CKGG: A Chinese Knowledge Graph for High-School Geography Education and Beyond

Yulin Shen, Ziheng Chen, Gong Cheng(✉), and Yuzhong Qu

State Key Laboratory for Novel Software Technology,
Nanjing University, Nanjing, China
{ylshen,chenziheng}@smail.nju.edu.cn, {gcheng,yzqu}@nju.edu.cn

Abstract. As part of a long-term research effort to provide students with better computer-aided education, we create CKGG, a Chinese knowledge graph for the geography domain at the high school level. Using GeoNames and Wikidata as a basis, we transform and integrate various kinds of geographical data in different formats from diverse sources, including gridded temperature data in NetCDF, precipitation data in HDF5, solar radiation data in AAIGrid, polygon data in GPKG, climate and ocean current data in images, and government data in tables. The current version of CKGG contains 1.5 billion triples and is accessible as Linked Data. We also publish a reified version for provenance tracking. We illustrate the potential application of CKGG with a prototype.

Keywords: Knowledge graph · Ontology · Geography

1 Introduction

Computers and artificial intelligence (AI) have fundamentally changed education. As part of a long-term research effort to provide students with better computer-aided and AI-powered education, in recent years we have been particularly focused on the geography subject in China's high-school education. Among others, we employed Semantic Web technology to enhance educational applications including question answering (QA) systems [7,9,17]. One lesson we learned from these research activities is that there is still a lack of high-quality knowledge graphs (KGs) that can cover the core geographical knowledge at the high-school level. Existing geographical KGs suffer from incompleteness or inaccuracy. For example, GeoNames[1] only covers basic geographical data such as location and administrative subdivision. Clinga [6] extracts rich geographical data such as climate from online encyclopedias, but the extracted KG is rather noisy. Indeed, for a QA system to answer high-school geographical questions such as those from [7], we need a KG providing rich and precise geographical knowledge (such as temperature and precipitation) for a large number of locations in the world.

[1] https://www.geonames.org/.

A. Hotho et al. (Eds.): ISWC 2021, LNCS 12922, pp. 429–445, 2021.
https://doi.org/10.1007/978-3-030-88361-4_25

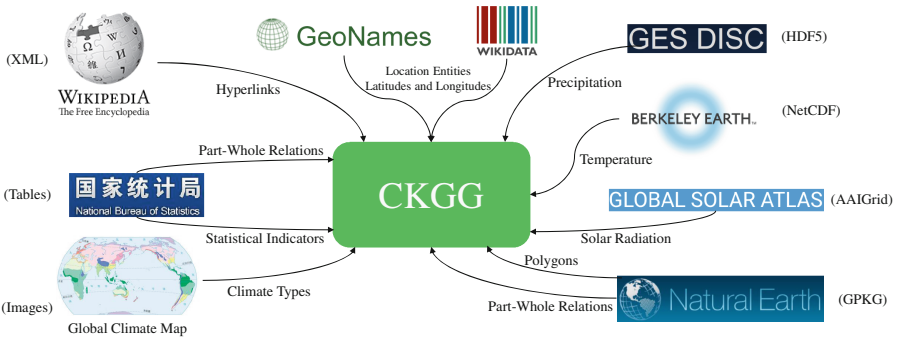


Fig. 1. Data sources integrated into CKGG.

Research Challenge. Despite the inadequacy of KGs, a variety of high-quality geographical data is publicly available in other formats on the Web, but their integration is a non-trivial task. For example, Berkeley Earth and GES DISC have published global temperature and precipitation data, respectively. Such data is in gridded NetCDF or HDF5 formats but is not directly associated with named geographical features (e.g., cities). More challenging examples include the global climate map used in China which is only available as an image. Transforming and integrating such highly heterogeneous data is complicated and labour-intensive.

Contributed Resource. To meet the challenge, we firstly constructed an ontology to cover the core concepts in a popular study guide for geography used in China's high schools. Using this ontology as the schema, we constructed the Chinese Knowledge Graph for Geography (CKGG) to cover the core geographical knowledge at the high-school level. Specifically, we collected and consolidated location entities from GeoNames and Wikidata [14]. Moreover, we used or developed a variety of NLP, math, and GIS tools to integrate heterogeneous data in grids, polygons, images, and tables from diverse sources to enrich location entities with valuable geographical properties including temperature, precipitation, solar radiation, part-whole relations, climate types, and statistical indicators, as depicted in Fig. 1. The ontology and all entities in CKGG are identified by permanent dereferenceable URIs in w3id.[2] The data is also available as RDF dump files on Zenodo. The source code for constructing CKGG and the VoID metadata about CKGG are available on GitHub.[3] All the resources are published under CC BY-SA 4.0. Below we summarize our contribution in the paper.

– We integrate Web data and construct CKGG containing 1.5 billion RDF triples and we publish it following Linked Data best practices. Our preliminary evaluation demonstrates the high quality of location entities in CKGG.

[2] https://w3id.org/ckgg/1.0/.
[3] https://github.com/nju-websoft/CKGG.

