A grammar for interpreting geo-analytical questions as concept transformations

Haiqi Xu, Enkhbold Nyamsuren, Simon Scheider & Eric Top

Published online: 30 May 2022.

Submit your article to this journal ⊠

Article views: 1887

View related articles ⊠

View Crossmark data ⊠

Citing articles: 2 View citing articles ⊠

Taylor & Francis
Taylor & Francis Group

RESEARCH ARTICLE

ə OPEN ACCESS   Check for updates

A grammar for interpreting geo-analytical questions as concept transformations

Haiqi Xu, Enkhbold Nyamsuren, Simon Scheider 🅓 and Eric Top

Department of Human Geography and Spatial Planning, Utrecht University, Utrecht, Netherlands

ABSTRACT

Geographic Question Answering (GeoQA) systems can automatically answer questions phrased in natural language. Potentially this may enable data analysts to make use of geographic information without requiring any GIS skills. However, going beyond the retrieval of existing geographic facts on particular places remains a challenge. Current systems usually cannot handle geo-analytical questions that require GIS analysis procedures to arrive at answers. To enable *geo-analytical QA*, GeoQA systems need to interpret questions in terms of a transformation that can be implemented in a GIS workflow. To this end, we propose a novel approach to question parsing that interprets questions in terms of core concepts of spatial information and their functional roles in context-free grammar. The core concepts help model spatial information in questions independently from implementation formats, and their functional roles indicate how concepts are transformed and used in a workflow. Using our parser, geo-analytical questions can be converted into expressions of concept transformations corresponding to abstract GIS workflows. We developed our approach on a corpus of 309 GIS-related questions and tested it on an independent source of 134 test questions including workflows. The evaluation results show high precision and recall on a gold standard of concept transformations.

1. Introduction

Research on question answering (QA) systems has often been centered on retrieving factual knowledge from documents or knowledge bases, with limited attention being directed to procedures that create such retrievable and declarative knowledge (Fader *et al.* 2014, Chen *et al.* 2020, Buzaaba and Amagasa 2021, Kapanipathi *et al.* 2021). This retrieval-oriented approach also dominates current geographic QA systems, which are capable of answering a question like 'Where is Amsterdam?', but not 'Where are the clusters of elderly people in Amsterdam?'(Chen *et al.* 2013, Punjani *et al.* 2018, Mai

CONTACT Haiqi Xu ✉ h.xu1@uu.nl 🖃 Department of Human Geography and Spatial Planning, Utrecht University, Utrecht, Netherlands

et al. 2019a, 2019b, Hamzei *et al.* 2022). The answer to the latter question is not a known fact, but needs to be estimated using geo-spatial analysis. In other words, geo-spatial analysis requires procedural knowledge for generating declarative knowledge (Nyerges 1995, Ten Berge and Van Hezewijk 1999). In geographical QA, little attention has been devoted to investigating such procedural knowledge that is central for analysis with Geographic Information Systems (GIS), such as ArcGIS[1] or QGIS.[2] Yet, full-fledged geographical QA requires procedural knowledge for interpreting the problem and composing analytic workflows (Scheider *et al.* 2020b, Xu *et al.* 2020).

When formulating a task in terms of a question, geo-spatial analysts implicitly encode information about transformations of underlying concepts into the question. The task is to recover this information from a question to narrow down relevant data and GIS tools that need to be applied in a meaningful order to generate the requested answer. For example, suppose a question is 'From where are wind farms visible in Amsterdam?', what we need is an input representing 'wind farms' and 'Amsterdam' and a tool that implements visibility analysis to determine areas of visibility. Furthermore, we need to bring these resources into a logical order: 'Wind farms' should be the input of the visibility tool, while 'Amsterdam' defines the spatial extent of the visibility analysis. This problem was called '*indirect QA*' in Scheider *et al.* (2020b). If machines could extract this information from a question, this would enable QA systems to generate relevant answers on their own. The goal of this paper is to propose a method that enables such machine interpretation of geo-analytical questions for 'indirect QA'. Our study is limited to English language questions for GIS tasks. However, even with this limitation, the method needs to address multiple challenges.

Firstly, the same analytical task can be expressed with different syntactic structures. For example, the visibility question mentioned above can be rephrased as 'Where can people see the wind farms in Amsterdam?'. Though the questions have a different syntax, the semantics required for generating the workflow remains the same. As a first step, in this article, we do not yet address the syntactic diversity of geo-analytical questions. We rather aim at capturing the semantic variety of conceptual transformations in terms of their most prevalent syntactic patterns. These patterns need to be translated into transformation steps on a conceptual level. For this purpose, we explore question patterns and show how the syntactic variety can be reduced to one prevalent pattern that captures an intended transformation. The intended conceptual transformations form a basis for later handling syntactic variety, too.[3]

Secondly, since geo-analytical questions need to be interpreted differently than other kinds of geographic questions [e.g. place questions (Hamzei *et al.* 2019), geographic questions (Huang *et al.* 2019, Mai *et al.* 2021)], the current techniques from QA systems and natural language processing (NLP) are insufficient for our purpose (Scheider *et al.* 2020b). We need to derive the latent semantics of a transformation process that could be implemented in GIS workflows. For example, a terrain model is necessary for answering the visibility question, but this is not explicit in the question. The approach furthermore needs to distinguish functional phrases from auxiliary content in questions, such as 'wind farms', 'Amsterdam', and 'visibility'.

Thirdly, information about the types of data and operations needed for constructing a workflow is not directly contained in the questions. When interpreting geo-analytical

questions, analysts rather create conceptual abstractions of requirements necessary for composing a workflow. These requirements are abstract from common data types we use in GIS, such as vector and raster (Kuhn and Ballatore 2015). Kuhn refers to these abstractions as core concepts of spatial information (Kuhn 2012). Example core concepts are *Object* and *Field*. For instance, the wind farms that are input to the visibility tool are interpreted as objects that might be represented as points, whereas the continuous height surface is interpreted as a field that might be represented as a surface raster or in terms of a contour map. In this way, core concepts narrow down GIS resources in terms of meaning and computational constraints (Scheider *et al.* 2020a). Furthermore, in one question we may encounter multiple instances of a core concept, which need to be distinguished by their roles. The role of a core concept can be used to specify ordering constraints over a workflow. For example, in the visibility question, 'wind farms' and 'Amsterdam' are both *Objects*, yet they play very different roles in the workflow: one functions as an input (the object to be observed), the other as a spatial extent. To the best of our knowledge, the current QA systems and NLP techniques are hardly capable of achieving any of these steps. To address the first step, we might reuse NLP models (Diefenbach *et al.* 2018, Xu *et al.* 2020), but we lack so far a *model of semantic variety* in the first place, which might be used for supervised learning. Furthermore, approaches for handling the second and third steps are entirely unknown.

To address these challenges, in this paper, we design and test an approach to parsing questions in terms of core concepts and interpreting their functional roles in terms of concept transformations. Our contributions are as follows:

1. *extract concepts* from geo-analytical questions for constructing GIS workflows.
2. recognize the *functional roles* of the extracted concepts based on syntactic question patterns.
3. propose a *model* for possible concept transformations.
4. *categorize questions* based on the kind of concept transformations they encode.
5. design a *geo-analytical grammar* to capture variations of concept transformations across various question categories using prevalent syntactic structures.
6. test the grammar on a corpus of geo-analytical questions that come with answer workflows.

In this paper, we first illustrate the scientific challenge using a simple scenario in Section 2. Based on the scenario, we discuss the relevance of core concepts of spatial information and functional roles in Sections 2.2 and 2.3. Afterward, we introduce our methodology, including a question corpus in Section 3. Then we discuss the variety of question categories in terms of concept transformations occurring in our corpus (Section 4.1), as well as automatically extracted transformations (Section 4.2). Finally, we test and discuss how well the grammar captures manual concept interpretations using a gold standard in Section 5 before we conclude.

2. Core concepts and their functional roles in geo-analytical questions

In this section, we first illustrate the challenges using three geo-analytical questions and answer workflows. Based on these examples, we explain why core concepts take on different functional roles important for interpreting questions in terms of transformations and workflows.

2.1. What is the situation of living for the elderly in Amsterdam?

We start with a scenario taken from a GIS course at VU Amsterdam.[4] The scenario is about assessing the distribution of the elderly and the accessibility of parks for each postcode area (level 4) of Amsterdam. All the input geodata and statistic data are available from open data portals: the portal of the City of Amsterdam[5] and Centraal Bureau voor de Statistiek (CBS).[6] ArcGIS Pro Model Builder[7] was used to build executable workflows. The following urban environmental factors are considered:

1. To assess the distribution of the elderly in Amsterdam, we ask '*What is the proportion of people older than 65 for each PC4 area in Amsterdam?*'.

    The given input data is a vector polygon layer on the level of 'Buurt' (neighborhood) in Amsterdam; each polygon has an attribute value indicating the proportion of people over 65 (see Figure 1(a)). The geo-analytic workflow (Figure 1(c)) interpolates the density of the elderly from neighborhoods into PC4 areas (Figure 1(b)).

2. To assess the distribution of parks in Amsterdam, we ask '*What is the density of parks for each PC4 area in Amsterdam?*'

    The given data is a land use vector dataset that contains park areas (see Figure 2(a)). The workflow (Figure 2(c)) first generates a raster denoting locations covered by parks and then measures the area covered by parks within PC4 objects to compute park density per PC4 area are shown in Figure 2(b).

3. Distance analysis could also matter to answer the main question, therefore we also ask '*What is the average Euclidean distance to parks for each PC4 area in Amsterdam?*'.

    Starting from a land use vector dataset, Euclidean distance[8] in Figure 3(c) computes the straight-line distance to the closest park for each cell in the extent. The distance is averaged within PC4 areas, as shown in Figure 3(b).

2.2. Using core concepts to interpret geo-analytical questions

How can core concepts serve as a way to interpret the questions mentioned above? We start by introducing what core concepts are, how they can be quantified, and discuss the challenges of interpreting questions in this way.

