located in the north of the country. |
| Persian | شهر مراکش (به بربری: <rare>) یکی از شهرهای کشور مراکش و مرکز استان مراکش <rare> است که در سال ۱۰۶۲ بنیان‌گذاری شده‌است و حدود یک میلیون نفر جمعیت دارد. این شهر قبلا بخشی از فرانسه تا سال ۱۹۵۶ بوده و بعد از استقلال مراکش، در این کشور قرار دارد. | The city of Morocco (Berber: <rare>) is one of the cities of Morocco and the capital of the province of Morocco <rare>, which was founded in 1062 and has a population of about one million people. The city was previously part of France until 1956 and is located in Morocco after independence. |
| Ukrainian | Марракеш (араб. مراكش) — важливе імперське місто в Марокко, розташованого біля підніжжя гір. | Marrakech (Arabic: مراكش) is an important imperial city in Morocco, located at the foot of the mountains. |
| Indonesian | Marrakesh (Arab: <rare>) adalah sebuah kota yang terletak di bagian barat daya negara Maroko <rare>. | Marrakesh (Arabic: <rare>) is a city located in the southwestern part of Morocco <rare>. |
| Hebrew | מרקש (בערבית: مراكش; בתמאזיגת: □□□□□□) היא עיר מדברית בדרום מערב מרוקו למרגלות הרי האטלס. | Marrakesh (Arabic: مراكش; Tamazigat: □□□□□□) is a desert city in southwestern Morocco at the foot of the Atlas Mountains. |

FIGURE D.1: Sentences editors created in the interviews and English translations.

# References

David Abián, F. Guerra, J. Martínez-Romanos, and Raquel Trillo Lado. Wikidata and DBpedia: A Comparative Study. In Julian Szymanski and Yannis Velegrakis, editors, *Semantic Keyword-Based Search on Structured Data Sources - Third International KEYSTONE Conference, IKC 2017, Gdańsk, Poland, September 11-12, 2017, Revised Selected Papers and COST Action IC1302 Reports*, volume 10546 of *Lecture Notes in Computer Science*, pages 142–154. Springer, 2017. . URL https://doi.org/10.1007/978-3-319-74497-1_14.

Bilal Abu-Salih. Domain-specific Knowledge Graphs: A Survey. *Journal of Network and Computer Applications*, 185:103076, 2021.

B. Thomas Adler and Luca de Alfaro. A Content-driven Reputation System for the Wikipedia. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 261–270, 2007. . URL http://doi.acm.org/10.1145/1242572.1242608.

B. Thomas Adler, Krishnendu Chatterjee, Luca de Alfaro, Marco Faella, Ian Pye, and Vishwanath Raman. Assigning Trust to Wikipedia Content. In *Proceedings of the 2008 International Symposium on Wikis, 2008, Porto, Portugal, September 8-10, 2008*, 2008. . URL https://doi.org/10.1145/1822258.1822293.

Nitish Aggarwal, Tamara Polajnar, and Paul Buitelaar. *Cross-Lingual Natural Language Querying over the Web of Data*, pages 152–163. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-38824-8. .

Iñaki Alegria, Unai Cabezón, Unai Fernandez de Betoño, Gorka Labaka, Aingeru Mayor, Kepa Sarasola, and Arkaitz Zubiaga. Reciprocal Enrichment Between Basque Wikipedia and Machine Translation. In *The People's Web Meets NLP, Collaboratively Constructed Language Resources*, pages 101–118. 2013. . URL https://doi.org/10.1007/978-3-642-35085-6_4.

Dean Allemang and James A. Hendler. *Semantic Web for the Working Ontologist - Effective Modeling in RDFS and OWL, Second Edition*. Morgan Kaufmann, 2011. ISBN 978-0-12-385965-5.

Gabor Angeli, Percy Liang, and Dan Klein. A Simple Domain-independent Probabilistic Approach to Generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 502–512, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1870658.1870707.

Judd Antin and Coye Cheshire. Readers are not free-riders: Reading as a form of participation on Wikipedia. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work, CSCW 2010, Savannah, Georgia, USA, February 6-10, 2010*, pages 127–130, 2010. . URL https://doi.org/10.1145/1718918.1718942.

Mihael Arcan and Paul Buitelaar. Ontology Label Translation. In Lucy Vanderwende, Hal Daumé III, and Katrin Kirchhoff, editors, *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 40–46. The Association for Computational Linguistics, 2013. URL https://www.aclweb.org/anthology/N13-2006/.

Mihael Arcan and Paul Buitelaar. Translating Domain-Specific Expressions in Knowledge Bases with Neural Machine Translation. *CoRR*, abs/1709.02184, 2017. URL http://arxiv.org/abs/1709.02184.

Ram G. Athreya, Axel-Cyrille Ngonga Ngomo, and Ricardo Usbeck. Enhancing Community Interactions with Data-Driven Chatbots-The DBpedia Chatbot. In Pierre-Antoine Champin, Fabien L. Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis, editors, *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, pages 143–146. ACM, 2018. . URL https://doi.org/10.1145/3184558.3186964.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary G. Ives. Dbpedia: A Nucleus for a Web of Open Data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007.*, pages 722–735, 2007. . URL https://doi.org/10.1007/978-3-540-76298-0_52.

David Bamman, Brendan O'Connor, and Noah A. Smith. Censorship and deletion practices in Chinese social media. *First Monday*, 17(3):429–440, 2012. . URL http://www.ojphi.org/ojs/index.php/fm/article/view/3943/3169.

Patti Bao, Brent Hecht, Samuel Carton, Mahmood Quaderi, Michael Horn, and Darren Gergle. Omnipedia: Bridging the Wikipedia Language Gap. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1075–1084. ACM, 2012.

Loïc Barrault, Ondrej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. Findings of the 2019 conference on machine translation (WMT19). In Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno-Yepes, Philipp Koehn, André Martins, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana L. Neves, Matt Post, Marco Turchi, and Karin Verspoor, editors, *Proceedings of the Fourth Conference on Machine Translation, WMT 2019, Florence, Italy, August 1-2, 2019 - Volume 2: Shared Task Papers, Day 1*, pages 1–61. Association for Computational Linguistics, 2019. . URL https://doi.org/10.18653/v1/w19-5301.

Wouter Beek, Laurens Rietveld, Hamid R. Bazoobandi, Jan Wielemaker, and Stefan Schlobach. LOD laundromat: A Uniform Way of Publishing Other People's Dirty Data. In Peter Mika, Tania Tudorache, Abraham Bernstein, Chris Welty, Craig A. Knoblock, Denny Vrandecic, Paul Groth, Natasha F. Noy, Krzysztof Janowicz, and Carole A. Goble, editors, *The Semantic*

*Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, volume 8796 of *Lecture Notes in Computer Science*, pages 213–228. Springer, 2014. . URL https://doi.org/10.1007/978-3-319-11964-9_14.