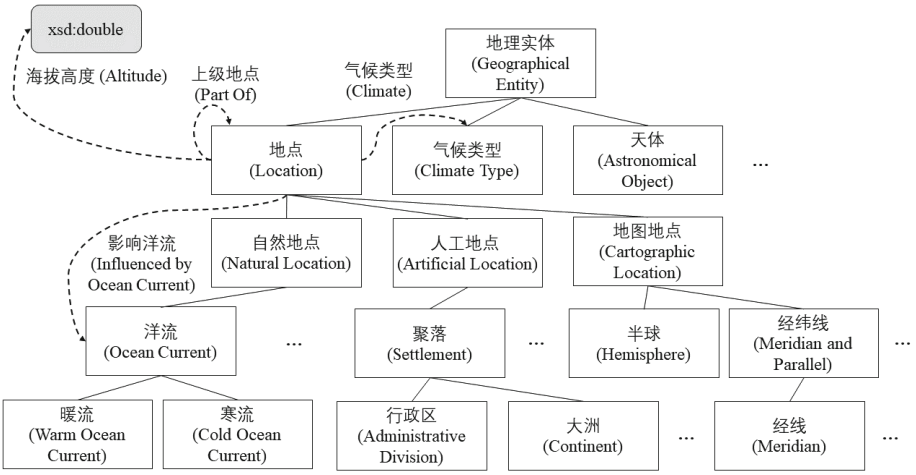


Fig. 2. A sample of the CKGG ontology.

– We present a prototype educational information system based on CKGG. It can be used to search and browse geographical knowledge in and related to CKGG. We also discuss the potential use of CKGG in question answering.

Outline. The remainder of the paper is organized as follows. We introduce the construction of the CKGG ontology in Sect. 2, describe the construction of CKGG in Sect. 3, and show its potential application in Sect. 4. Related work is discussed in Sect. 5. The paper is concluded in Sect. 6.

2 Schema of CKGG

This section describes the construction of an ontology as the schema of CKGG.

2.1 Construction of Ontology

We followed Ontology Development 101 [13] to construct *the CKGG ontology* as the schema of CKGG. In Fig. 2 we illustrate a part of it.

Step 1: Determine the domain and scope of the ontology. CKGG is expected to cover the core geographical knowledge at the high-school level. Since our current focus is on China's high-school education, the CKGG ontology is expected to cover the core concepts in major teaching/learning materials used in China. We selected one of the most popular study guides as the source of concepts.

Step 2: Consider reusing existing ontologies. In addition to the standard RDF and RDFS vocabularies such as `rdf:type` and `rdfs:label`, we considered reusing ontologies that are popular or highly relevant to our domain and scope. We selected two ontologies in the geographical domain: WGS84 Geo Positioning[4] and Clinga [6]. We reused two basic properties in WGS84 Geo Positioning representing the latitude (`wgs84_pos:lat`) and longitude (`wgs84_pos:long`) of a location, and we followed the hierarchy of administrative division types in Clinga.

Step 3: Enumerate important terms in the ontology. We read the study guide and manually identified a list of important concept-level terms. For example, important geographical concepts include "location", a location's "altitude" and "climate type", "ocean current", different types of ocean current such as "warm ocean current". Then we reviewed the identified terms and added a few missing ones, most of which were common concepts such as "public facility".

Step 4: Define the classes and the class hierarchy. We followed a top-down approach. We started with creating a single top-level class: `GeographicalEntity`. Then we specialized it by creating its subclasses such as `Location` (i.e., geographical feature) and `ClimateType`. We further categorized each class. For example, we categorized `Location` into `NaturalLocation`, `ArtificialLocation`, etc. We followed categorizations available in the study guide. For example, we categorized `OceanCurrent` into `WarmOceanCurrent` and `ColdOceanCurrent`.

Step 5: Define the properties of classes. After selecting classes from the list of terms, most of the remaining terms were properties. We attached each property to a class as its `rdfs:domain`. Most properties were attached to `Location` which is a central class in the ontology. For example, `altitude`, `climate`, and `influencedByOceanCurrent` are such properties. In particular, a `Location` can be part of another `Location`, represented by the property `partOf`. We specialized this property by creating its subproperties such as `inCountry`.

Step 6: Define the facets of the properties. We specified the value type or allowed values of each property by defining its `rdfs:range`. The range of a datatype property is an XML Schema datatype. For example, we defined the range of `altitude` as `xsd:double`. For some properties we defined a new datatype by enumerating its allowed values using `owl:oneOf`. For example, `technologyLevel` is chosen from {`very high`, `high`, `medium`, `low`, `very low`}. The range of an object property is a class. For example, `climate` and `influencedByOceanCurrent` relate `Location` to `ClimateType` and `OceanCurrent`, respectively.

Step 7: Create instances. We did not define instances in the ontology but only used it as the schema of CKGG. The creation of instances, i.e., the construction of CKGG, will be described in Sect. 3.

[4] `wgs84_pos`: http://www.w3.org/2003/01/geo/wgs84_pos#.

2.2 Statistics About Ontology

The constructed ontology is online: https://w3id.org/ckgg/1.0/ontology/. It contains 755 classes, 304 datatype properties, and 89 object properties. The maximum depth of a class in the class hierarchy is 10, and the maximum depth of a property in the property hierarchy is 3.

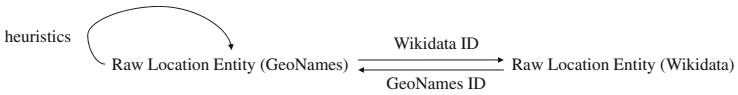


Fig. 3. Inter-source and intra-source matches between raw location entities.