(a)

(b)

Figure 1. Input data, ArcGIS workflow, and answer map for answering 'What is the proportion of people older than 65 for each PC4 area in Amsterdam?' (a) The density of people over 65 in neighborhoods of Amsterdam, provided by CBS.[19] (b) Map of the density of people over 65 for each PC4 area in Amsterdam. (c) ArcGIS workflow computing the density of people over 65 for each PC4 area in Amsterdam.
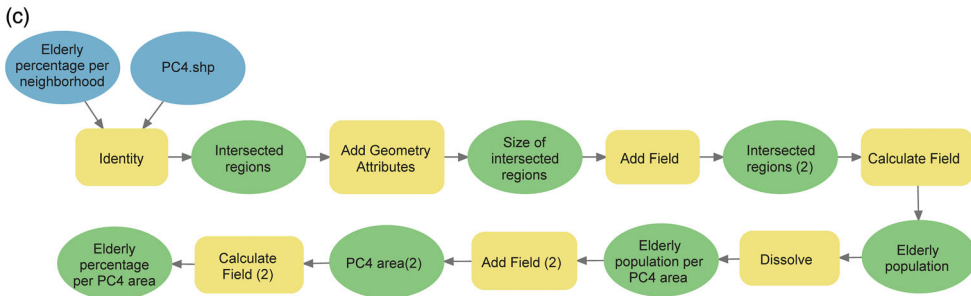
Figure 1. Continued.

2.2.1. Core concepts of spatial information

The most recent version[9] (Kuhn and Ballatore 2015) includes five content related core concepts of spatial information that generalize spatial information in terms of the kinds of underlying phenomena:

Location describes where a spatial phenomenon is, according to a spatial reference system in GIS (Kuhn 2012). *Location* information can be used to answer where questions, as well as to compute geometric properties (e.g. size) and distribution patterns (e.g. cluster) of spatial phenomena.

Field is a spatial phenomenon that has continuous values (e.g. elevation) or homogeneous values (e.g. land use) at arbitrary locations in space. We can aggregate field values into statistical zones (Figure 3), and interpolate missing values from field point measurements (Scheider *et al*. 2020a).

Object is a spatially bounded phenomenon that has its own identity and quality. For example, administrative units are considered *objects* because each of them has a boundary, a population as quality, and a distinct identity. Although both *object* and *nominal field* (e.g. landuse) can be represented by a vector geometry, the object's quality is not homeomerous, which means parts of the object do not have the same quality as the whole (Scheider *et al*. 2020a). Therefore areal interpolation (Figure 1(c)) is needed to transform an object's quality into incongruent object layers.

Event is considered a temporally bounded entity similar to objects. For example, a trip of a person has a location and a quality, such as duration. Common operations for events include counting the number of events, computing density, and measuring spatio-temporal patterns.

Network is a quantifiable relation between objects, such as a travel distance between two cities or migrant flows between two countries. Networks also tell us whether two objects are connected.

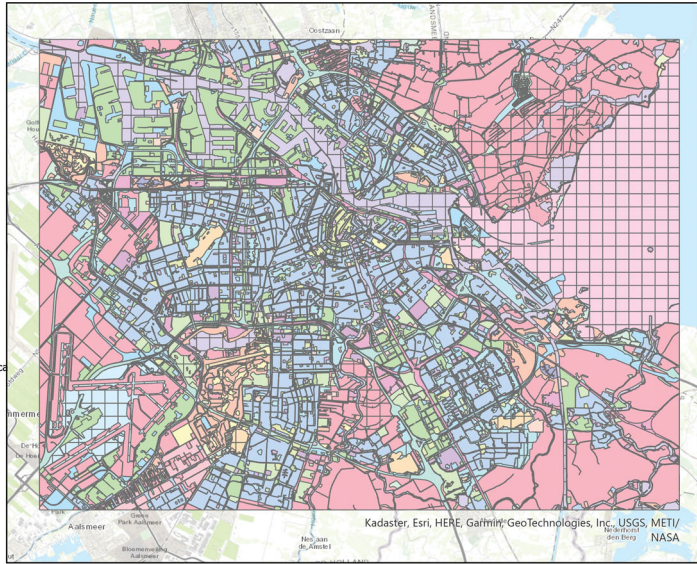### 2.2.2. Amounts and proportions of core concepts

Important types of transformations are based on core concepts, yet are not covered by any of the five content concepts. For example, 'the number of crime cases near Amsterdam' cannot be attributed to any specific core concept, yet is central to capture transformations. For this reason, we introduce two further concepts, namely *amount* and *proportion*.

(a)

(b)

Figure 2. Input data, ArcGIS workflow, and answer map for answering 'What is the density of parks for each PC4 area in Amsterdam?' (a) Land use map of Amsterdam, provided by the City of Amsterdam.[20] (b) Map of the density of parks for each PC4 area in Amsterdam. (c) ArcGIS workflow computing the density of parks for each PC4 area in Amsterdam.

Figure 2. Continued.

Amounts quantify core concepts or their qualities. According to the way of measuring amounts, two types can be distinguished, namely *content amount* and *coverage amount*. *Content amounts* represent aggregation results of core concepts and their qualities in a region given as control. For discrete core concepts, such as event and object, content amount refers to a count, while for continuous fields or discrete concept qualities, the content a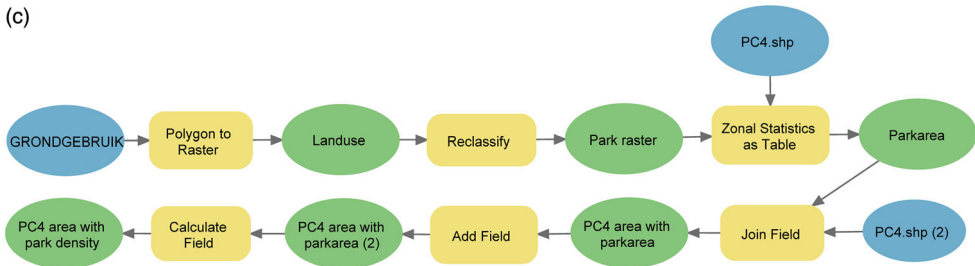mount is measured by a sum, average or median. For example, average Euclidean distance (field content amount) in PC4 areas or the sum of household income (object quality content amount) in a city. Different content amounts can be on different measurement levels. Discrete object and event content amounts are always measured on a count scale level. Field and core concept qualities are measured on an extensive ratio level (ERA) (Scheider and Huisjes 2019).

Coverage amounts quantify the spatial 'coverage' of core concepts. We distinguish two types: 1) size of core concepts; 2) spatial distribution of core concepts. Examples of the former are the total area of forest or the length of highways. An example of distribution is a central feature[10] of banks or clusters of emergency calls. Central features and spatial clusters cannot be simply mapped to a measurement level. A central feature is still an object, but with the additional information that this object is centrally located. Therefore, we use a measurement level 'loc' for distribution measurements to distinguish it from size.

Proportion is a ratio of quantity generated from amounts. The different combinations of the content amount and coverage amount generate different types of proportions. For example, 'How much of a city is covered by green space?' asks for a proportion of coverage amount (green space area) with respect to another coverage amount (city area). Population density is a proportion of content amount (population count) relative to coverage amount (city area). Crime rate is a ratio of crime count and population count, thus a ratio between two content amounts. Unlike amount, the proportion is measured on an intensive ratio scale (IRA) (Scheider and Huisjes 2019).

2.2.3. Identifying core concepts in questions
To illustrate the relevance of core concepts for interpreting geo-analytical questions in GIS workflows, the first step is to recognize and annotate them in questions:

1. What is the density[(Proportion IRA)] of people[(Object)] older than 65[(Object Quality Interval)] for each PC4 area[(Object)] in Amsterdam[(Object)]?

(a)

Legend

GRONDGEBRUIK
- Bedrijventerrein
- Begraafplaats
- Bos
- Bouwterrein
- Dagrecreatief terrein
- IJsselmeer/Markermeer
- Open droog-natuurlijk terrein
- Open nat-natuurlijk terrein
- Overig agrarisch terrein
- Overig water breder dan 6 m.
- Park en plantsoen
- Recreatief binnenwater
- Semi verhard overig terrein
- Spoorterrein
- Sportterrein
- Stortplaats
- Terrein voor detailhandel, winkelcentra en horeca
- Terrein voor glastuinbouw
- Terrein voor openbare voorzieningen
- Terrein voor sociaal-culturele voorzieningen
- Verblijfsrecreatief terrein
- Vliegveld
- Volkstuin
- Wegverkeersterrein
- Woonterrein
- Wrakkenopslagplaats
- <all other values>

Kadaster, Esri, HERE, Garmin, GeoTechnologies, Inc., USGS, METI/NASA
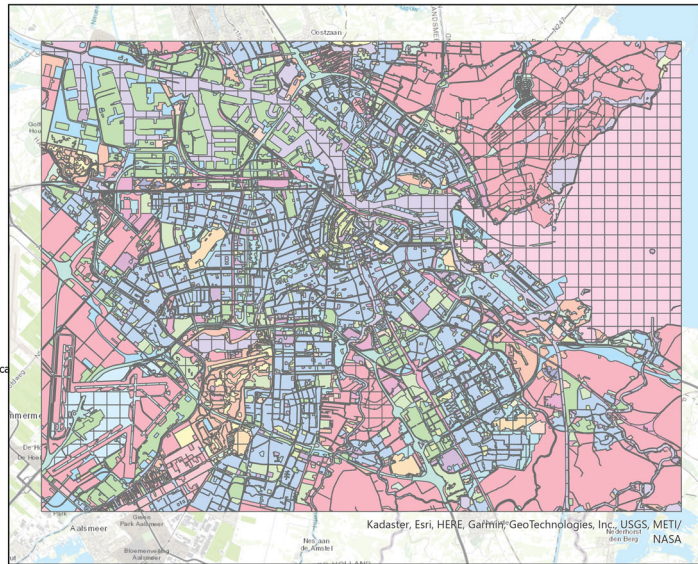
(b)

Legend

PC4

Average distance to parks
- 50-165
- 165-300
- 300-450
- 450-650
- 650-900
- 900-1600
- 1600-2700