Yonatan Belinkov and Yonatan Bisk. Synthetic and Natural Noise Both Break Neural Machine Translation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018. URL https://openreview.net/forum?id=BJ8vJebC-.

Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific american*, 284 (5):34–43, 2001.

Tim Berners-Lee, Yuhsin Chen, Lydia Chilton, Dan Connolly, Ruth Dhanaraj, James Hollenbach, Adam Lerer, and David Sheets. Tabulator: Exploring and analyzing linked data on the semantic web. In *Proceedings of the 3rd international semantic web user interaction workshop*, volume 2006, page 159. Athens, Georgia, 2006.

Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked Data - The Story So Far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009. . URL https://doi.org/10.4018/jswis.2009081901.

Joshua Evan Blumenstock. Size Matters: Word Count as a Measure of Quality on Wikipedia. In *Proceedings of the 17th International Conference on World Wide Web, WWW 2008, Beijing, China, April 21-25, 2008*, pages 1095–1096, 2008. . URL https://doi.org/10.1145/1367497.1367673.

Nadjet Bouayad-Agha, Gerard Casamayor, and Leo Wanner. Natural Language Generation in the context of the Semantic Web. *Semantic Web*, 5(6):493–513, 2014. . URL http://dx.doi.org/10.3233/SW-130125.

Freddy Brasileiro, João Paulo A. Almeida, Victorio Albani de Carvalho, and Giancarlo Guizzardi. Applying a Multi-level Modeling Theory to Assess Taxonomic Hierarchies in Wikidata. In Jacqueline Bourdeau, Jim Hendler, Roger Nkambou, Ian Horrocks, and Ben Y. Zhao, editors, *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11-15, 2016, Companion Volume*, pages 975–980. ACM, 2016. . URL https://doi.org/10.1145/2872518.2891117.

Dan Brickley and Ramanathan V Guha. RDF vocabulary description language 1.0: RDF schema. 2004.

Elena Cabrio, Julien Cojan, Alessio Palmero Aprosio, Bernardo Magnini, Alberto Lavelli, and Fabien Gandon. Qakis: an Open Domain QA System based on Relational Patterns. In Birte Glimm and David Huynh, editors, *Proceedings of the ISWC 2012 Posters & Demonstrations Track, Boston, USA, November 11-15, 2012*, volume 914 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2012. URL http://ceur-ws.org/Vol-914/paper_24.pdf.

HongYun Cai, Vincent W. Zheng, and Kevin Chen-Chuan Chang. A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. *IEEE Trans. Knowl. Data Eng.*, 30(9):1616–1637, 2018. . URL https://doi.org/10.1109/TKDE.2018.2807452.

Marcirio Chaves and Cássia Trojahn. Towards a Multilingual Ontology for Ontology-driven Content Mining in Social Web Sites. November 2010. URL http://repositorio-cientifico.uatlantica.pt/handle/10884/305.

Gong Cheng and Yuzhong Qu. Searching linked objects with falcons: Approach, implementation and evaluation. *Int. J. Semantic Web Inf. Syst.*, 5(3):49–70, 2009. . URL https://doi.org/10.4018/jswis.2009081903.

Andrew Chisholm, Will Radford, and Ben Hachey. Learning to generate one-sentence biographies from Wikidata. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 633–642, Valencia, Spain, April 2017. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *CoRR*, abs/1406.1078, 2014.

Christopher Cieri, Mike Maxwell, Stephanie M. Strassel, and Jennifer Tracey. Selection Criteria for Low Resource Language Programs. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*. European Language Resources Association (ELRA), 2016. URL http://www.lrec-conf.org/proceedings/lrec2016/summaries/1254.html.

Paul D. Clough, Robert J. Gaizauskas, Scott S. L. Piao, and Yorick Wilks. METER: MEasuring TExt Reuse. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA.*, pages 152–159, 2002.

Benjamin Collier and Julia Bear. Conflict, criticism, or confidence: an empirical examination of the gender gap in Wikipedia contributions. In *CSCW '12 Computer Supported Cooperative Work, Seattle, WA, USA, February 11-15, 2012*, pages 383–392, 2012. . URL http://doi.acm.org/10.1145/2145204.2145265.

World Wide Web Consortium et al. Rdf 1.1 Concepts and Abstract Syntax. 2014.

Jeremy Debattista, Sören Auer, and Christoph Lange. Luzzu - A Methodology and Framework for Linked Data Quality Assessment. *J. Data and Information Quality*, 8(1):4:1–4:32, 2016.

Dennis Diefenbach, Vanessa López, Kamal Deep Singh, and Pierre Maret. Core Techniques of Question Answering Systems over Knowledge Bases: A Survey. *Knowl. Inf. Syst.*, 55(3): 529–569, 2018. . URL https://doi.org/10.1007/s10115-017-1100-y.

Paul DiMaggio, Eszter Hargittai, Coral Celeste, and Steven Shafer. Digital Inequality: From Unequal Access to Differentiated Use. *Social inequality*, pages 355–400, 2004.

Daniel Duma and Ewan Klein. Generating Natural Language from Linked Data. 2013. URL http://www.research.ed.ac.uk/portal/en/publications/generating-natural-language-from-linked-data(c0561842-5e92-40d2-84a1-5b3b3f92063f).html.

Lisa Ehrlinger and Wolfram Wöß. Towards a Definition of Knowledge Graphs. In Michael Martin, Martí Cuquet, and Erwin Folmer, editors, *Joint Proceedings of the Posters and Demos Track of the 12th International Conference on Semantic Systems - SEMANTiCS2016 and the 1st International Workshop on Semantic Change & Evolving Semantics (SuCCESS'16) co-located with the 12th International Conference on Semantic Systems (SEMANTiCS 2016), Leipzig, Germany, September 12-15, 2016*, volume 1695 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016. URL `http://ceur-ws.org/Vol-1695/paper4.pdf`.

Maud Ehrmann, Francesco Cecconi, Daniele Vannella, John Philip McCrae, Philipp Cimiano, and Roberto Navigli. Representing Multilingual Data as Linked Data: the Case of BabelNet 2.0. In *LREC*, pages 401–408, 2014.

Basil Ell and Andreas Harth. A language-independent method for the extraction of RDF verbalization templates. In *INLG 2014 - Proceedings of the Eighth International Natural Language Generation Conference, Including Proceedings of the INLG and SIGDIAL 2014 Joint Session, 19-21 June 2014, Philadelphia, PA, USA*, pages 26–34, 2014.

Basil Ell, Denny Vrandecic, and Elena Paslaru Bontas Simperl. Labels in the Web of Data. In *The Semantic Web - ISWC 2011 - 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011, Proceedings, Part I*, pages 162–176, 2011.

Hady ElSahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon S. Hare, Frédérique Laforest, and Elena Simperl. T-REx: A Large Scale Alignment of Natural Language with Knowledge Base Triples. In Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Kôiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA), 2018. URL `http://www.lrec-conf.org/proceedings/lrec2018/summaries/632.html`.