3 Construction of CKGG

This section describes the construction of CKGG. Due to space limitations, we cannot address every detail of CKGG but will focus on its main content.

3.1 Location Entities

Collection of Location Entities. Location entities are central to CKGG. We collected *raw location entities* from GeoNames and Wikidata [14].

From GeoNames (accessed on 10/26/2020) we imported all the 12,051,898 geographical points as raw location entities. For Wikidata (accessed on 11/18/2020) we filtered its entities as follows. We only chose entities from the class of geographical entity (`Q27096213`), and we filtered out entities not having a well-formed value of coordinate location (`P625`) because later we relied on latitude and longitude for integrating data from other sources. Moreover, as our current focus is on China's education, we filtered out entities not having any Chinese label; this operation removed 94.65% of entities. The remaining 412,187 entities were imported as raw location entities.

Consolidation of Location Entities. Raw location entities might refer to the same real-world location entity. We identified both *inter-source matches* and *intra-source matches*, as depicted in Fig. 3. Specifically, we employed both Wikidata IDs (`wkdt`) attached to the entities in GeoNames and GeoNames IDs (`P1566`) attached to the entities in Wikidata to identify inter-source matches. Furthermore, we observed matches between raw location entities both imported from GeoNames. For example, both `1799960` and `1799962` in GeoNames refer to the Nanjing city in China. We identified such intra-source matches using the following heuristics: having at least one common Chinese name, having at least one common word in their English names (to reduce false positives derived from noisy Chinese names), belonging to the same administrative divisions, and located close to each other ($\leq$10 km for `P.PPL`; $\leq$70 km otherwise).

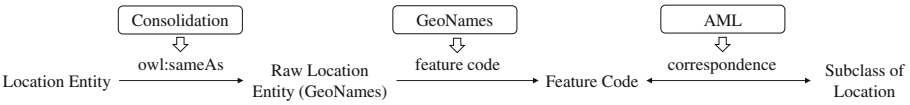


Fig. 4. Typing location entities with subclasses of `Location`.

We constructed a graph representing matches between raw location entities. We consolidated each component of the graph into a location entity in CKGG, and we linked it to each consolidated raw location entity via `owl:sameAs`. For example, the following four raw location entities were consolidated: `1799960` and `1799962` in GeoNames, `Q16666` and `Q28794795` in Wikidata.

There were 8,481,827 trivial components. We consolidated 3,710,324 non-trivial ones, most of which (98.6%) consisted of two raw location entities in an inter-source match. The largest component contained 20 raw location entities.

3.2 Essential Properties

For location entities, we considered type (`rdf:type`), label (`rdfs:label`), latitude (`wgs84_pos:lat`), and longitude (`wgs84_pos:long`) as essential properties.

Type. For each location entity, we assigned it as an instance of `Location`. Moreover, we identified a set of subclasses of `Location` in the CKGG ontology as its specific types. Specifically, we employed a state-of-the-art multilingual ontology matching tool, AgreementMakerLight (AML) [4], to compute correspondences between the hierarchy rooted at `Location` in the CKGG ontology (in Chinese) and the hierarchy of feature codes in GeoNames (in English), and we manually checked the computed correspondences. For each location entity, its specific types were identified by successively following: its raw location entities from GeoNames (if available), the feature codes of these raw location entities, and the corresponding classes of these feature codes. The process is depicted in Fig. 4.

Label. For each location entity, we kept all distinct Chinese names of its raw location entities from GeoNames and Wikidata, and kept the standard English names of its raw location entities from GeoNames. We converted traditional Chinese into simplified Chinese using OpenCC.[5]

Latitude and Longitude. For each location entity, we obtained a set of candidate latitude-longitude pairs from properties of its raw location entities: latitude/longitude in GeoNames, and `P625` in Wikidata. We chose the latitude-longitude pair having the smallest total spherical distance from the other latitude-longitude pairs as the *canonical latitude and longitude* of this location entity.

[5] https://github.com/BYVoid/OpenCC.

3.3 Other Geographical Properties

For each location entity, we imported some useful properties of its raw location entities from GeoNames (e.g., altitude, population). Furthermore, we found other high-quality geographical data from different sources on the Web, but they were published in different formats. We transformed and integrated the following data into CKGG based on the essential properties of location entities.

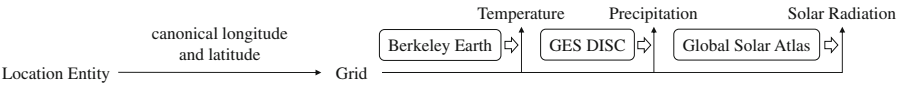


Fig. 5. Associating location entities with gridded temperature, precipitation, and solar radiation.

Fig. 6. Associating location entities with polygons.

Grids to KG. We collected monthly global average temperature data in NetCDF format from Berkeley Earth (accessed on 12/08/2020),[6] monthly global precipitation data in HDF5 format from GES DISC (accessed on 11/17/2020),[7] and daily global solar radiation data in AAIGrid format from Global Solar Atlas (accessed on 12/18/2020).[8] For solar radiation we converted daily totals into annual totals. We also augmented data as follows: for temperature we calculated annual averages based on monthly averages; for precipitation we calculated annual totals based on monthly totals.