Esri, HERE, Garmin, USGS, NGA
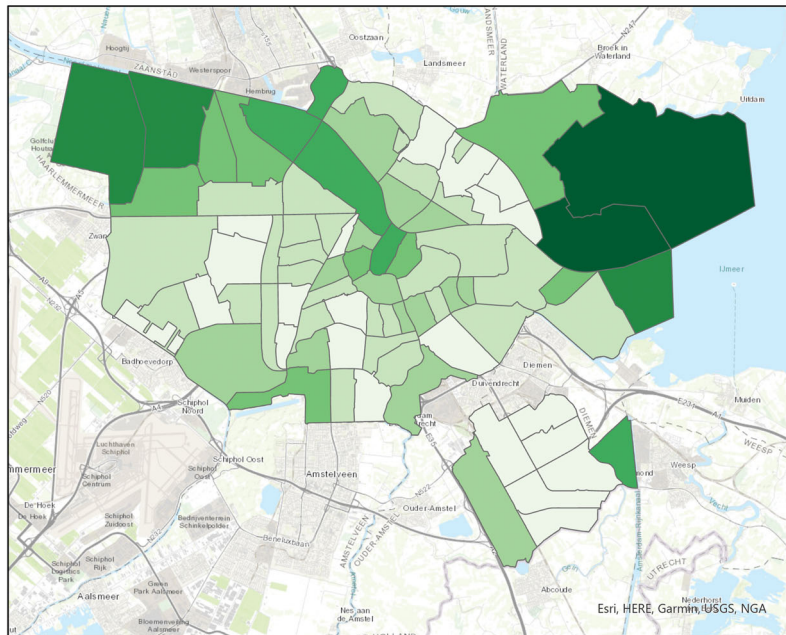
Figure 3. Input data, ArcGIS workflow, and answer map for answering 'What is the average Euclidean distance to parks for each PC4 area in Amsterdam?' (a) Land use map of Amsterdam, provided by the City of Amsterdam. (b) Map of the average distance to parks for each PC4 area in Amsterdam. (c) ArcGIS workflow computing the average Euclidean distance to parks for each PC4 area in Amsterdam.

Figure 3. Continued.

Figure 4. Annotating GIS operations with concepts. (a) Annotate Euclidean distance operation with input object and output field. (b) Annotate Zonal statistic operation with input object, field, and output content amount.

2. What is the density$^{(Proportion\ IRA)}$ of parks$^{(Object)}$ for each PC4 area$^{(Object)}$ in Amsterdam$^{(Object)}$?
3. What is the average Euclidean distance$^{(Field)(Content\ Amount\ ratio)}$ to parks$^{(Object)}$ for each PC4 area$^{(Object)}$ in Amsterdam$^{(Object)}$?
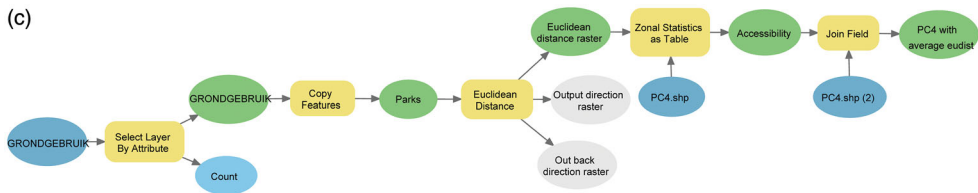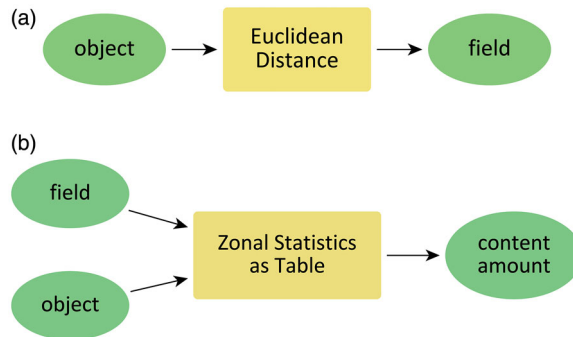
These examples demonstrate that nouns and noun phrases in the questions can be interpreted in terms of core concepts, amounts and proportions, and their measurement levels.[11] The same can also be done with GIS workflows: Every GIS operation requires input and produces output. These inputs and outputs can be annotated in the same manner. For example, the Euclidean Distance tool requires an object input and generates a distance field. Similarly, the Zonal Statistics tool transforms fields into field content amounts (Scheider *et al.* 2020a). Figure 4 depicts corresponding examples. Given such annotations of both GIS operations and question components, it becomes possible to map question components to the input and output of workflows. For example, the components of question (3) can be mapped as shown in Figure 5.

However, note that a question *abstracts* from the exact operations or the formats used in the workflow. For example, the use of Zonal Statistics is a technicality of implementing rather than interpreting the question. More important is to understand that a *Field* should be transformed into some *Content Amount* via adequate operations (see Figure 6). For this reason, we concentrate in the article on the challenge of inferring transformations on a conceptual level. Tool or data technicalities have been recently addressed in separate workflow synthesis studies (Scheider *et al.* 2020a).
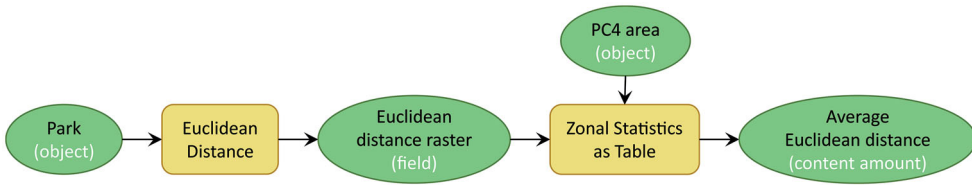
Figure 5. Mapping question components to input and output of GIS operations.
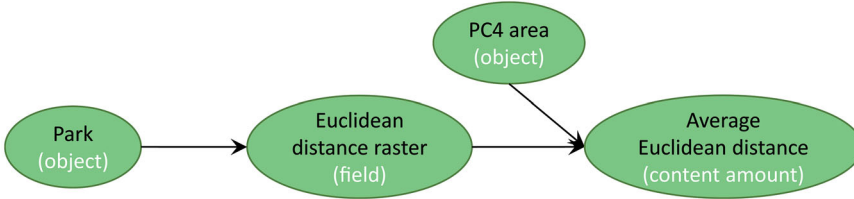

Figure 6. An abstract workflow formed by concepts.

2.2.4. Concept transformation model

The possible concept combinations that can occur in a transformation can be captured in a *context-free grammar* as shown below:

$$
\begin{aligned}
\langle \text{concept} \rangle \ &\models\ \langle \text{coreConcept} \rangle \mid \langle \text{amount} \rangle \mid \langle \text{proportion} \rangle \mid && (1) \\
& \quad\ \langle \text{discreteQuality} \rangle \mid \text{`Location'} && (2) \\
\langle \text{coreConcept} \rangle \ &\models\ \langle \text{discreteCoreConcept} \rangle \mid && (3) \\
& \quad\ \text{`Field'} \ \langle \text{MeasurementLevel} \rangle && (4) \\
\langle \text{discreteCoreConcept} \rangle \ &\models\ \text{`Object'} \mid \text{`Event'} \mid \text{`Network'} && (5) \\
\langle \text{amount} \rangle \ &\models\ \langle \text{contentAmount} \rangle \mid \langle \text{coverageAmount} \rangle && (6) \\
\langle \text{contentAmount} \rangle \ &\models\ (\langle \text{coreConcept} \rangle \mid \langle \text{discreteQuality} \rangle) && (7) \\
& \quad\ \text{`ContentAmount'} \ \langle \text{MeasurementLevel} \rangle && (8) \\
\langle \text{coverageAmount} \rangle \ &\models\ \langle \text{coreConcept} \rangle \ \text{`CoverageAmount'} && (9) \\
& \quad\ \langle \text{MeasurementLevel} \rangle && (10) \\
\langle \text{proportion} \rangle \ &\models\ \langle \text{amount} \rangle \ \langle \text{amount} \rangle \ \text{`Proportion'} && (11) \\
& \quad\ \langle \text{MeasurementLevel} \rangle && (12) \\
\langle \text{discreteQuality} \rangle \ &\models\ \langle \text{discreteCoreConcept} \rangle \ \text{`Quality'} && (13) \\
& \quad\ \langle \text{MeasurementLevel} \rangle && (14) \\
\langle \text{MeasurementLevel} \rangle \ &\models\ \text{`Boolean'} \mid \text{`Nominal'} \mid \text{`Ordinal'} \mid && (15) \\
& \quad\ \text{`Interval'} \mid \text{`Ratio'} \mid \text{`ERA'} \mid \text{`IRA'} \mid && (16) \\
& \quad\ \text{`Count'} \mid \text{`loc'} && (17)
\end{aligned}
$$

A context-free grammar consists of non-terminal production rules and terminal symbols (here in Extended Backus-Naur form), where each rule can be applied independently from the context of non-terminals (Hopcroft *et al.* 2006, Aho *et al.* 2007). For example, ⟨*concept*⟩ and ⟨*coreConcept*⟩ denote non-terminal production rules, while words in single quotes like 'Location', 'Object' are terminal symbols. Based on these rules, ⟨*MeasurementLevel*⟩ can always be replaced by 'Nominal' or other symbols in this rule, and ⟨*coreConcept*⟩ can always be replaced by 'Field Nominal' or other symbols substitutable in the head of the corresponding rule. From this grammar, we can
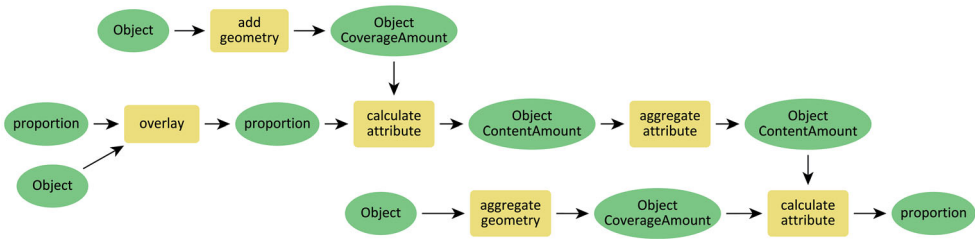
Figure 7. A DAG of concept transformations in Figure 1(c). Each vertex represents a concept of the data, and each directed edge refers to an abstract GIS operation (concrete tools can be found in Table 1). Concept transformation models abstract information transformations in questions and GIS workflows, and thus are different from GIS workflows.

learn, e.g. that a discrete quality can be an object quality, event quality, or a network quality, but not a location quality.