Kemele M. Endris, Philipp D. Rohde, Maria-Esther Vidal, and Sören Auer. Ontario: Federated Query Processing Against a Semantic Data Lake. In *Database and Expert Systems Applications, DEXA 2019, Linz, Austria, Proceedings, Part I*, 2019.

Mauricio Espinoza, Asunción Gómez-Pérez, and Eduardo Mena. Enriching an Ontology with Multilingual Information. In Sean Bechhofer, Manfred Hauswirth, Jörg Hoffmann, and Manolis Koubarakis, editors, *The Semantic Web: Research and Applications, 5th European Semantic Web Conference, ESWC 2008, Tenerife, Canary Islands, Spain, June 1-5, 2008, Proceedings*, volume 5021 of *Lecture Notes in Computer Science*, pages 333–347. Springer, 2008a. . URL `https://doi.org/10.1007/978-3-540-68234-9_26`.

Mauricio Espinoza, Asunción Gómez-Pérez, and Eduardo Mena. LabelTranslator - A Tool to Automatically Localize an Ontology. In Sean Bechhofer, Manfred Hauswirth, Jörg Hoffmann, and Manolis Koubarakis, editors, *The Semantic Web: Research and Applications, 5th European Semantic Web Conference, ESWC 2008, Tenerife, Canary Islands, Spain, June 1-5, 2008, Proceedings*, volume 5021 of *Lecture Notes in Computer Science*, pages 792–796. Springer, 2008b. . URL `https://doi.org/10.1007/978-3-540-68234-9_60`.

Michael Färber, Frederic Bartscherer, Carsten Menne, and Achim Rettinger. Linked data quality of dbpedia, freebase, opencyc, wikidata, and YAGO. *Semantic Web*, 9(1):77–129, 2018. . URL `https://doi.org/10.3233/SW-170275`.

Mariam Farda-Sarbas and Claudia Müller-Birn. Wikidata from a Research Perspective - A Systematic Mapping Study of Wikidata. *CoRR*, abs/1908.11153, 2019. URL `http://arxiv.org/abs/1908.11153`.

Javier D. Fernández, Wouter Beek, Miguel A. Martínez-Prieto, and Mario Arias. LOD-a-lot - A Queryable Dump of the LOD Cloud. In *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part II*, pages 75–83, 2017.

∀, Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, Freshia Sackey, Rubungo Andre Niyongabo, Ricky Macharm, Perez Ogayo, Orevaoghene Ahia, Musie Meressa Berhe, Mofetoluwa Adeyemi, Masabata Mokgesi-Selinga, Lawrence Okegbemi, Laura Martinus, Kolawole Tajudeen, Kevin Degila, Kelechi Ogueji, Kathleen Siminyu, Julia Kreutzer, Jason Webster, Jamiil Toure Ali, Jade Abbott, Iroro Orife, Ignatius Ezeani, Idris Abdulkadir Dangana, Herman Kamper, Hady Elsahar, Goodness Duru, Ghollah Kioko, Murhabazi Espoir, Elan van Biljon, Daniel Whitenack, Christopher Onyefuluchi, Chris Chinenye Emezue, Bonaventure F. P. Dossou, Blessing Sibanda, Blessing Bassey, Ayodele Olabiyi, Arshath Ramkilowan, Alp Öktem, Adewale Akinfaderin, and Abdallah Bashir. Participatory Research for Low-resourced Machine Translation: A Case Study in African Languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2144–2160, Online, November 2020. Association for Computational Linguistics. . URL `https://www.aclweb.org/anthology/2020.findings-emnlp.195`.

Dimitrios Galanis and Ion Androutsopoulos. Generating Multilingual Descriptions from Linguistically Annotated OWL Ontologies: The NaturalOWL system. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, pages 143–146. Association for Computational Linguistics, 2007.

Luis Galárraga, Christina Teflioudi, Katja Hose, and Fabian M. Suchanek. Fast rule mining in ontological knowledge bases with AMIE+. *VLDB J.*, 24(6):707–730, 2015. . URL `https://doi.org/10.1007/s00778-015-0394-1`.

Luis Antonio Galárraga, Christina Teflioudi, Katja Hose, and Fabian M. Suchanek. AMIE: association rule mining under incomplete evidence in ontological knowledge bases. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 413–422, 2013. . URL `http://doi.acm.org/10.1145/2488388.2488425`.

Lola García Santiago, María Dolores Olvera Lobo, et al. Automatic Web Translators as part of a Multilingual Question-answering (QA) System: Translation of Questions. 2010.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133. Association for Computational Linguistics, 2017. URL `http://aclweb.org/anthology/W17-3518`.

Judith Gaspers, Penny Karanasou, and Rajen Chatterjee. Selecting Machine-Translated Data for Quick Bootstrapping of a Natural Language Understanding System. In Srinivas Bangalore, Jennifer Chu-Carroll, and Yunyao Li, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 3 (Industry Papers)*, pages 137–144. Association for Computational Linguistics, 2018. . URL https://doi.org/10.18653/v1/n18-3017.

Sebastian Gehrmann, Falcon Dai, Henry Elder, and Alexander Rush. End-to-End Content and Plan Selection for Data-to-Text Generation. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 46–56, Tilburg University, The Netherlands, November 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-6505.

R. Stuart Geiger and David Ribes. The Work of Sustaining Order in Wikipedia: The Banning of a Vandal. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work, CSCW 2010, Savannah, Georgia, USA, February 6-10, 2010*, pages 117–126, 2010. . URL https://doi.org/10.1145/1718918.1718941.

Asunción Gómez-Pérez, Daniel Vila-Suero, Elena Montiel-Ponsoda, Jorge Gracia, and Guadalupe Aguado de Cea. Guidelines for Multilingual Linked Data. In *3rd International Conference on Web Intelligence, Mining and Semantics, WIMS '13, Madrid, Spain, June 12-14, 2013*, page 3, 2013.

Jorge Gracia, Elena Montiel-Ponsoda, Philipp Cimiano, Asunción Gómez-Pérez, Paul Buitelaar, and John P. McCrae. Challenges for the Multilingual Web of Data. *J. Web Semant.*, 11:63–71, 2012. . URL https://doi.org/10.1016/j.websem.2011.09.001.

Mark Graham, Bernie Hogan, Ralph K Straumann, and Ahmed Medhat. Uneven Geographies of User-generated Information: Patterns of Increasing Informational Poverty. *Annals of the Association of American Geographers*, 104(4):746–764, 2014.

Spence Green, Jeffrey Heer, and Christopher D. Manning. The Efficacy of Human Post-editing for Language Translation. In *2013 ACM SIGCHI Conference on Human Factors in Computing Systems, CHI '13, Paris, France, April 27 - May 2, 2013*, pages 439–448, 2013. . URL https://doi.org/10.1145/2470654.2470718.