To integrate the above data provided for each latitude-longitude grid, for each location entity we identified its grid based on its canonical latitude and longitude, and then added the monthly/annual average temperature, monthly/annual total precipitation, and annual total solar radiation in the grid as its properties. The process is depicted in Fig. 5.

Polygons to KG. For each location entity, we imported the lowest-level administrative divisions of its raw location entities from GeoNames and added them as values of its `partOf` property. To discover and add more part-whole relations, particularly those between locations other than administrative divisions, we exploited their polygon representations.

[6] http://berkeleyearth.lbl.gov/auto/Global/Gridded/Land_and_\penalty-\@MOcean_Alternate_LatLong1.nc.
[7] https://disc.gsfc.nasa.gov/datasets/GPM_3IMERGM_06/summary.
[8] https://api.globalsolaratlas.info/download/World/World_GHI_GISdata\penalty-\@M_LTAy_AvgDailyTotals_GlobalSolarAtlas-v2_AAIGRID.zip.

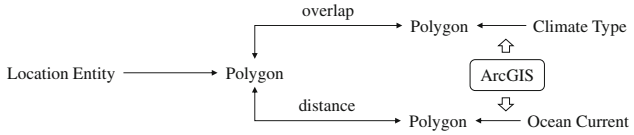


Fig. 7. Associating location entities with climate types and influence of ocean currents in map images.

Observe that a location is generally not a point but has an area. For each location entity, we associated it with a polygon. Specifically, we collected global polygon data in GPKG format from Natural Earth (accessed on 12/08/2020)[9] which contains links to Wikidata entities. For each location entity, its polygon was identified by successively following: its raw location entity from Wikidata (if available), and the corresponding polygon of this raw location entity. The process is depicted in Fig. 6.

We employed polygons to heuristically identify part-whole relations between location entities. For two location entities e_i and e_j associated with polygons $\texttt{plg}(e_i)$ and $\texttt{plg}(e_j)$, we added `partOf` as e_i's property with value e_j if the following two conditions about their areas were satisfied:

$$\texttt{area}(\texttt{plg}(e_i)) < \texttt{area}(\texttt{plg}(e_j)) \quad \text{and} \quad \frac{\texttt{area}(\texttt{plg}(e_i) \cap \texttt{plg}(e_j))}{\texttt{area}(\texttt{plg}(e_i))} \geq 95\% \,. \quad (1)$$

Rather than requiring $\texttt{plg}(e_i) \subset \texttt{plg}(e_j)$, the second condition in Eq. (1) could tolerate noise in the collected polygon data. However, if e_i was not associated with any polygon, we could not use Eq. (1) but instead, we used e_i's canonical latitude and longitude to decide whether $e_i \in \texttt{plg}(e_j)$. This heuristic could be dangerous. For example, Wales was not associated with any polygon and would be considered as part of every polygon containing the canonical latitude and longitude of Wales. To avoid making such errors on important locations, we did not apply this heuristic to countries and first-level administrative divisions.

For each location entity, we also employed its polygon or its canonical latitude and longitude to calculate its distance from the nearest coastline based on the polygons of all coastlines. The distance was added as a property.

Images to KG. Unlike the popular Köppen climate classification available as structured data, the climate classification used in China's teaching/learning materials was only available as a map image. We employed ArcGIS to annotate the map and represent the distributions of climate types as polygons. For each location entity associated with a polygon, we identified its climate types by computing all its overlapping polygons of climate types. For each location entity not associated with any polygon, we computed all the polygons of climate types containing its canonical latitude and longitude. The process is depicted in Fig. 7.

[9] http://naciscdn.org/naturalearth/packages/natural_earth_vector.gpkg.zip.

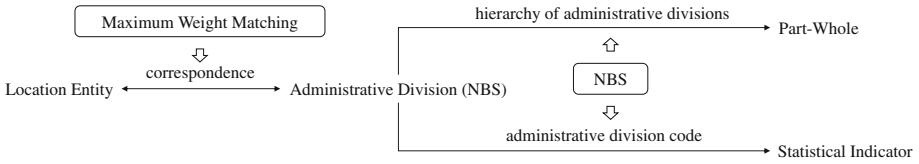


Fig. 8. Associating administrative divisions with part-whole relations and statistical indicators in tables.

Similarly, the global distribution of ocean surface currents was only available as a map image (accessed on 12/18/2020).[10] We also employed ArcGIS to annotate the map and represent ocean currents as polygons. For each location entity e, we used its polygon or its canonical latitude and longitude to calculate its distance from each ocean current. We added `influencedByOceanCurrent` as e's property with all ocean currents as values such that their distances from e were smaller than 1,000 km. The process is depicted in Fig. 7.