Concept transformations are then defined as directed acyclic graphs (DAG) of these concepts. As shown in Figure 7, transformation is directed from one or more concepts to another without forming any closed cycles. We will illustrate the variety of possible transformation graphs occurring in a geo-analytical question corpus in Section 4.1.

2.3. Using functional roles to identify concept transformation ordering

A final challenge on the way toward a question parser is to identify the order of transformation specified in question components. For example, how to identify that average Euclidean distance (Content Amount) is the desired goal concept? To address this problem, we introduce a notion of functional roles to clarify the roles that spatial information can take within the analytic process specified by the questions. In total, we propose six functional roles, in rough correspondence with Sinton's spatial measurement framework (Sinton 1978):

- *Measure* provides the question's goal and the inputs for GIS workflows
- *Condition* narrows down the input of *measure* or *support* based on spatial relationships and thematic attributes.
- *Sub-condition* narrows down input of *condition* based on spatial relationships and thematic attributes.
- *Support* provides the spatial control of the *measure* (called *control* by Sinton 1978). An example of the support role would be the statistical zones used to aggregate information as in zonal map algebra.
- *Extent* defines the spatial boundary for *support*, *measure*, *condition*, and *sub-condition*.
- *Temporal extent* defines the temporal boundary for the *support*, *measure*, *condition*, and *sub-condition*.

We annotated the geo-analytical questions below with the functional roles as an example. Two more questions are added here to explain the sub-condition and condition for support.

Table 1. Mapping abstract GIS operations in Figure 7 to GIS tools.

Abstract operations	ArcGIS tool	QGIS tool
Overlay	Identity	Intersection
Add geometry	Add geometry attribute	Add geometry attributes
Calculate attribute	Calculate field	Field calculator
Aggregate attribute	Dissolve	Aggregate
Aggregate geometry	Dissolve	Aggregate

1. What is the density of people[Measure] older than 65[Condition] for each PC4 area[Support] in Amsterdam[Extent]?
2. What is the density of parks[Measure] for each PC4 area[Support] in Amsterdam[Extent]?
3. What is the average Euclidean distance to parks[Measure] for each PC4 area[Support] in Amsterdam[Extent]?
4. Which houses[Measure] are in the neighborhoods[Condition] with the lowest crime rate[Subcondition] in Amsterdam[Extent] in 2020[TemporalExtent]?
5. What is the population density[Measure] for each neighborhoods[Support] with no health care facility[Condition] in Riverside-San Bernardino[Extent]?

In the first question, the *measure* points out that in the GIS workflow the input and output are both a density of people. The population density is constrained to the people over 65 by the *condition*. The *support* indicates that the output density is aggregated into PC4 areas. Similarly, in question (4), the *measure* is constrained by a topological relation to neighborhoods in the *condition*. However, the neighborhoods should first be selected based on the contents of the *sub-condition*. Besides that, *condition* can also constrain the *support*, as shown in question (5). The *support* 'neighborhoods' is selected by the *condition* before using it in generating the final population density. What this shows is that the functional roles reveal a *generation order* for transforming information in workflows. Based on these considerations, it becomes possible to specify a set of *ordering rules* for translating functional roles into transformations.

3. Methodology

In this section, we explain how to translate geo-analytical questions into concept transformations. We first introduce a training question corpus in Section 3.1. Then a method for extracting transformations from questions is introduced which was developed based on the corpus.

3.1. Geo-analytical question corpus

GeoAnQu is a geo-analytical question corpus proposed by Xu *et al.* (2020). It contains 429 questions formulated from geo-analytical use cases in GIS textbooks (O'Looney 2000, Heywood *et al.* 2011, Allen 2013, Kraak and Ormeling 2013), and scientific articles collected by Wielemann (2019). However, in this corpus, questions and answer workflows are very diverse in terms of goals, methods, and analysis platforms. To obtain a typical set of GIS questions, we made the following changes:

1. We only kept those questions that can be answered by ArcGIS and QGIS work-flows. Questions that require, e.g. temporal analysis or programming are out of scope.
2. Abstract questions often require the human effort of interpretation to make them concrete in terms of GIS transformations. For example, to answer a question like '*Do freeways have directional influence on fire responses of fire stations in Fort Worth?*', we need to compare the directional trend maps of fire calls with a fire station map. We, therefore, reformulated such questions in concrete terms, adding any missing (implicit) information. In this case: '*What is the directional trend of fire calls for each fire station in Fort Worth?*'.
3. Similarly, some questions are too vague to construct workflows. For example, '*What areas are far away from the road in Spain?*'. In such questions, we replaced vague phrases like 'far away' with a specific parameter '1 km'.

The purpose of this reformulation is to limit the scope of our study to the most typical and most concrete kinds of questions, leaving other options for future research. The resulting corpus contains 309 questions of which 196 are generated from the GIS textbooks mentioned above, 76 are from the scientific articles, and others are from GIS courses. These questions are used as a training data set in the study.

3.2. Extracting concept transformations from geo-analytical questions

Our method was developed in two steps, where the knowledge acquired manually in the first step is being automated in the second step. As shown in Figure 8, each step involved development cycles that need to be repeated until a sufficient quality level was reached. The proposed method was implemented using Python.

3.2.1. Manual question categorization

To understand the functional roles that concepts can play in a question, we first con-ducted an exploratory study on the questions in GeoAnQu and their GIS workflows.

We manually annotated the functional roles of concepts in each GeoAnQu question as explained in Sections 2.2 and 2.3. The keywords associated with each concept were stored in a separate dictionary. In this way, we discovered functional roles of concepts, which were then used to generate concept transformations for each question. To this end, we interpreted both the annotations and the given workflows in terms of trans-formations. This process was repeated until all questions could be interpreted into such transformations in a straightforward manner. As a result, each question obtained a manually produced transformation.

Next, we categorized each GeoAnQu question by the types of transformations, and then explored the dominant syntactic structures underlying functional roles in each category. The results of this question categorization are discussed in Section 4.1.

3.2.2. Functional grammar development

To automate the extraction of functional roles for arbitrary geo-analytical questions, we formalized recurrent syntactic structures in terms of context-free grammar.
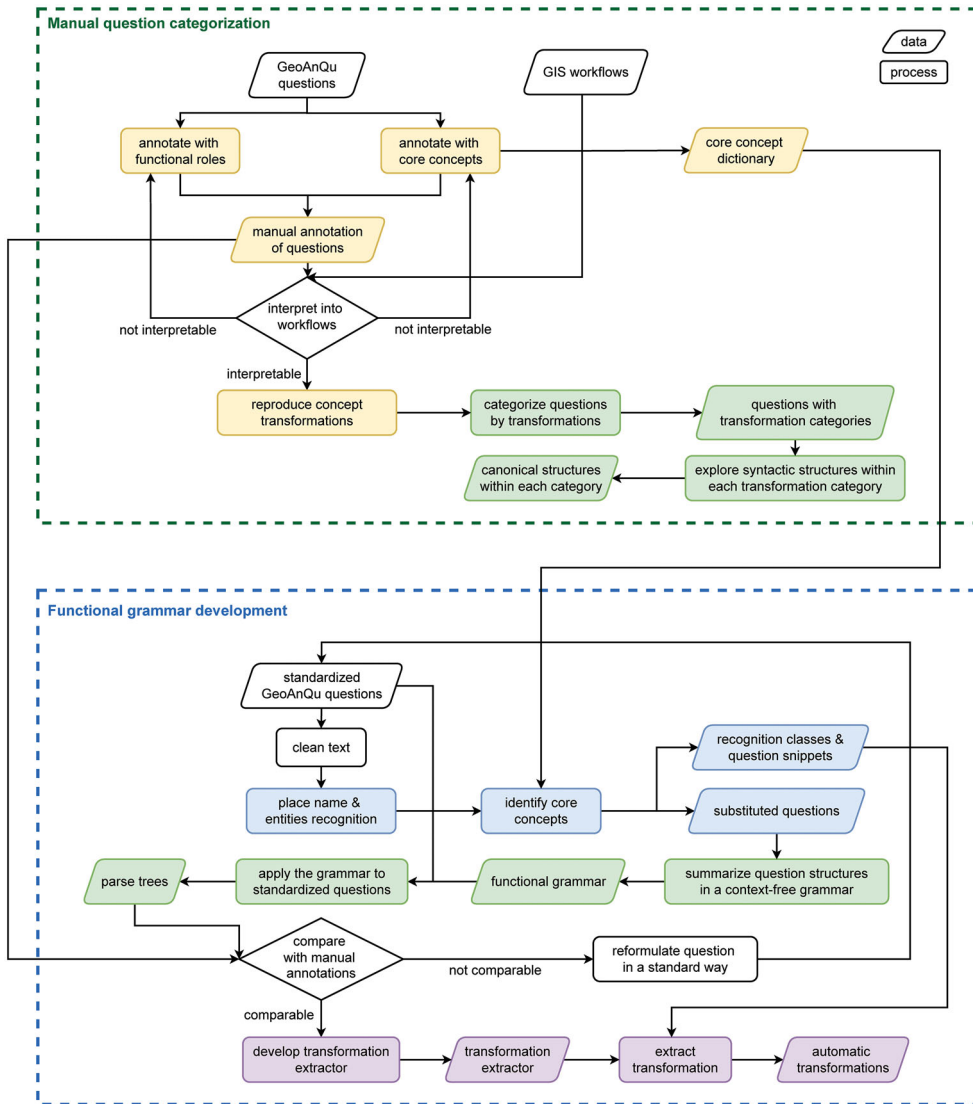
Figure 8. Methodology of developing a grammar for interpreting geo-analytical questions as concept transformations.