Thomas Gschwind, Christoph Miksovic, Julian Minder, Katsiaryna Mirylenka, and Paolo Scotton. Fast Record Linkage for Company Entities. In *2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, December 9-12, 2019*, pages 623–630. IEEE, 2019. . URL https://doi.org/10.1109/BigData47090.2019.9006095.

Sherzod Hakimov, Soufian Jebbara, and Philipp Cimiano. AMUSE: Multilingual Semantic Parsing for Question Answering over Linked Data. In Claudia d'Amato, Miriam Fernández, Valentina A. M. Tamma, Freddy Lécué, Philippe Cudré-Mauroux, Juan F. Sequeda, Christoph Lange, and Jeff Heflin, editors, *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part I*, volume 10587 of *Lecture Notes in Computer Science*, pages 329–346. Springer, 2017. . URL https://doi.org/10.1007/978-3-319-68288-4_20.

Scott A. Hale. Multilinguals and Wikipedia editing. In *ACM Web Science Conference, WebSci '14, Bloomington, IN, USA, June 23-26, 2014*, pages 99–108, 2014. . URL http://doi.acm.org/10.1145/2615569.2615684.

Scott A. Hale. Cross-language Wikipedia Editing of Okinawa, Japan. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, Seoul, Republic of Korea, April 18-23, 2015*, pages 183–192, 2015. . URL https://doi.org/10.1145/2702123.2702346.

Andreas Harth. Billion Triples Challenge dataset. Downloaded from http://km.aifb.kit.edu/projects/btc-2010/, 2010.

Oktie Hassanzadeh and Mariano P. Consens. Linked Movie Data Base. In *Proceedings of the WWW2009 Workshop on Linked Data on the Web, LDOW 2009, Madrid, Spain, April 20, 2009.*, 2009.

Brent Hecht and Darren Gergle. The Tower of Babel Meets Web 2.0: User-generated Content and Its Applications in a Multilingual Context. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 291–300. ACM, 2010.

Brent Jaron Hecht. *The Mining and Application of Diverse Cultural Perspectives in User-generated Content*. PhD thesis, Northwestern University, 2013.

Stefan Heindorf, Martin Potthast, Benno Stein, and Gregor Engels. Vandalism Detection in Wikidata. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*, pages 327–336, 2016. . URL http://doi.acm.org/10.1145/2983323.2983740.

Monica Heller and Bonnie McElhinny. *Language, capitalism, colonialism: Toward a critical history*. University of Toronto Press, 2017.

Jirí Helmich, Jakub Klímek, and Martin Necaský. Visualizing RDF Data Cubes Using the Linked Data Visualization Model. In *The Semantic Web: ESWC 2014 Satellite Events - ESWC 2014 Satellite Events, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers*, pages 368–373, 2014. . URL https://doi.org/10.1007/978-3-319-11955-7_50.

Ulf Hermjakob, Kevin Knight, and Hal Daumé III. Name Translation in Statistical Machine Translation - Learning When to Transliterate. In *ACL 2008, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA*, pages 389–397, 2008. URL https://www.aclweb.org/anthology/P08-1045/.

Konrad Höffner, Sebastian Walter, Edgard Marx, Jens Lehmann, Axel-Cyrille Ngonga Ngomo, and Ricardo Usbeck. Overcoming Challenges of Semantic Question Answering in the Semantic Web. *Semantic Web Journal*, 2016.

Konrad Höffner, Sebastian Walter, Edgard Marx, Ricardo Usbeck, Jens Lehmann, and Axel-Cyrille Ngonga Ngomo. Survey on Challenges of Question Answering in the Semantic Web. *Semantic Web*, 8(6):895–920, 2017. . URL https://doi.org/10.3233/SW-160247.

Chang Hu, Benjamin B. Bederson, Philip Resnik, and Yakov Kronrod. Monotrans2: A New Human Computation System to Support Monolingual Translation. In *Proceedings of the International Conference on Human Factors in Computing Systems, CHI 2011, Vancouver, BC,* *Canada, May 7-12, 2011*, pages 1133–1136, 2011. . URL https://doi.org/10.1145/1978942.1979111.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5967–5976, 2017. . URL https://doi.org/10.1109/CVPR.2017.632.

Tobias Käfer and Andreas Harth. Billion Triples Challenge dataset. Downloaded from http://km.aifb.kit.edu/projects/btc-2014/, 2014.

Lucie-Aimée Kaffee. *Generating Article Placeholders from Wikidata for Wikipedia: Increasing Access to Free and Open Knowledge*. Bachelor's thesis, HTW Berlin, 2016.

Lucie-Aimée Kaffee and Elena Simperl. The Human Face of the Web of Data: A Cross-sectional Study of Labels. In *Proceedings of the 14th International Conference on Semantic Systems, SEMANTICS 2018, Vienna, Austria, September 10-13, 2018*, pages 66–77, 2018a.

Lucie-Aimée Kaffee and Elena Simperl. Analysis of Editors' Languages in Wikidata. In *Proceedings of the 14th International Symposium on Open Collaboration, OpenSym 2018, Paris, France, August 22-24, 2018*, pages 21:1–21:5, 2018b.

Lucie-Aimée Kaffee, Alessandro Piscopo, Pavlos Vougiouklis, Elena Simperl, Leslie Carr, and Lydia Pintscher. A Glimpse into Babel: An Analysis of Multilinguality in Wikidata. In *Proceedings of the 13th International Symposium on Open Collaboration*, page 14. ACM, 2017.

Lucie-Aimée Kaffee, Hady Elsahar, Pavlos Vougiouklis, Christophe Gravier, Frédérique Laforest, Jonathon S. Hare, and Elena Simperl. Mind the (Language) Gap: Generation of Multilingual Wikipedia Summaries from Wikidata for ArticlePlaceholders. In Aldo Gangemi, Roberto Navigli, Maria-Esther Vidal, Pascal Hitzler, Raphaël Troncy, Laura Hollink, Anna Tordai, and Mehwish Alam, editors, *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings*, volume 10843 of *Lecture Notes in Computer Science*, pages 319–334. Springer, 2018a. . URL https://doi.org/10.1007/978-3-319-93417-4_21.

Lucie-Aimée Kaffee, Hady Elsahar, Pavlos Vougiouklis, Christophe Gravier, Frédérique Laforest, Jonathon S. Hare, and Elena Simperl. Learning to Generate Wikipedia Summaries for Underserved Languages from Wikidata. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 640–645, 2018b. URL https://aclanthology.info/papers/N18-2101/n18-2101.

Lucie-Aimée Kaffee, Kemele M. Endris, and Elena Simperl. When Humans and Machines Collaborate: Cross-lingual Label Editing in Wikidata. In *Proceedings of the 15th International Symposium on Open Collaboration, OpenSym 2019, Skövde, Sweden, August 20-22, 2019*, pages 16:1–16:9, 2019a. . URL https://doi.org/10.1145/3306446.3340826.