Tables to KG. Observe that the part-whole relations obtained from GeoNames and polygons were incomplete. To enrich part-whole relations, particularly those about administrative divisions of China, we collected the official hierarchy of administrative divisions of China in tabular format from the website of National Bureau of Statistics of China (NBS, accessed on 01/12/2021).[11] To compute correspondences between the administrative divisions in CKGG and those at the top four levels of the official hierarchy, we processed the official hierarchy level by level in a top-down manner. For each level, we created an edge-weighted bipartite graph: nodes representing administrative divisions at this level and those in CKGG, and edges connecting administrative divisions having a common name. An edge was assigned a large (resp. small) weight if for the two incident administrative divisions there was a correspondence (resp. mismatch) between their ancestor administrative divisions at higher levels. We computed a maximum weight matching in this graph, from which we derived correspondences at this level. Based on the computed correspondences we enriched part-whole relations with the official hierarchy. The process is depicted in Fig. 8.

Also based on the above correspondences, we associated each administrative division of China with its official administrative division code in NBS. These codes helped us integrate many and various statistical indicators indexed by administrative division code in NBS. Specifically, we collected all statistical indicators about provincial-level administrative divisions of China in tabular format from NBS (accessed on 02/02/2021).[12] To compute correspondences between the properties in the CKGG ontology and the columns in NBS, we calculated the cosine similarity between their TF-IDF vectors and we manually checked the

[10] https://commons.wikimedia.org/wiki/File:Corrientes-oceanicas.png.
[11] http://www.stats.gov.cn/tjsj/tjbz/tjyqhdmhcxhfdm/2020/.
[12] https://data.stats.gov.cn/adv.htm?cn=E0103.

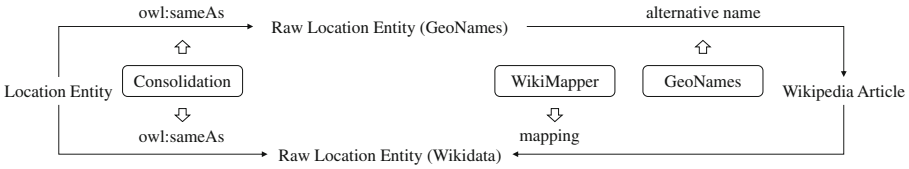


Fig. 9. Associating location entities with Wikipedia articles.

computed correspondences. We obtained ten correspondences including birth rate, crop yield, GDP per capita, and unemployment rate in the latest year. For each provincial-level administrative division of China, we added these latest statistical indicators as its properties by successively following: its corresponding administrative division in NBS, the administrative division code thereof, and its indexed statistical indicators. The process is depicted in Fig. 8.

3.4 Entity Ranking

To facilitate downstream tasks, we associated each location entity with a score representing its salience. Scores could be used to rank location entities in entity search, entity browsing, entity linking, etc. We defined the score of a location entity as the number of hyperlinks to its corresponding articles in Wikipedia.

For each location entity, we obtained its English and Chinese Wikipedia articles by two methods. The first method successively followed its raw location entity from GeoNames (if available), and the Wikipedia links in the alternative names of this raw location entity in GeoNames. The second method successively followed its raw location entity from Wikidata (if available), and the mappings to this raw location entity from Wikipedia article titles. The mappings were computed by WikiMapper.[13] We used precomputed mappings (accessed on 03/16/2021).[14] The process is depicted in Fig. 9.

From the XML dumps of English (accessed on 03/18/2021) and Chinese (accessed on 03/23/2021) versions of Wikipedia[15] we employed Annotated-WikiExtractor[16] to extract all hyperlinks to each of the above Wikipedia article. For each location entity, we associated it with the total number of hyperlinks to its English and Chinese Wikipedia articles as its ranking score.

3.5 Statistics About CKGG

CKGG is available as two sets of RDF dump files on Zenodo containing different versions of the KG: the *standard* version[17] and the *reified* version.[18] All entity

[13] https://github.com/jcklie/wikimapper.
[14] https://public.ukp.informatik.tu-darmstadt.de/wikimapper/.
[15] https://dumps.wikimedia.org/.
[16] https://github.com/jodaiber/Annotated-WikiExtractor.
[17] https://doi.org/10.5281/zenodo.4668711.
[18] https://doi.org/10.5281/zenodo.4678089.

URIs in the namespace https://w3id.org/ckgg/1.0/instances/ are dereferenceable. The standard version contains 1.50B triples. It contains 12.19M location entities, each described by an average of 1.34 types and 121.45 other triples. In this version, for the convenience of downstream applications, we resolved conflicting property values based on predefined rules. For example, for each location entity we only kept its canonical latitude and longitude. The reified version contains 7.49B triples. For this version, we did not resolve conflicts but kept all property values associated with provenance information. Since CKGG is very large, we split it into a set of small dump files by partitioning the triples by properties. Users who are interested in only a few properties do not need to download all the dump files.

3.6 Quality of CKGG

It is difficult, if not impossible, to comprehensively evaluate the quality of a very large and integrated KG like CKGG. Observe that location entities are central to CKGG. Our evaluation was focused on their quality, including their coverage, consolidation, and part-whole relations.

Coverage of Location Entities. Recall that CKGG is expected to cover the core geographical knowledge at the high-school level. We manually identified 295 location entities mentioned in the study guide and checked CKGG's coverage of these entities. We successfully found 233 of them (79%) in CKGG. Among the uncovered ones: 42 (14%) were mainly complex entities (e.g., the drainage basin of the Yangtze River) and did not exist in GeoNames or Wikidata; 20 (7%) could be found in Wikidata but were filtered out due to their missing type, coordinate location, or Chinese label according to the filtering rules we used to collect location entities in Sect. 3.1. We would regard the former case as an open problem. For the latter case, including those entities in CKGG would not benefit downstream applications due to their missing essential properties.