The main problem is that syntactic structures of the same transformation can widely differ in terms of keywords and syntactic form. Based on our exploratory study, we therefore first identified geo-analytical questions which specify the same concept transformation and reformulated them into a *canonical form*. The questions in our training corpus could always be reformulated such that it preserves the information about the transformation. For example, coverage amount questions, such as '*Where are fire calls highly clustered in Fort Worth?*' were reformulated into '*Where are the clusters of fire calls in Fort Worth?*', and '*How clustered is the distribution of the fire calls in Fort Worth in 2015?*' was reformulated into '*What is the degree of clustering of fire calls in Fort Worth in 2015?*'. Questions with '*network*' as a goal concept can always be
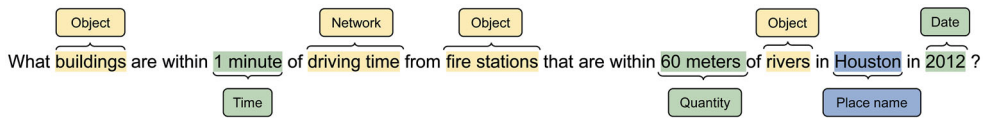
Figure 9. Named entity recognition in a geo-analytical question. The place name is captured by the Elmo-based NER model from AllenNLP. Entities, such as time, quantity, and date are recognized by the SpaCy Python package. Concepts, such as object, network are recognized by a pre-defined concept dictionary.

Table 2. Recognition results of the question 'What buildings are within 1 minute of driving time from fire stations that are within 60 meters of rivers in Houston in 2010?'.

Recognition classes	Question snippets
Object	['buildings', 'fire stations', 'rivers']
Network	['driving time']
Etime	['1 minute']
Equantity	['60 meters']
Placename	['Houston']
Edate	['2010']

reformulated with the pattern *'network from object to object'*, like in *'the shortest route from home to hospital'*.

Another difficulty was the handling of terminals that denote countless place names and other possible referents in a question, such as 'Amsterdam' and 'parks'. To handle this problem, *Named Entity Recognition* (NER) models (Mohit 2014) were used to recognize place names and entities in the GeoAnQu corpus. As shown in Figure 9, we first recognized place names using ELMO-based Named Entity Recognition[12] from AllenNLP, then identified *Time*, *Quantity*, and *Date* via the NER model from the spaCy python package.[13] Lastly, we used the concept dictionary developed in the exploratory study to capture geographic phenomena as concepts in a question. After this step, the corresponding text snippets were substituted by NER classes and concepts, such as 'What *object0* are within *etime0* of *network0* from *object1* that are within *equantity0* of *object2* in *placename0* in *edate0*'. The numeric index in the class name distinguishes between different entities of the same class. The classes and the question snippets were stored separately (see Table 2). The functional grammar was then formulated on top of the NER classes and concepts, where the classes appear as terminals in the grammar.

Since the complete functional grammar is extensive, we show here only a simplified version. The complete version is accessible via the link in Data and codes availability statement. The ⟨*start*⟩ parser rule represents syntactic structures of whole questions which consist of question words ⟨*W*⟩, auxiliary words ⟨*AUX*⟩, functional roles [measure, (sub)condition, support, extent], and their connection terminals (e.g. 'for each'). It can be seen that all the questions in Section 2.3 are variants of the structure specified in the ⟨*start*⟩ rule. Functional roles, in turn, are then defined as sequences of core concepts ⟨*coreC*⟩, intermediary concepts (e.g. ⟨*distField*⟩), and quantities (e.g. 'equantity') captured before. To specify ⟨*condition*⟩ and subcondition ⟨*subcon*⟩, we used four relation types as occurring in GeoAnQu. Namely, topological relations [e.g. *within* forest

areas (coreC)], comparison relations [e.g. *lower than* 40 millimeters (equantity)], extrema relations [e.g. *highest* crime rate (coreC)], and Boolean relations [e.g. *responded by* fire stations (coreC)]. We also defined intermediary concepts that are subclasses of core concepts. ⟨*distField*⟩ describes a distance field usually generated from distance to objects or events, such as '*50 meters of a chemical explosion*'. ⟨*serviceObj*⟩ refers to service areas[14] in Network analysis, such as '*100-meter network distance from primary schools*' and '*3 minutes of driving time from hospitals*'. As the direction is important in Network analysis, we also captured objects of origin in the grammar. Intermediary concepts identify types of transformations that are more specific than the core concept level.

$$
\begin{array}{rcll}
\langle\text{start}\rangle & \models & \langle\text{W}\rangle\ \langle\text{AUX}\rangle\ \langle\text{measure}\rangle\ \langle\text{condition}\rangle* & (18)\\
 & & ((\text{`with'}\ |\ \text{`that'}\ \langle\text{AUX}\rangle)\langle\text{subcondition}\rangle)? & (19)\\
 & & (\text{`for each'}\ \langle\text{support}\rangle)?\ \langle\text{condition}\rangle? & (20)\\
 & & (\text{`in'}\ \langle\text{extent}\rangle)+ & (21)\\
 & & ((\text{`in'}\ |\ \text{`on'})\ \langle\text{temporalEx}\rangle)? & (22)\\
\langle\text{measure}\rangle & \models & \langle\text{coreC}\rangle\ ((\text{`of'}\ |\ \text{`from'}\ |\ \text{`to'})\ \langle\text{coreC}\rangle)* & (23)\\
\langle\text{condition}\rangle & \models & ((\langle\text{topoR}\rangle\ |\ \langle\text{extreR}\rangle\ |\ \langle\text{boolR}\rangle) & (24)\\
 & & (\langle\text{coreC}\rangle\ |\ \langle\text{distField}\rangle\ |\ \langle\text{serviceObj}\rangle))\ | & (25)\\
 & & (\langle\text{coreC}\rangle?\ \langle\text{comparisonR}\rangle\ \text{`equantity'}\langle\text{NUM}\rangle) & (26)\\
\langle\text{subcondition}\rangle & \models & ((\langle\text{topoR}\rangle\ |\ \langle\text{extreR}\rangle\ |\ \langle\text{boolR}\rangle) & (27)\\
 & & (\langle\text{coreC}\rangle\ |\ \langle\text{distField}\rangle\ |\ \langle\text{serviceObj}\rangle))\ | & (28)\\
 & & (\langle\text{coreC}\rangle?\ \langle\text{comparisonR}\rangle\ \text{`equantity'}\langle\text{NUM}\rangle) & (29)\\
\langle\text{support}\rangle & \models & \langle\text{coreC}\rangle & (30)\\
\langle\text{extent}\rangle & \models & \text{`placename'}\ \langle\text{NUM}\rangle & (31)\\
\langle\text{temporalEx}\rangle & \models & \text{`date'}\ \langle\text{NUM}\rangle & (32)\\
\langle\text{coreC}\rangle & \models & (\text{`field'}\ \langle\text{NUM}\rangle\ \langle\text{ML}\rangle)\ |\ (\text{`object'}\ \langle\text{NUM}\rangle)\ | & (33)\\
 & & (\text{`proportion'}\ \langle\text{NUM}\rangle\ \langle\text{ML}\rangle)\ |\ \dots & (34)\\
\langle\text{distField}\rangle & \models & \text{`equantity'}\langle\text{NUM}\rangle\ \text{`of'}\ \langle\text{coreC}\rangle & (35)\\
\langle\text{serviceObj}\rangle & \models & (\text{`etime'}\ |\ \text{`equantity'})\langle\text{NUM}\rangle\ \text{`of'}?\ \langle\text{networkC}\rangle? & (36)\\
 & & (\text{`from'}\ \langle\text{origin}\rangle) & (37)\\
\langle\text{networkC}\rangle & \models & \text{`network'}\ \langle\text{NUM}\rangle & (38)\\
\langle\text{origin}\rangle & \models & \langle\text{objectC}\rangle\ |\ \langle\text{eventC}\rangle & (39)\\
\langle\text{objectC}\rangle & \models & \text{`object'}\ \langle\text{NUM}\rangle & (40)\\
\langle\text{eventC}\rangle & \models & \text{`event'}\ \langle\text{NUM}\rangle & (41)\\
\langle\text{topoR}\rangle & \models & \text{`within'}\ |\ \text{`overlap'}\ |\ \dots & (42)\\
\langle\text{extreR}\rangle & \models & \text{`maximum'}\ |\ \text{`highest'}\ |\ \dots & (43)\\
\langle\text{boolR}\rangle & \models & \text{`have'}\ |\ \text{`responded by'}\ |\ \dots & (44)\\
\langle\text{comparisonR}\rangle & \models & \text{`larger than'}\ |\ \text{`less than'}\ |\ \dots & (45)\\
\langle\text{W}\rangle & \models & \text{`what'}\ |\ \text{`which'}\ |\ \dots & (46)\\
\langle\text{AUX}\rangle & \models & \text{`are'}\ |\ \text{`was'}\ |\ \dots & (47)\\
\langle\text{ML}\rangle & \models & \text{`nominal'}\ |\ \text{`ratio'}\ |\ \dots & (48)\\
\langle\text{NUM}\rangle & \models & \text{[0-9]}+ & (49)\\
\end{array}
$$

The functional grammar was converted into a parser in ANTLR.[15] We then parsed (entity-substituted) GeoAnQu questions with this parser, automatically matching the question string by grammar rules and generating parse trees for each question. One example is shown in Figure 10.
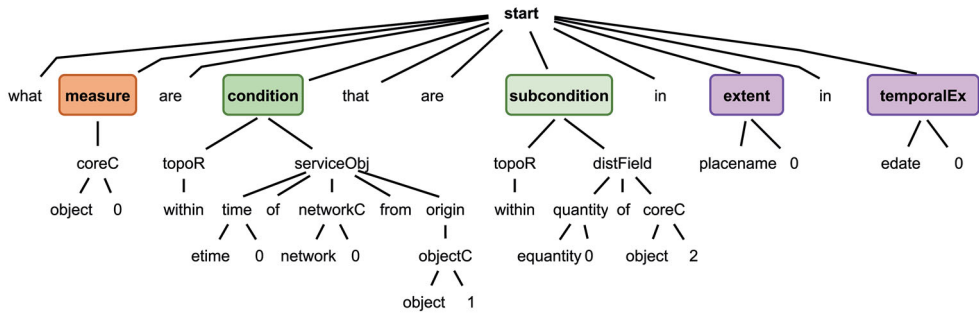
Figure 10. The parse tree of the question 'What buildings are within 1 minute of driving time from fire stations that are within 60 meters of rivers in Houston in 2010?'.

Table 3. Extraction results of the question '*What buildings are within 1 minute of driving time from fire stations that are within 60 meters of rivers in Houston in 2010?*'.