Lucie-Aimée Kaffee, Kemele M. Endris, Elena Simperl, and Maria-Esther Vidal. Ranking Knowledge Graphs by Capturing Knowledge about Languages and Labels. In Mayank Kejriwal, Pedro A. Szekely, and Raphaël Troncy, editors, *Proceedings of the 10th International Conference on Knowledge Capture, K-CAP 2019, Marina Del Rey, CA, USA, November 19-21, 2019*, pages 21–28. ACM, 2019b. . URL https://doi.org/10.1145/3360901.3364443.

Lucie-Aimée Kaffee, Pavlos Vougiouklis, and Elena Simperl. Using Natural Language Generation to Bootstrap Missing Wikipedia Articles: A Human-centric Perspective. *Semantic Web*, 2021.

Denys Katerenchuk and Andrew Rosenberg. Rankdcg: Rank-Oordering Evaluation Measure. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, number 978-2-9517408-9-1. European Language Resources Association (ELRA), 2016.

Mayank Kejriwal, Juan F. Sequeda, and Vanessa Lopez. Knowledge graphs: Construction, management and querying. *Semantic Web*, 10(6):961–962, 2019. . URL `https://doi.org/10.3233/SW-190370`.

Aniket Kittur, Bongwon Suh, and Ed H. Chi. Can You Ever Trust a Wiki?: Impacting Perceived Trustworthiness in Wikipedia. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work, CSCW 2008, San Diego, CA, USA, November 8-12, 2008*, pages 477–480, 2008. . URL `https://doi.org/10.1145/1460563.1460639`.

Kevin Knight and Jonathan Graehl. Machine Transliteration. *Computational Linguistics*, 24(4): 599–612, 1998.

Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. Text Generation from Knowledge Graphs with Graph Transformers. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2284–2293, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. . URL `https://www.aclweb.org/anthology/N19-1238`.

Ioannis Konstas and Mirella Lapata. A Global Model for Concept-to-text Generation. *J. Artif. Int. Res.*, 48(1):305–346, October 2013. ISSN 1076-9757. URL `http://dl.acm.org/citation.cfm?id=2591248.2591256`.

Claire Kramsch and HG Widdowson. *Language and Culture*. Oxford University Press, 1998.

Julia Kreutzer, Joost Bastings, and Stefan Riezler. Joey NMT: A Minimalist NMT Toolkit for Novices. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019 - System Demonstrations*, pages 109–114, 2019. . URL `https://doi.org/10.18653/v1/D19-3019`.

Charlie Kritschmar. *Facilitating the Use of Wikidata in Wikimedia Projects with a User-centered Design Approach (Thesis)*. PhD thesis, Bachelor's thesis written at the HTW Berlin in Internationale Medieninformatik, 2016.

Stacey Kuznetsov. Motivations of Contributors to Wikipedia. *SIGCAS Computers and Society*, 36(2):1, 2006. . URL `http://doi.acm.org/10.1145/1215942.1215943`.

Elina Lagoudaki. Translation Editing Environments. In *MT Summit XII: Workshop on Beyond Translation Memories*, 2009.

Ora Lassila and Ralph R Swick. Resource description framework (rdf) model and syntax specification. 1999.

Alon Lavie and Abhaya Agarwal. METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 228–231, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

Rémi Lebret, David Grangier, and Michael Auli. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1203–1213, 2016.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web*, 6(2): 167–195, 2015.

Florian Lemmerich, Diego Sáez-Trumper, Robert West, and Leila Zia. Why the World Reads Wikipedia: Beyond English Speakers. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, February 11-15, 2019*, pages 618–626, 2019. . URL `https://doi.org/10.1145/3289600.3291021`.

DB Lenat and RV Guha. Building Large Knowledge-based Systems: Representation and Inference in the CYC Project. *Artificial Intelligence*, 61(1):4152, 1993.

Adam Lerer, Ledell Wu, Jiajun Shen, Timothee Lacroix, Luca Wehrstedt, Abhijit Bose, and Alex Peysakhovich. PyTorch-BigGraph: A Large-scale Graph Embedding System. In *Proceedings of the 2nd SysML Conference*, Palo Alto, CA, USA, 2019.

William D Lewis and Phong Yang. Building MT for a Severely Under-Resourced Language: White Hmong. *Association for Machine Translation in the Americas, October*, 2012.

Min Li, Marina Danilevsky, Sara Noeman, and Yunyao Li. DIMSIM: An Accurate Chinese Phonetic Similarity Algorithm Based on Learned High Dimensional Encoding. In *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, pages 444–453, 2018.

Zhenhao Li and Lucia Specia. A Comparison on Fine-grained Pre-trained embeddings for the WMT19Chinese-English News Translation Task. In *Proceedings of the Fourth Conference on Machine Translation, WMT 2019, Florence, Italy, August 1-2, 2019 - Volume 2: Shared Task Papers, Day 1*, pages 249–256, 2019. . URL `https://doi.org/10.18653/v1/w19-5324`.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8. Barcelona, Spain, 2004.

Ying Lin, Xiaoman Pan, Aliya Deri, Heng Ji, and Kevin Knight. Leveraging Entity Linking and Related Language Projection to Improve Name Transliteration. In *Proceedings of the Sixth Named Entity Workshop, NEWS@ACL 2016, Berlin, Germany, August 12, 2016*, pages 1–10, 2016. . URL `https://doi.org/10.18653/v1/W16-2701`.

Hairong Liu, Mingbo Ma, Liang Huang, Hao Xiong, and Zhongjun He. Robust Neural Machine Translation with Joint Textual and Phonetic Embedding. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3044–3049. Association for Computational Linguistics, 2019. . URL https://doi.org/10.18653/v1/p19-1291.

Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. Table-to-Text Generation by Structure-Aware Seq2seq Learning. 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16599.

Ekaterina Loginova, Stalin Varanasi, and Günter Neumann. Towards End-to-End Multilingual Question Answering. *Information Systems Frontiers*, pages 1–15, 2020.

Teun Lucassen and Jan Maarten Schraagen. Trust in Wikipedia: How Users Trust Information from an Unknown Source. In *Proceedings of the 4th ACM Workshop on Information Credibility on the Web, WICOW 2010, Raleigh, North Carolina, USA, April 27, 2010*, pages 19–26, 2010. . URL https://doi.org/10.1145/1772938.1772944.

Thang Luong, Ilya Sutskever, Quoc V. Le, Oriol Vinyals, and Wojciech Zaremba. Addressing the Rare Word Problem in Neural Machine Translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 11–19, 2015.

Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Farzaneh Mahdisoltani, Joanna Biega, and Fabian M. Suchanek. Yago3: A Knowledge Base from Multilingual Wikipedias. In *CIDR 2015, Seventh Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, January 4-7, 2015, Online Proceedings*, 2015.

Hongyuan Mei, Mohit Bansal, and Matthew R. Walter. What to talk about and how? Selective Generation using LSTMs with Coarse-to-Fine Alignment. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 720–730, San Diego, California, June 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/N16-1086.