Consolidation of Location Entities. We randomly sampled 100 small components containing two raw location entities and 200 large components containing three or more raw location entities which were consolidated based on inter-source and intra-source matches in Sect. 3.1. We manually checked all matches in each component. We confirmed the correctness of all matches in the sampled 100 small components, demonstrating the high quality of consolidation since small components occupied 98.6% of all non-trivial components. We found incorrect matches in 14 of the sampled 200 large components (7%): 8 (4%) were due to incorrect inter-source matches between GeoNames and Wikidata provided by these sources; only 6 (3%) were related to the heuristics we used to identify intra-source matches.

Part-Whole Relations Between Location Entities. In Sect. 3.3, for each location entity we used heuristics to identify its `partOf` properties based on its

polygon or, if not available, based on its canonical latitude and longitude. For each case we randomly sampled and manually checked 100 `partOf` properties. We confirmed the correctness of all the sampled 200 `partOf` properties, demonstrating the high precision of `partOf` properties in CKGG.

4 Application of CKGG

To illustrate the potential application of CKGG, this section describes the implementation of a prototype educational information system based on CKGG, and also discusses the potential use of CKGG in QA systems.

4.1 Prototype: An Educational Information System

We have implemented a prototype educational information system[19] based on CKGG. Students, teachers, and other potential users can use the system to search and browse geographical knowledge in and related to CKGG.

Location Search. We stored and indexed CKGG in Virtuoso, based on which we provided two search functions: keyword search and proximity search. For keyword search, we relied on Virtuoso's full-text search (`bif:contains`) to find location entities matching an input keyword query. For proximity search, we relied on Virtuoso's spatial search (`bif:st_within`) to find location entities at most 30 km away from an input point on the map. Location entities were ranked by their associated scores computed in Sect. 3.4. Top-ranked location entities were presented, from which the user could select a location entity to browse.

Entity Browsing and Navigation. The browsing interface is illustrated in Fig. 10. On the left-hand side we visualized the location entity on the map provided by OpenStreetMap based on its polygon (if available) or canonical latitude and longitude. On the right-hand side we showed its types and properties. In particular, we employed Chart.js to visualize its monthly precipitation as bars and its monthly temperature as points. The user could navigate to its related entities by clicking its property values to further browse.

For example, by clicking the climate type of a location entity, the user could browse the definition of this climate type and its distribution on the map based on its polygon, as illustrated in Fig. 11. The user could further ask to show top-ranked location entities having this climate type on the map. A similar interface was implemented for browsing ocean currents.

[19] https://w3id.org/ckgg/1.0/demo/.

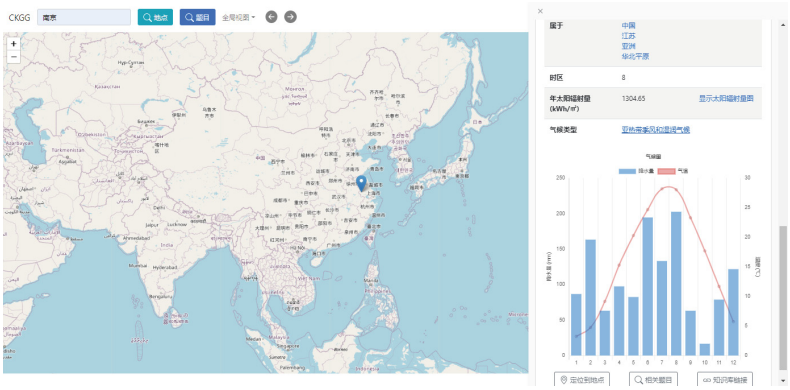


Fig. 10. Browse a location entity.

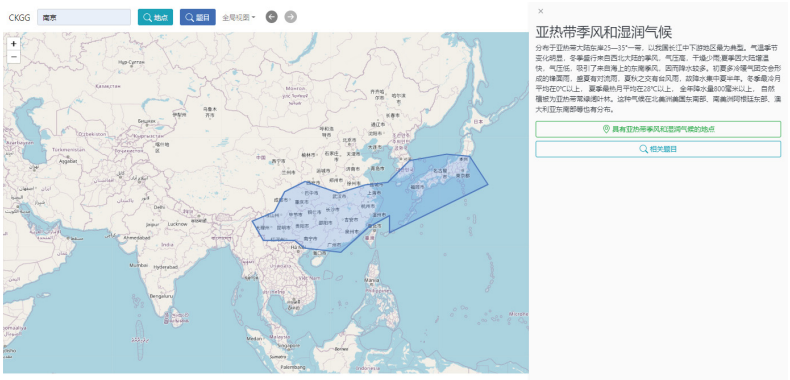


Fig. 11. Browse a climate type.

Question Search and Linking. We imported thousands of high-school geographical questions from existing datasets [7,9], and we employed Apache Lucene to index all questions and support full-text search. Moreover, we employed LTP [1] to recognize mentions of locations in each question and then linked them to location entities in CKGG. When browsing a question, all the location entities mentioned in the question were highlighted, as illustrated in Fig. 12. The user could click a location entity to further browse. It could help the user better understand and answer this question. When browsing a location entity, the user could also ask to show all questions mentioning this location entity. It could help the user better understand this location.