Functional roles	Concepts	Question snippets
Measure	['object0']	['buildings']
MeasureCondition	['serviceObj', 'networkC', 'network', 'object1']	['', 'driving time', 'road network', 'fire stations']
Subcondition	['distField', 'object2']	['', 'rivers']
Extent		['Houston']
TemporalExtent		['2010']

The parse trees were compared with the manual transformation annotations. In case they did not correspond to each other, we either reformulated the question into a standard structure or adapted steps in the NER and parsing process. We did this until the parse trees were corresponding with the manual annotations.
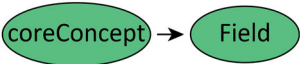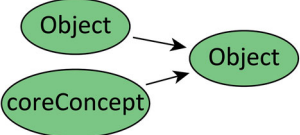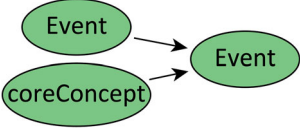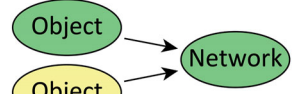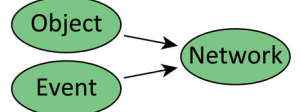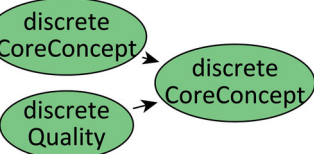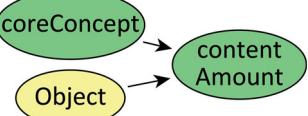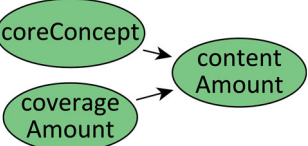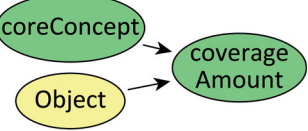
3.2.3. Transformation extractor

We then developed a program that turns parse trees into expressions of concept transformations. The transformation extractor consists of two blocks. The first block defines *input and output concepts* for each type of transformation. These transformation patterns were obtained by manual question categorization and are discussed later in Table 4. The patterns help us complete input concepts that are required to generate some concepts but are missing in questions. For example, to generate a service object, in addition to 'driving time' ('networkC') which was mentioned in the question, a road network is needed but was never mentioned (see Table 3). The second block forms transformation expressions based on ordering concepts using functional roles as they appear in the canonical syntax.

Algorithm 1 shows the approach for generating the concept transformation order in the second block. We first separately stored question snippets (we call these 'keywords' in the following) together with corresponding concepts for each functional role, as shown in Table 3. In the Table, 'object0' is captured as a measure and is stored with the keyword 'buildings' in the questions. Keywords can also be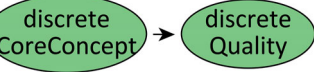 missing. For example, there are usually no keywords to be found for intermediary concepts (serviceObj, distField) in questions.

Then, the order of concepts was determined based on the *order of occurrence* of functional roles and concepts within a functional role. More precisely, if multiple

Table 4. Question categories in GeoAnQu with concept transformation models.

Category	Concept transformation model	Question example
Field-based category	coreConcept → Field	'What is the *buffer zone*(Field) within 5 kilometers of *parks*(Object) in Amsterdam?'
Object-based category	Object, coreConcept → Object	'Which *fire station*(Object) is closest to each *fire call*(Event) in Fort Worth?'
Event-based category	Event, coreConcept → Event	'Which *migratory tracks of birds*(Event) cross *national parks*(Object) in the US in 2010?'
Network-based category	Object, Object → Network	'What is the *shortest route*(Network) from *centers of municipalities*(Object) to *outpatient services*(Object) in Tuscany in Italy?'
	Object, Event → Network	'What is the *shortest route*(Network) from *hospitals*(Object) to *a traffic accident*(Event) in New York?'
Discrete core concept based category	discrete CoreConcept, discrete Quality → discrete CoreConcept	'Which *shops*(Object) are *open at 9 am*(Object Quality) in Happy Valley ski resort?'
Discrete quality based category	discrete CoreConcept → discrete Quality	'What is the *walkability*(Object Quality) for each *neighborhood*(Object) in Ghent?'
Content amount based category	coreConcept, Object → content Amount	'What is the *number*(Content Amount) of *traffic accidents*(Event) in Amsterdam?'
	coreConcept, coverage Amount → content Amount	'What is the *elderly population*(Content Amount) in the *Oost district*(Object) in Amsterdam?'[a]
Coverage amount based category	coreConcept, Object → coverage Amount	'What is the *total area*(Coverage Amount) of *agriculture land*(Field) in Netherlands?'

Table 4. Continued.

Category	Concept transformation model	Question example
		'What is the *mean center*[(Coverage Amount)] of *accidents*[(Event)] for each *alarm territory*[(Object)] in Fort Worth?'
Proportion based category		'What is the *proportion*[(Proportion)] of *Hispanic population*[(Content Amount)] to the *total population*[(Content Amount)] in Tarrant County in Texas?'
		'What is the *density*[(Proportion)] of *green space*[(Field)] in *Amsterdam*[(Object)]?'[b]
		'What is *tree*[(Object)] *density*[(Proportion)] for each *PC4 area*[(Object)] in Amsterdam?'[c]

Yellow nodes represent optional input concepts.
[a]The elderly population can be calculated by taking the elder population density to multiply the area of the Oost district. Thus, the content amount can also be generated from proportion and coverage amount.
[b]Density equals green space area divided by Amsterdam area. Therefore, the field and object were transformed into coverage amount and then transformed into proportion.
[c]Tree density equals the number of tree divided by the acreage of PC4 areas. Therefore, the proportion here is transformed from the content amount of tree and the coverage amount of PC4 areas.
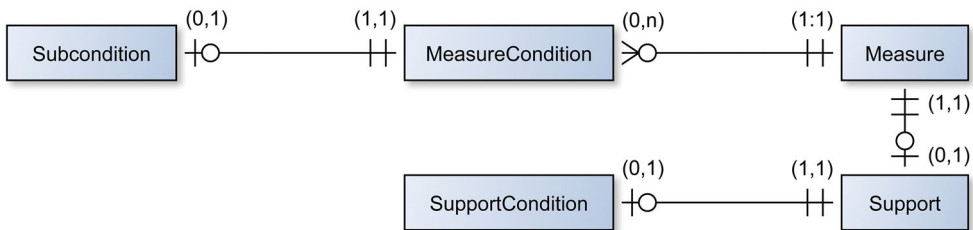
Figure 11. ER diagram of the functional roles, representing the constrained relationship among functional roles. For example, each measure can be constrained by 'zero, one, or many' measureConditions, while each measureCondition can constrain 'only one' measure.

concepts occur in a functional role, we added a transformation where the first concept occurring in the role's phrase is output and the rest is input. The transformation order across functional roles (as encoded in the canonical syntax) is specified in Figure 11. Such transformations are generated by linking the output concept of the previous role with the first input concept of the successor role. If a role has only one concept, it is used both as input and output. This leads to transformation steps as depicted in Figure 12, where we start from subcondition, over measurecondition, toward support

and measure. Extent and temporal extent are used for data collection, not for data processing, thus do not occur in the ordering.

Algorithm 1: generating concept transformations in transformation extractor

---

Input : The parse tree P of question Q, Recognition results N
Output: Concept types $types$, keywords k, transformation $trans$

1 **begin**
    /* Extract functional roles and concepts from parse tree, and
       question snippets from recognition results   */
2   **for** $non\text{-}terminal\ p\ in\ P$ **do**
3   **if** $p\ in\ [measure,\ condition,\ subcondition,\ support,\ extent,$
        $temporalExtent]$ **and** $p\ captures\ some\ recognition\ classes\ in\ N$ **then**
4   Save p as functional role in $frole$
5   Extract concepts in P under p and store them in $types$
6   Retrieve keywords k from N for each concept

    /* Distinguish conditions for support and measure   */
7   **for** $functional\ role\ f\ in\ frole$ **do**
8   **if** $f\ equal\ to\ condition\ and\ f\ occurs\ behind\ support\ in\ P$ **then**
9         $f \leftarrow supportCondition$
10   **else**
11         $f \leftarrow measureCondition$

   /* Generate transformations for functional roles except for
       measure   */
12   **for** $f\ in\ frole$ **do**
13   **if** $f\ in\ [supportCondition,\ support,\ subcondition,\ measureCondition]$
      then
14   **if** $f\ consists\ of\ more\ than\ one\ concept$ **then**
15   generate $trans$ for input/output $types$ in f
16   generate $trans$ across functional roles based on the transformation
        order

    /* Generate transformations for measure (using outputs from
       other roles)   */
17   **for** $f\ in\ frole$ **do**
18   **if** $f\ equal\ to\ measure$ **then**
19   **if** $measureCondition\ in\ frole$ **and** $support\ not\ in\ frole$ **then**
20   Generate $trans$ from the output concept of measureCondition to
          the first input concept of measure
21   **if** $measure\ consists\ of\ more\ than\ one\ concept$ **then**
22   Generate $trans$ from the input concept of measure to its
            output concept

23   **else if** $support\ in\ frole$ **and** $measureCondition\ not\ in\ frole$ **then**
24   Generate $trans$ from the output of support and the input of
          measure to the output of measure
25   **else if** $measureCondition\ in\ frole$ **and** $support\ in\ frole$ **then**
26   Generate $trans$ from the output concept of measureCondition to
          the first input concept of measure
27   Generate $trans$ from the output of support and the input of
          measure to the output of measure

---

4. Transformation results

In this section, we present the results of the manual question categorization and the functional grammar development in our methodology. Section 4.1 illustrate question
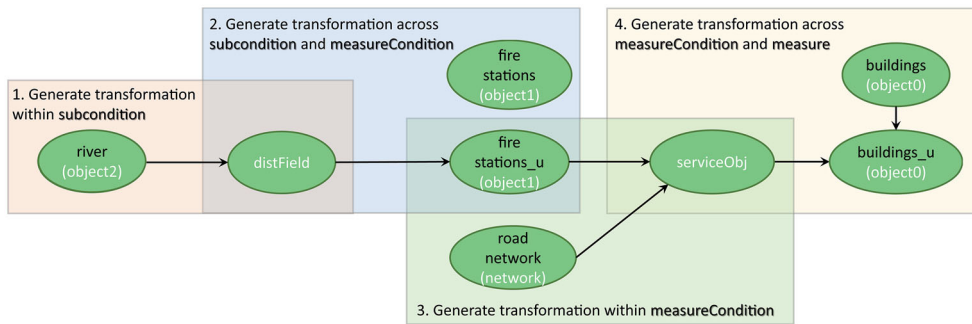
Figure 12. Transformations of the question '*What buildings are within 1 minute of driving time from fire stations that are within 60 meters of rivers in Houston in 2010?*'.

categories found in GeoAnQu with the concept transformation model. Section 4.2 shows an example of automatically generated concept transformations.