Chris Mellish and R. Dale. Evaluation in the context of natural language generation. *Computer Speech & Language*, 12(4):349–373, 1998. .

Yuval Merhav and Stephen Ash. Design Challenges in Named Entity Transliteration. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 630–640, 2018. URL https://aclanthology.info/papers/C18-1053/c18-1053.

Elena Montiel-Ponsoda, Jorge Gracia, Guadalupe Aguado-de Cea, and Asunción Gómez-Pérez. Representing Translations on the Semantic Web. In *Proceedings of the 2nd International Conference on Multilingual Semantic Web-Volume 775*, pages 25–37. CEUR-WS. org, 2011.

Elena Montiel-Ponsoda, Daniel Vila-Suero, Boris Villazón-Terrazas, Gordon Dunsire, Elena Escolano Rodriguez, and Asunción Gómez-Pérez. Style Guidelines for Naming and

Labeling Ontologies in the Multilingual Web. In *Proceedings of the 2011 International Conference on Dublin Core and Metadata Applications, DC 2011, The Hague, The Netherlands, September 21-23, 2011*, pages 105–115, 2011. URL http://dcpapers.dublincore.org/pubs/article/view/3626.

Jonathan T. Morgan, Siko Bouterse, Heather Walls, and Sarah Stierch. Tea and Sympathy: Crafting Positive New User Experiences on Wikipedia. In *Computer Supported Cooperative Work, CSCW 2013, San Antonio, TX, USA, February 23-27, 2013*, pages 839–848, 2013. . URL https://doi.org/10.1145/2441776.2441871.

Diego Moussallem, Tommaso Soru, and Axel-Cyrille Ngonga Ngomo. THOTH: Neural Translation and Enrichment of Knowledge Graphs. In *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I*, pages 505–522, 2019. . URL https://doi.org/10.1007/978-3-030-30793-6_29.

Mozilla. Internet Health Report v.0.1 2017, 2017. URL https://internethealthreport.org/v01/.

Claudia Müller-Birn, Benjamin Karran, Janette Lehmann, and Markus Luczak-Rösch. Peer-production System or Collaborative Ontology Engineering Effort: What is Wikidata? In *Proceedings of the 11th International Symposium on Open Collaboration, San Francisco, CA, USA, August 19-21, 2015*, pages 20:1–20:10, 2015. . URL http://doi.acm.org/10.1145/2788993.2789836.

Sneha Narayan, Jake Orlowitz, Jonathan T. Morgan, Benjamin Mako Hill, and Aaron D. Shaw. The Wikipedia Adventure: Field Evaluation of an Interactive Tutorial for New Users. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing, CSCW 2017, Portland, OR, USA, February 25 - March 1, 2017*, pages 1785–1799, 2017. URL http://dl.acm.org/citation.cfm?id=2998307.

Roberto Navigli and Simone Paolo Ponzetto. BabelNet: Building a Very Large Multilingual Semantic Network. In Jan Hajic, Sandra Carberry, and Stephen Clark, editors, *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pages 216–225. The Association for Computer Linguistics, 2010. URL https://www.aclweb.org/anthology/P10-1023/.

Axel-Cyrille Ngonga Ngomo, Lorenz Bühmann, Christina Unger, Jens Lehmann, and Daniel Gerber. Sorry, I don't speak SPARQL – Translating SPARQL Queries into Natural Language. In *Proceedings of the 22nd international conference on World Wide Web*, pages 977–988. ACM, 2013.

Finn Nielsen. Lexemes in Wikidata: 2020 status. In *Proceedings of the 7th Workshop on Linked Data in Linguistics (LDL-2020)*, pages 82–86, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-36-8. URL https://www.aclweb.org/anthology/2020.ldl-1.12.

Katherine A. Panciera, Aaron Halfaker, and Loren G. Terveen. Wikipedians Are Born, Not Made: A Study of Power Editors on Wikipedia. In Stephanie D. Teasley, Erling C. Havn, Wolfgang Prinz, and Wayne G. Lutters, editors, *Proceedings of the 2009 International ACM SIGGROUP Conference on Supporting Group Work, GROUP 2009, Sanibel Island, Florida, USA,* *May 10-13, 2009*, pages 51–60. ACM, 2009. . URL https://doi.org/10.1145/1531674.1531682.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA.*, pages 311–318, 2002.

Mikael Parkvall. Världens 100 största språk 2007. *The World's*, 100, 2007.

Molly Jackman Pat Wu. State of connectivity 2015: A report on global internet access, February 2016. URL http://newsroom.fb.com/news/2016/02/state-of-connectivity-2015-a-report-on-global-internet-access/.

Heiko Paulheim. Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods. *Semantic Web*, 8(3):489–508, 2017. . URL https://doi.org/10.3233/SW-160218.

Maria Teresa Pazienza, Armando Stellato, Lina Henriksen, Patrizia Paggio, and Fabio Massimo Zanzotto. Ontology Mapping to Support Multilingual Ontology-based Question Answering. In *Proceedings of the Fourth International Semantic Web Conference (ISWC), Galway, Ireland*, 2005.

Silvio Peroni, David M. Shotton, and Fabio Vitali. Tools for the Automatic Generation of Ontology Documentation: A Task-Based Evaluation. *Int. J. Semantic Web Inf. Syst.*, 9(1): 21–44, 2013. . URL https://doi.org/10.4018/jswis.2013010102.

Carol Peters, Martin Braschler, and Paul D. Clough. *Multilingual Information Retrieval - From Research To Practice*. Springer, 2012. ISBN 978-3-642-23007-3. . URL https://doi.org/10.1007/978-3-642-23008-0.

Peter Pirolli, Evelin Wollny, and Bongwon Suh. So You Know You're Getting the Best Possible Information: a Tool that Increases Wikipedia Credibility. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems, CHI 2009, Boston, MA, USA, April 4-9, 2009*, pages 1505–1508, 2009. . URL https://doi.org/10.1145/1518701.1518929.

Alessandro Piscopo and Elena Simperl. Who Models the World?: Collaborative Ontology Creation and User Roles in Wikidata. *Proc. ACM Hum. Comput. Interact.*, 2(CSCW): 141:1–141:18, 2018. . URL https://doi.org/10.1145/3274410.

Alessandro Piscopo and Elena Simperl. What We Talk About When We Talk About Wikidata Quality: A Literature Survey. In Björn Lundell, Jonas Gamalielsson, Lorraine Morgan, and Gregorio Robles, editors, *Proceedings of the 15th International Symposium on Open Collaboration, OpenSym 2019, Skövde, Sweden, August 20-22, 2019*, pages 17:1–17:11. ACM, 2019. . URL https://doi.org/10.1145/3306446.3340822.

Alessandro Piscopo, Lucie-Aimée Kaffee, Chris Phethean, and Elena Simperl. Provenance Information in a Collaborative Knowledge Graph: An Evaluation of Wikidata External References. In *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part I*, pages 542–558, 2017a. . URL https://doi.org/10.1007/978-3-319-68288-4_32.