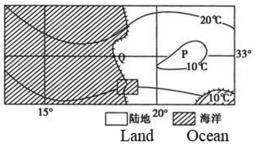
4.2 Discussion: QA Systems

In recent years we have been working on QA systems for answering high-school geographical questions [7,9,17]. A question sampled from the GeoSQA

Fig. 12. Browse a group of questions.

下图示意某地区某月等温线分布，读图完成以下问题。

The figure below shows isotherms in some area in some month. Please answer the following question based on the figure.



Natural language annotations of the image (provided by GeoSQA)

- 图中区域的经度位于15° 和20° 之间，纬度是33° The longitude of the area is between 15° and 20° , and the latitude is 33° .
- Q处位于大陆西岸 Location Q is at the west coast of the land.
- Q处纬度为33° The latitude of location Q is 33° .

Q地与北京气候相比较，一年中____

Comparing the climates of location Q and Beijing, we know that during a year

(A) 两地都雨热同期 Both places feature simultaneous high precipitation and high temperature.

(B) Q地气温较高的月份，北京的气温也较高 When location Q features high temperature, Beijing also features it.

(C) Q地受高压控制的季节，北京盛行偏南风 When location Q is influenced by high atmospheric pressure, south wind prevails in Beijing.

(D) Q地的多雨期与北京基本一致 The rainy periods of location Q and Beijing are basically the same.

Fig. 13. A high-school geographical question sampled from the GeoSQA dataset [7].

dataset [7] is illustrated in Fig. 13. Students in China's high schools would answer this question in the following steps. First, from the isotherms we infer that location Q is in the Southern Hemisphere. Then according to its latitude and longitude we know it is somewhere in South Africa. The west and east coasts of South Africa have different climate types. Note that location Q is at the west coast. Now we can obtain its climate type and compare it with Beijing to answer the question.

Current neural methods can hardly realize the above inference process, as suggested by the extensive experimental results reported in [7]. Symbolic methods are needed where CKGG would exhibit usefulness. For example, based on the latitude and longitude of location Q, we can identify the nearest town in CKGG (i.e., Vredenburg) and then retrieve its climate type, precipitation, and temperature data from CKGG. The obtained knowledge is clearly very useful for determining the correctness of the four options in the question.

Table 1. Comparison between KGs.

	CKGG	Clinga	CrowdGeoKG	GeoNames	Wikidata
Latitude and Longitude	✓	✓	✓	✓	✓
Altitude	✓	X	X	✓	✓
Polygon	✓	X	✓	X	✓
Part-Whole	✓	✓	X	✓	✓
Administrative Division	✓	✓	X	✓	✓
Climate	✓	✓	X	X	X
Temperature	✓	X	X	X	X
Precipitation	✓	X	X	X	X
Solar Radiation	✓	X	X	X	X
Statistical Indicator	✓	✓	X	X	✓

That said, current QA systems are still far away from answering such a question. For example, understanding the complex natural language description in the question is a great challenge. A hybrid neuro-symbolic method is demanded.

5 Related Work

Clinga [6] is one of the first Chinese KGs for the geography domain. It mainly extracted information from online encyclopedias. As a result, for many location entities some important properties are missing. For example, only 12% of the entities in Clinga have a latitude and a longitude, making it difficult to be integrated with other data sources. GeoKG [15] formalizes a geographical ontology but only populates it with a manually created small KG. By contrast, GeoNames provides latitudes and longitudes for a large number of location entities. We used it as a basis for integrating other data. CrowdGeoKG [2] is another geographical KG extracted from OpenStreetMap and Wikidata. However, the kinds of geographical knowledge covered by GeoNames and CrowdGeoKG are limited and insufficient for high-school education. For example, they lack temperature and precipitation data which are core concepts in high-school geography education and are needed for answering the question in Fig. 13.

Encyclopedic KGs such as Wikidata [14] and DBpedia [8] also contain many location entities. We imported location entities from Wikidata as a complement to GeoNames but still, there is a lack of domain-specific knowledge in Wikidata such as temperature and precipitation. We did not use DBpedia because we were concerned about the quality of the data it integrated. For example, in DBpedia, some triples describing the Yunnan Province mistakenly refer to the Baoshan District in Shanghai. Wikidata appeared better in this aspect.

Table 1 compares the above-mentioned KGs. By integrating a variety of domain-specific data from reliable sources, CKGG provides high-quality geographical knowledge and is more comprehensive than existing KGs. It provides

latitudes, longitudes, climate, temperature, and precipitation data, all of which are very useful for answering high-school geographical questions such as the one in Fig. 13 as we discussed in Sect. 4.2.

6 Conclusion and Future Work

By transforming and integrating high-quality geographical data in different formats from diverse sources, we constructed and published CKGG. To the best of our knowledge, it is the most comprehensive geographical KG available on the Web. Although our original goal of constructing this KG was to cover the core geographical knowledge at the high-school level, the current CKGG has the potential to support a wider range of applications. Still, our work has the following limitations which we will address in the future.