## 4.1. Question categories with concept transformation models

Table 4 gives an overview of question categories defined by the output concept in a transformation model of a question. All question examples are given here follow the canonical syntax. Models (as well as questions) can be concatenated with each other based on a common concept node. For example, a proportion model can be combined with amount based model when taking a field or object as input. In this way, one may even find subquestions for a given superquestion.

In Figure 13, we have quantified the frequency of question models occurring in the corpus. CoverageAmount questions are most frequent. They include also 'where' questions that ask for the coverage of a place. Field, Object, contentAmount, and proportion questions are also abundant, whereas network and event questions occur seldomly. Furthermore, the total number of questions in the figure is larger than GeoAnQu because some questions contain different conditions and thus are counted in different models. For example, the question '*What houses are for sale and within 0.5 km from the main roads in Utrecht?*' is counted both in the object and discreteCoreConcept model.

4.2. Automatic concept transformations

Concept transformations were serialized in a JSON file as shown in List. 1. It mainly consists of three parts: recognition results (Line 4 to 7); a string of parse tree (Line 8), and concept transformations (Line 9 to 26). Under 'cctrans', 'types' encompasses all the concept types and question keywords involved in 'transformations'. In List. 1, 'types' contains two elements, in which the first one is about object 'recreational sites' and the second is about field 'Euclidean distance'. The 'extent' and 'temporal extent' (if it exists) keys store two separate arrays with keywords as elements. Last, each pair of 'before' and 'after' in 'transformations' represents the input and output of a transformation step. The elements of 'before' and 'after' are identified by the ids given in 'types'. The question example in List. 1 shows one transformation step from the object (keyword 'recreational sites') to the field ('Euclidean distance').

Frequency of GeoAnQu questions for concept transformation models
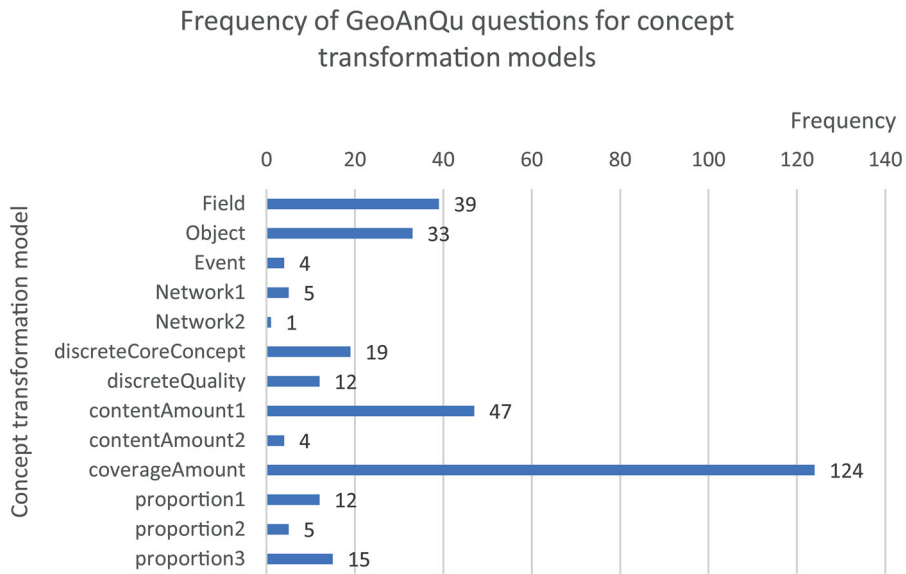
Figure 13. Frequency of questions for concept transformation models in GeoAnQu. Models are named according to output concepts in the leftmost column. The numbers (e.g. Network1, Network2) refer to different input concepts for a given output.

Listing 1 Results of the functional grammar development in JSON syntax. The extracted concepts and transformations are listed under 'cctrans'.

```
1  {
2       "question": "What is the Euclidean distance to recreational sites in Utrecht",
3       "cleaned_Question": "What is Euclidean distance to recreational sites in Utrecht",
4       "placename": ["Utrecht"],
5       "field": ["euclidean distance"],
6       "object": ["recreational sites"],
7       "ner_Question": "what is field0 ratio to object0 in placename0",
8       "parseTreeStr": "(start what is (measure (coreC field 0 ratio) to (coreC object 0)) in
               (extent placename 0))",
9       "cctrans": {
10          "types": [
11              {
12                  "type": ["object"],
13                  "id": "0",
14                  "keyword": "recreational sites"
15              },
16              {
17                  "type": ["field"],
18                  "id": "1",
19                  "keyword": "euclidean distance",
20                  "measureLevel": "ratio"
21              }],
22          "extent": ["Utrecht"],
23          "transformations": [
24              {
25                  "before": ["0"],
26                  "after": ["1"]
27              }]
28       }
29  }
```

5. Evaluation

Our evaluation is based on testing to what extent the automatically extracted concept transformations matched a manually generated gold standard. As a first step, we generated an independent test corpus consisting of 134 geo-analytical questions that were taken from *GIS tutorials* available at ArcGISs[16] and QGIS online repositories.[17] All the test questions were first reformulated into the canonical syntax. Then, to obtain automatic transformations (see List 1), we applied the NER models, the functional parser, and the transformation extractor that are introduced above to the test questions. Based on the information in the test questions, we manually generated concept transformations as a gold standard (List 2). The automatic and manual transformations share the same JSON structure, allowing us to compute matches based on *extent*, *temporal extent*, *types*, and *transformations*, respectively.

Listing 2 Manual concept transformations in JSON syntax as a gold standard

```
 1  {
 2        "question": "What is the Euclidean distance to recreational sites in Utrecht",
 3        "cctrans": {
 4              "types": [
 5                    {
 6                          "type": ["object"],
 7                          "id": "0",
 8                          "keyword": "recreational sites"
 9                    },
10                    {
11                          "type": ["field"],
12                          "id": "1",
13                          "keyword": "euclidean distance",
14                          "measureLevel": "ratio"
15                    }],
16              "extent": ["Utrecht"],
17              "transformations": [
18                    {
19                          "before": ["0"],
20                          "after": ["1"]
21                    }]
22        }
23  }
```

We used information retrieval metrics based on recall, precision, and *f*1-score (Zuva and Zuva 2012, Ceri *et al.* 2013). True positives in Table 5 were computed as follows: to compare automatic (temporal) 'extent' and the manual (temporal) 'extent', we counted the number of matched keywords. For 'types', we counted the number of elements that have both the same concept type and keyword. For 'transformation', if before and after concepts and keywords of one transformation step in the gold standard are identical to the automatic transformation step, this will be counted as a true positive transformation step. Next, false negatives were computed as the true

Table 5. Recall, precision and f1 score for 134 test geo-analytical questions.

Dimension	True positive	False positive	False negative	Recall	Precision	f1
Extent	147	0	18	89%	100%	94%
Temporal extent	14	0	7	67%	100%	80%
Types	356	9	55	87%	98%	92%
Transformations	166	27	55	75%	86%	80%

elements in the gold standard that are missing in the automatic transformations. False positives are the elements captured in automatic transformations that are missing in the gold standard. Finally, we measured the recall, precision, and *f*1 score with the formula below:

$$Recall = \frac{True\ positive}{True\ positive + False\ negative}$$

$$Precision = \frac{True\ positive}{True\ positive + False\ positive}$$

$$f1 = 2 * \frac{Recall * Precision}{Recall + Precision}$$

Table 5 indicates that our approach performs considerably well in interpreting geo-analytical questions on an independent test set, especially with respect to the extent and concept types, reaching a high *f*1 score of over 90%. Concept transformations reached a lower *f*1 score of 80%, the lowest among the four dimensions. This is expectable since transformations additionally include order information. In general, recall scores were lower than precision scores, which can be explained by the differences between the training and test corpus (see below).

To analyze the reasons for incorrectly captured or missing elements, we investigated the error sources for all the test questions. As shown in Table 6, 101 out of 134 test questions (accounting for 75%) were well-interpreted in terms of extent, temporal extent, types, and transformations. Since errors can be propagated to the subsequent steps in a processing chain, we investigated error sources based on their first occurrence.

We found that recognition models failed to recognize place names, entities, and core concepts in 7 questions. For example, in the question '*What are the four fire stations with shortest network-based paths to 1202 Twin Peaks Blvd in San Francisco?*', ELMO-based NER failed to recognize street number '1202' with the street name 'Twin Peaks Blvd' together as a place name. Thus the question after recognition became '*What are the 4 object0 with network0 to 1202 placename0 in placename1?*', which does not match the syntactic structures in the functional grammar. Four of these questions can be fully translated into concept transformations if the recognition error were ignored. The remaining three questions have also errors in the functional grammar and transformation extractor.

The reasons for the functional grammar and transformation extractor errors are that some semantic and syntactic variety was not included in the training corpus and thus was not considered in our method. 19 questions with errors in the functional grammar could not be interpreted in terms of both analysis types and question structures. These questions pertain to temporal analysis, spatial autocorrelation, spatial trend analysis, change analysis, and vehicle routing problem. For example, '*At which range of*

Table 6. The number of correctly and falsely interpreted test questions with error sources.

		Error sources		
Total count	Correct count	Recognition models	Functional grammar	Transformation extractor
134	101	7	19	7

time of a day is the number of car accidents highest between 2010 and 2015 in Florida?', '*What are the areas within a four-minute drive of each fire station at 2 a.m. on Tuesday in Utrecht?*', '*What are the countries where literacy rates are rising or falling in Africa between 1990 and 2015?*' and '*What are the fastest routes for three 15000-pound capacity trucks to meet the delivery demands of 25 grocery stores in Utrecht?*'. Although the time keywords, such as '2 a.m.' can be recognized by NER models, such question structures are missing in the training corpus and thus not considered in our model. This inability to interpret temporal analysis questions is the main reason that temporalEx reaches the lowest recall score.