Alessandro Piscopo, Christopher Phethean, and Elena Simperl. Wikidatians are Born: Paths to Full Participation in a Collaborative Structured Knowledge Base. In *50th Hawaii International Conference on System Sciences, HICSS 2017, Hilton Waikoloa Village, Hawaii, USA, January 4-7, 2017*, 2017b. URL `http://aisel.aisnet.org/hicss-50/ks/crowd_science/6`.

Alessandro Piscopo, Pavlos Vougiouklis, Lucie-Aimée Kaffee, Christopher Phethean, Jonathon S. Hare, and Elena Simperl. What do Wikidata and Wikipedia Have in Common?: An Analysis of their Use of External References. In *Proceedings of the 13th International Symposium on Open Collaboration, OpenSym 2017, Galway, Ireland, August 23-25, 2017*, pages 1:1–1:10, 2017c. . URL `http://doi.acm.org/10.1145/3125433.3125445`.

Yashaswi Pochampally, Kamalakar Karlapalem, and Navya Yarrabelly. Semi-Supervised Automatic Generation of Wikipedia Articles for Named Entities. In *Wiki@ ICWSM*, 2016.

Ehud Reiter. *Natural Language Generation*, chapter 20, pages 574–598. Wiley-Blackwell, 2010. ISBN 9781444324044. . URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/9781444324044.ch20`.

Ehud Reiter and Anja Belz. An Investigation into the Validity of Some Metrics for Automatically Evaluating Natural Language Generation Systems. *Comput. Linguist.*, 35(4): 529–558, December 2009. ISSN 0891-2017. . URL `http://dx.doi.org/10.1162/coli.2009.35.4.35405`.

Ehud Reiter, Roma Robertson, and Somayajulu Sripada. Acquiring Correct Knowledge for Natural Language Generation. *J. Artif. Int. Res.*, 18:491–516, 2003. ISSN 1076-9757. . URL `https://jair.org/index.php/jair/article/view/10332`.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object Hallucination in Image Captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4035–4045, 2018. URL `https://aclanthology.info/papers/D18-1437/d18-1437`.

Tomás Sáez and Aidan Hogan. Automatically Generating Wikipedia Info-boxes from Wikidata. In Pierre-Antoine Champin, Fabien L. Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis, editors, *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, pages 1823–1830. ACM, 2018. . URL `https://doi.org/10.1145/3184558.3191647`.

Muhammad Saleem, Yasar Khan, Ali Hasnain, Ivan Ermilov, and Axel-Cyrille Ngonga Ngomo. A Fine-Grained Evaluation of SPARQL Endpoint Federation Systems. *Semantic Web*, 7(5):493–518, 2016. . URL `https://doi.org/10.3233/SW-150186`.

John Samuel. Analyzing and Visualizing Translation Patterns of Wikidata Properties. In Patrice Bellot, Chiraz Trabelsi, Josiane Mothe, Fionn Murtagh, Jian-Yun Nie, Laure Soulier, Eric SanJuan, Linda Cappellato, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 9th International Conference of the CLEF Association, CLEF 2018, Avignon, France, September 10-14, 2018, Proceedings*, volume 11018 of *Lecture Notes in Computer Science*, pages 128–134. Springer, 2018. . URL `https://doi.org/10.1007/978-3-319-98932-7_12`.

Christina Sauper and Regina Barzilay. Automatically Generating Wikipedia Articles: A Structure-aware Approach. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 208–216. Association for Computational Linguistics, 2009.

Michael Schmidt, Michael Meier, and Georg Lausen. Foundations of SPARQL Query Optimization. In *Proceedings of the 13th International Conference on Database Theory*, pages 4–33. ACM, 2010.

Abigail See, Peter J. Liu, and Christopher D. Manning. Get To The Point: Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083. Association for Computational Linguistics, 2017. . URL http://www.aclweb.org/anthology/P17-1099.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, 2016. URL http://aclweb.org/anthology/P/P16/P16-1162.pdf.

Nigel Shadbolt, Kieron O'Hara, Tim Berners-Lee, Nicholas Gibbins, Hugh Glaser, Wendy Hall, and m. c. schraefel. Linked Open Government Data: Lessons from Data.gov.uk. *IEEE Intelligent Systems*, 27(3):16–24, 2012. . URL https://doi.org/10.1109/MIS.2012.23.

Anil Kumar Singh. Natural Language Processing for Less Privileged Languages: Where do we come from? Where are we going? In *Third International Joint Conference on Natural Language Processing, IJCNLP 2008, Hyderabad, India, January 7-12, 2008*, pages 7–12. The Association for Computer Linguistics, 2008. URL https://aclanthology.org/I08-3004/.

Amin Sleimi and Claire Gardent. Generating Paraphrases from DBpedia using Deep Learning. In *Proceedings of the 2nd International Workshop on Natural Language Generation and the Semantic Web (WebNLG 2016)*, pages 54–57. Association for Computational Linguistics, 2016. URL http://www.aclweb.org/anthology/W16-3511.

Linda Tuhiwai Smith. *Decolonizing Methodologies: Research and indigenous peoples*. Zed Books Ltd., 2021.

Thomas Steiner. Bots vs. Wikipedians, Anons vs. Logged-Ins (Redux): A Global Study of Edit Activity on Wikipedia and Wikidata. In *Proceedings of The International Symposium on Open Collaboration*, OpenSym '14, pages 25:1–25:7, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3016-9. .

Kyoshiro Sugiyama, Masahiro Mizukami, Graham Neubig, Koichiro Yoshino, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura. An Investigation of Machine Translation Evaluation Metrics in Cross-lingual Question Answering. In *Proceedings of the Tenth Workshop on Statistical Machine Translation, WMT@EMNLP 2015, 17-18 September 2015, Lisbon, Portugal*, pages 442–449. The Association for Computer Linguistics, 2015. . URL https://doi.org/10.18653/v1/w15-3057.

Xiantang Sun and Chris Mellish. An Experiment on "Free Generation" from Single RDF triples. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, pages 105–108. Association for Computational Linguistics, 2007.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to Sequence Learning with Neural Networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc., 2014.

Aaron Swartz. Musicbrainz: A Semantic Web Service. *IEEE Intelligent Systems*, 17(1):76–77, 2002.

Thomas Pellissier Tanon and Lucie-Aimée Kaffee. Property Label Stability in Wikidata: Evolution and Convergence of Schemas in Collaborative Knowledge Bases. In *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, pages 1801–1803, 2018. . URL http://doi.acm.org/10.1145/3184558.3191643.

Thomas Pellissier Tanon, Marcos Dias de Assunção, Eddy Caron, and Fabian M. Suchanek. Demoing Platypus - A Multilingual Question Answering Platform for Wikidata. In *The Semantic Web: ESWC 2018 Satellite Events - ESWC 2018 Satellite Events, Heraklion, Crete, Greece, June 3-7, 2018, Revised Selected Papers*, pages 111–116, 2018. . URL https://doi.org/10.1007/978-3-319-98192-5_21.