Quality of CKGG. We have conducted a preliminary evaluation of CKGG. While the location entities were generally shown to be of high quality, a few errors were identified due to the original data sources and/or our integration methods. We will continue improving the quality of CKGG. In the meantime we will continue extending CKGG to cover broader kinds of geographical knowledge. Indeed, 655 classes and 353 properties defined in the CKGG ontology have not been populated in the current KG. For some properties we have not found any relevant and reliable data source to integrate. We will consider text mining, but accuracy rather than coverage will be our primary concern at all times.

Application of CKGG. We have discussed the potential use of CKGG in QA systems. At the time of writing we implemented two BERT-based QA systems [12,16] incorporating CKGG as domain knowledge. Their experimental results on the GeoSQA dataset [7] were not satisfying: we did not observe significant improvement by using CKGG. Indeed, most properties in the current KG have numerical values, which could not be effectively used by existing embedding-based QA models. Therefore, one possible solution is to further incorporate a rule engine to infer qualitative facts from numerical properties, which will be our future work. We will also explore novel hybrid neuro-symbolic methods.

We have also implemented an educational information system. We will continue extending its functions and evaluate its value for high-school education. Among others, we plan to employ entity summarization techniques [11] to generate an interactive summary for each location entity [10], and to generate comparative and connective summaries for multiple related location entities mentioned in a question [3,5].

Acknowledgements. This work was supported by the National Key Research and Development Program of China (2018YFB1005100).

References

1. Che, W., Feng, Y., Qin, L., Liu, T.: N-LTP: a open-source neural Chinese language technology platform with pretrained models. CoRR abs/2009.11616 (2020)
2. Chen, J., Deng, S., Chen, H.: CrowdGeoKG: crowdsourced geo-knowledge graph. In: Li, J., Zhou, M., Qi, G., Lao, N., Ruan, T., Du, J. (eds.) CCKS 2017. CCIS, vol. 784, pp. 165–172. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-7359-5_17
3. Cheng, G., Xu, D., Qu, Y.: Summarizing entity descriptions for effective and efficient human-centered entity linking. In: WWW 2015, pp. 184–194 (2015). https://doi.org/10.1145/2736277.2741094
4. Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I.F., Couto, F.M.: The AgreementMakerLight ontology matching system. In: Meersman, R., et al. (eds.) OTM 2013. LNCS, vol. 8185, pp. 527–541. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41030-7_38
5. Gunaratna, K., Yazdavar, A.H., Thirunarayan, K., Sheth, A.P., Cheng, G.: Relatedness-based multi-entity summarization. In: IJCAI 2017, pp. 1060–1066 (2017). https://doi.org/10.24963/ijcai.2017/147
6. Hu, W., et al.: Clinga: bringing Chinese physical and human geography in linked open data. In: Groth, P., et al. (eds.) ISWC 2016. LNCS, vol. 9982, pp. 104–112. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46547-0_11
7. Huang, Z., et al.: GeoSQA: a benchmark for scenario-based question answering in the geography domain at high school level. In: EMNLP-IJCNLP 2019, pp. 5865–5870 (2019). https://doi.org/10.18653/v1/D19-1597
8. Lehmann, J., et al.: DBpedia - a large-scale, multilingual knowledge base extracted from Wikipedia. Semant. Web **6**(2), 167–195 (2015). https://doi.org/10.3233/SW-140134
9. Li, X., Sun, Y., Cheng, G.: TSQA: tabular scenario based question answering. In: AAAI 2021 (2021)
10. Liu, Q., Chen, Y., Cheng, G., Kharlamov, E., Li, J., Qu, Y.: Entity summarization with user feedback. In: Harth, A., et al. (eds.) ESWC 2020. LNCS, vol. 12123, pp. 376–392. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-49461-2_22
11. Liu, Q., Cheng, G., Gunaratna, K., Qu, Y.: Entity summarization: state of the art and future challenges. J. Web Semant. **69**, 100647 (2021). https://doi.org/10.1016/j.websem.2021.100647
12. Liu, W., et al.: K-BERT: enabling language representation with knowledge graph. In: AAAI 2020, pp. 2901–2908 (2020)
13. Noy, N.F., McGuinness, D.L.: Ontology development 101: a guide to creating your first ontology. Technical report, KSL-01-05, Stanford University (2001)
14. Vrandecic, D., Krötzsch, M.: Wikidata: a free collaborative knowledge base. Commun. ACM **57**(10), 78–85 (2014). https://doi.org/10.1145/2629489
15. Wang, S., Zhang, X., Ye, P., Du, M., Lu, Y., Xue, H.: Geographic knowledge graph (GeoKG): a formalized geographic knowledge representation. ISPRS Int. J. Geo Inf. **8**(4), 184 (2019). https://doi.org/10.3390/ijgi8040184
16. Yang, A., et al.: Enhancing pre-trained language representations with rich knowledge for machine reading comprehension. In: ACL 2019, vol. 1, pp. 2346–2357 (2019). https://doi.org/10.18653/v1/p19-1226
17. Zhang, Z., et al.: Towards answering geography questions in gaokao: a hybrid approach. In: Zhao, J., Harmelen, F., Tang, J., Han, X., Wang, Q., Li, X. (eds.) CCKS 2018. CCIS, vol. 957, pp. 1–13. Springer, Singapore (2019). https://doi.org/10.1007/978-981-13-3146-6_1