Tech Desk, New Delhi. Google Maps makes navigation easier with transliteration in 10 Indian regional languages. *The Indian Express*, 2021. URL https://indianexpress.com/article/technology/tech-news-technology/google-maps-transliteration-how-to-use-10-regional-languages-7163629/.

Harsh Thakkar, Kemele M. Endris, José M. Giménez-García, Jeremy Debattista, Christoph Lange, and Sören Auer. Are Linked Datasets fit for Open-domain Question Answering? A Quality Assessment. In Rajendra Akerkar, Michel Plantié, Sylvie Ranwez, Sébastien Harispe, Anne Laurent, Patrice Bellot, Jacky Montmain, and François Trousset, editors, *Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics, WIMS 2016, Nîmes, France, June 13-15, 2016*, pages 19:1–19:12. ACM, 2016. . URL https://doi.org/10.1145/2912845.2912857.

Chen-Tse Tsai and Dan Roth. Learning Better Name Translation for Cross-Lingual Wikification. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5528–5536, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17318.

Raghavendra Udupa and Mitesh M. Khapra. Improving the Multilingual User Experience of Wikipedia Using Cross-Language Name Search. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 2-4, 2010, Los Angeles, California, USA*, pages 492–500, 2010. URL http://www.aclweb.org/anthology/N10-1073.

Christina Unger, Corina Forascu, Vanessa López, Axel-Cyrille Ngonga Ngomo, Elena Cabrio, Philipp Cimiano, and Sebastian Walter. Question Answering over Linked Data (QALD-4). In Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij, editors, *Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014*, volume 1180 of *CEUR*

*Workshop Proceedings*, pages 1172–1180. CEUR-WS.org, 2014. URL http://ceur-ws.org/Vol-1180/CLEF2014wn-QA-UngerEt2014.pdf.

Shyam Upadhyay, Jordan Kodner, and Dan Roth. Bootstrapping Transliteration with Constrained Discovery for Low-Resource Languages. *CoRR*, abs/1809.07807, 2018. URL http://arxiv.org/abs/1809.07807.

Ricardo Usbeck, Ria Hari Gusmita, Axel-Cyrille Ngonga Ngomo, and Muhammad Saleem. 9th Challenge on Question Answering over Linked Data (QALD-9) (invited paper). In *Joint proceedings of the 4th Workshop on Semantic Deep Learning (SemDeep-4) and NLIWoD4; and 9th Question Answering over Linked Data challenge (QALD-9) co-located with 17th International Semantic Web Conference (ISWC 2018), Monterey, California, United States of America.*, 2018.

Fernanda B. Viégas, Martin Wattenberg, Jesse Kriss, and Frank van Ham. Talk Before You Type: Coordination in Wikipedia. In *40th Hawaii International International Conference on Systems Science (HICSS-40 2007), CD-ROM / Abstracts Proceedings, 3-6 January 2007, Waikoloa, Big Island, HI, USA*, page 78. IEEE Computer Society, 2007. . URL https://doi.org/10.1109/HICSS.2007.511.

Paola Virga and Sanjeev Khudanpur. Transliteration of Proper Names in Cross-Lingual Information Retrieval. In *Proceedings of the Workshop on Multilingual and Mixed-language Named Entity Recognition, NER@ACL 2003, Sapporo, Japan, 2003*, 2003. URL https://aclanthology.info/papers/W03-1508/w03-1508.

Jakob Voss. Measuring Wikipedia. *Proceedings of the ISSI 2005 conference*, 2005.

Pavlos Vougiouklis, Hady Elsahar, Lucie-Aimée Kaffee, Christophe Gravier, Frederique Laforest, Jonathon Hare, and Elena Simperl. Neural Wikipedian: Generating Textual Summaries from Knowledge Base Triples. *Journal of Web Semantics*, 2018.

Pavlos Vougiouklis, Eddy Maddalena, Jonathon S. Hare, and Elena Simperl. Point at the Triple: Generation of Text Summaries from Knowledge Base Triples. *J. Artif. Int. Res.*, 69: 1–31, September 2020. . URL https://doi.org/10.1613/jair.1.11694.

Denny Vrandecic. Capturing Meaning: Toward an Abstract Wikipedia. In Marieke van Erp, Medha Atre, Vanessa López, Kavitha Srinivas, and Carolina Fortuna, editors, *Proceedings of the ISWC 2018 Posters & Demonstrations, Industry and Blue Sky Ideas Tracks co-located with 17th International Semantic Web Conference (ISWC 2018), Monterey, USA, October 8th - to - 12th, 2018*, volume 2180 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2018. URL http://ceur-ws.org/Vol-2180/ISWC_2018_Outrageous_Ideas_paper_6.pdf.

Denny Vrandecic and Markus Krötzsch. Wikidata: A Free Collaborative Knowledgebase. *Commun. ACM*, 57(10):78–85, 2014. . URL https://doi.org/10.1145/2629489.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Trans. Knowl. Data Eng.*, 29(12):2724–2743, 2017. . URL https://doi.org/10.1109/TKDE.2017.2754499.

Leo Wanner, Bernd Bohnet, Nadjet Bouayad-Agha, François Lareau, and Daniel Nicklaß. Marquis: Generation of User-Tailored Multilingual Air Quality Bulletins. *Applied Artificial Intelligence*, 24(10):914–952, 2010. .

Sandra Williams and Ehud Reiter. Generating basic skills reports for low-skilled readers. *Natural Language Engineering*, 14(4):495–525, 2008. .

Michael J Wise. YAP3: Improved Detection Of Similarities In Computer Program And Other Texts. *ACM SIGCSE Bulletin*, 28(1):130–134, 1996.

Sam Wiseman, Stuart Shieber, and Alexander Rush. Challenges in Data-to-Document Generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D17-1239`.

Hao Yang, Gengui Xie, Ying Qin, and Song Peng. Domain Specific NMT based on Knowledge Graph Embedding and Attention. In *2019 21st International Conference on Advanced Communication Technology (ICACT)*, pages 516–521. IEEE, 2019.

S. Yeh, H. Huang, and H. Chen. Precise Description Generation for Knowledge Base Entities with Local Pointer Network. In *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 214–221, Dec 2018. .

Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. Quality Assessment for Linked Data: A Survey. *Semantic Web*, 7(1):63–93, 2016.

Guangyou Zhou, Zhiwen Xie, Tingting He, Jun Zhao, and Xiaohua Tony Hu. Learning the Multilingual Translation Representations for Question Retrieval in Community Question Answering via Non-Negative Matrix Factorization. *IEEE ACM Trans. Audio Speech Lang. Process.*, 24(7):1305–1314, 2016. . URL `https://doi.org/10.1109/TASLP.2016.2544661`.