The transformation extractor errors in questions were due to the syntax. Although the types of these questions can be well-captured, only parts of the transformations can be generated. For example, '*What liquor stores are within 1000 foot of schools, libraries, and parks in El Cajon?*'. All concepts and extents were well-extracted. Moreover, questions that select objects from a distance field of one core concept have been well-handled in our approach, such as '*What liquor stores are within 1000 foot of schools in El Cajon?*'. However, the current transformation extractor cannot yet identify a transformation from multiple core concepts to a distance field. Hence, the transformation extractor failed to identify the transformation from libraries and parks to the buffer area. Another transformation extractor error is related to the ambiguous role of the extent. In the question '*Which airports are within 50 mile of Crook, Deschutes, and Jefferson county*', Crook, Deschutes, and Jefferson county were only captured as extent, but here do also play the role of support, which was not foreseen.

6. Discussion and conclusion

In this paper, we suggested and tested the hypothesis that geo-analytical questions asked by GIS users and researchers contain a certain form of procedural knowledge, which can be extracted and used to generate answer workflows in 'indirect QA' systems. Based on core concepts of spatial information, we suggested to translate geo-analytical questions in natural language into concept transformations. To this end, we developed a functional grammar describing the canonical syntax of geo-analytical questions and then extracted concept transformations based on parsed functional roles and transformation patterns that were found in the training corpus GeoAnQu. Afterward, this approach was tested on an independent corpus with 134 geo-analytical questions. The automatically extracted transformations were compared with a manually generated gold standard. Our approach covered concept transformations for 75% of the test questions, providing a first solid basis for an 'indirect QA' question interpreter. To the best of our knowledge, our study is the first one about interpreting geo-analytical questions in terms of concept transformations. Our model not only provides a pipeline for extracting procedure knowledge from natural language questions but also captures an important part of the GIS expertise required for processing spatial information in geo-analytical questions. It makes the required reasoning process explicit and thus improves the interpretability of geographic QA systems.

We found that questions with a structure beyond the sample in our corpus and beyond the canonical syntax were the main causes for errors in the test. Although our

model was trained on over 300 questions, there are still many geo-analytical question types that are not yet covered. This is a predictable limitation for template-based or rule-based models, which can cover only a limited amount of variation. We opted for this approach because only context-free grammar provides the freedom to customize parse rules using the functional roles that represent procedure semantics. The dependency parser and constituency parser only analyze questions from syntactic perspectives. They predict dependency relations (e.g. *POBJ*) or syntactic tags (e.g. *WHNP*) for tokens and spans in questions, which are difficult to match to the functional roles (Xu *et al.* 2020). Second, a solid understanding and interpretable model of concept transformations in questions is a prerequisite for later scaling up this model over different syntactic and semantic variants. This is why machine learning (ML) methods are not our first choice because the training data of transformations is unavailable upfront.

To expand our model for more variants, one way is to collect new types of training questions and build models of their syntactic structures and transformation patterns. To avoid more manual work, ML techniques are widely used in open-domain QA systems (Sagara and Hagiwara 2014, Bao *et al.* 2016, Lukovnikov *et al.* 2017, Yang *et al.* 2019) could also be applied to scale up our model for larger sets of geo-analytical questions. Although they are targeted at retrieving declarative answers rather than generating procedure knowledge, they may provide support in mapping syntactic structures to certain types of transformations. In addition, ML techniques may also help train a NER model for recognizing more diverse concepts in questions, which can replace our manually-defined concept dictionary. State-of-the-art deep learning-based NLP models, such as bidirectional LSTM (BiLSTM) (Dozat and Manning 2016) and BERT (Devlin *et al.* 2018) could be used to deal with both issues.

Based on our model of concept transformations of geo-analytical questions, future work should address two more central issues that make this model useful in real QA systems: First, transformations need to be converted into a query language (e.g. SPARQL[18]) to be able to retrieve answer workflows (Scheider and Lemmens 2017, Yin *et al.* 2021). These answer workflows can be generated using synthesis models similar to the ones described in Kruiger *et al.* (2021). Note that a transformation query goes beyond workflows as they are implemented in standard GIS. Core concepts represent geographic information at a conceptual level regardless of the requirements of data types and tools. They provide a generalized language that can describe transformations across platforms. We, therefore, expect that transformation queries can not only match to workflows, but also to SQL queries that implement similar conceptual transformations in PostGIS. Second, an interface of question formulation needs to be developed. This interactive interface should allow users to formulate geo-analytical questions using our functional grammar as a controlled natural language (CNL) (Schwitter 2005, Mador-Haim *et al.* 2006), so that input questions can be automatically interpreted by the parser. Using the core concept model, it becomes also possible to provide suggestions for users to make their questions more concrete in terms of data inputs.

Notes

1. https://www.esri.com/en-us/arcgis/products/arcgis-pro/overview.
2. https://www.qgis.org/en/site/.
3. For example, based on using transformer models in natural language processing (NLP) (Devlin *et al.* 2018).
4. https://studiegids.vu.nl/en/2020-2021/courses/AB_1107.
5. https://maps.amsterdam.nl/open_geodata/?LANG=en.
6. https://www.cbs.nl/nl-nl/dossier/nederland-regionaal/wijk-en-buurtstatistieken.
7. https://pro.arcgis.com/en/pro-app/latest/help/analysis/geoprocessing/modelbuilder/what-is-modelbuilder-.htm.
8. https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-analyst/euclidean-distance.htm.
9. http://spatial.ucsb.edu/core-concepts-of-spatial-information/.
10. https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-statistics/central-feature.htm.
11. Measurement levels here do not help translate natural language questions into concept transformations but could help generate GIS workflows in the future. Hence, we annotated questions using core concepts together with measurement levels.
12. https://demo.allennlp.org/named-entity-recognition/named-entity-recognition.
13. https://spacy.io/.
14. https://pro.arcgis.com/en/pro-app/latest/help/analysis/networks/service-area-analysis-layer.htm.
15. https://www.antlr.org/.
16. https://learn.arcgis.com.
17. https://www.qgistutorials.com/en/docs.
18. https://www.w3.org/TR/rdf-sparql-query/.
19. https://cbsinuwbuurt.nl/#sub-buurten2019_perc_personen_65_jaar_en_ouder.
20. https://data.amsterdam.nl/datasets/It8vygqzGAantA/grondgebruik/.

Acknowledgments

Disclosure statement

Data and codes availability statement

The data and codes that support this study are available in 'figshare.com' with DOI https://doi.org/10.6084/m9.figshare.17009003.v7.

Funding

Notes on contributors

Haiqi Xu is a Ph.D. student at the Department of Human Geography and Spatial Planning Urban Geography at Utrecht University. Her research interests include GeoAI, geographic question

answering, and natural language processing. She contributed to the idea, methodology, evaluation, manuscript writing, and revision of this paper.

Enkhbold Nyamsuren is a postdoctoral researcher at the Department of Human Geography and Spatial Planning Urban Geography at Utrecht University. He holds research expertise in computer science and cognitive modeling. He contributed to the methodology, evaluation, and writing of this paper.

Simon Scheider is an assistant professor at the Department of Human Geography and Spatial Planning Urban Geography at Utrecht University. His research interest lies at the interface between conceptual modeling, geographic data analysis, and knowledge extraction. He and his group are currently working on a geo-analytical question answering system. He contributed to the idea, writing, and revision of this paper.

Eric Top is a Ph.D. student at the Department of Human Geography and Spatial Planning Urban Geography at Utrecht University. His research interests include knowledge representation, data semantics, and concept formalization in the context of geography and GIS. In particular, his research concerns developing a theory of quantities in geographic information. He contributed to the writing of this paper.

ORCID

Simon Scheider ⓘ http://orcid.org/0000-0002-2267-4810

References

Aho, A.V., *et al.*, 2007. *Compilers: principles, techniques, & tools*. Boston, MA: Pearson Addison Wesley.

Allen, D., 2013. *GIS tutorial 2: spatial analysis workbook*. Redlands: Esri Press.

Bao, J., *et al.*, 2016. Constraint-based question answering with knowledge graph. *In*: *Proceedings of COLING 2016, the 26th international conference on computational linguistics: technical papers*, 2503–2514.

Buzaaba, H. and Amagasa, T., 2021. Question answering over knowledge base: a scheme for integrating subject and the identified relation to answer simple questions. *SN Computer Science*, 2 (1), 1–13.

Ceri, S., *et al.*, 2013. An introduction to information retrieval. *In: Web information retrieval*. Springer, 3–11.

Chen, W., *et al.*, 2013. A synergistic framework for geographic question answering. In: *2013 IEEE seventh international conference on semantic computing*. New York, NY: IEEE Press, 94–99.

Chen, W., *et al.*, 2020. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. *arXiv preprint arXiv:2004.07347*.

Devlin, J., *et al.*, 2018. Bert: pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Diefenbach, D., *et al.*, 2018. Core techniques of question answering systems over knowledge bases: a survey. *Knowledge and Information Systems*, 55 (3), 529–569.

Dozat, T. and Manning, C.D., 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Fader, A., Zettlemoyer, L., and Etzioni, O., 2014. Open question answering over curated and extracted knowledge bases. *In*: *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*, 1156–1165.

Hamzei, E., *et al.*, 2019., Place questions and human-generated answers: a data analysis approach. In: *Proceedings of the 22nd AGILE conference on geographic information science*. Cham: Springer International Publishing, 3–19.

Hamzei, E., Winter, S., and Tomko, M., 2022. Templates of generic geographic information for answering where-questions. *International Journal of Geographical Information Science*, 36 (1), 188–127.

Heywood, I., Cornelius, S., and Carver, S., 2011. *An introduction to geographical information systems*. Harlow: Pearson Education Limited.

Hopcroft, J.E., Motwani, R., and Ullman, J.D., 2006. Automata theory, languages, and computation. *International Edition*, 24 (2), 171–183.

Huang, Z., *et al.*, 2019. Geosqa: A benchmark for scenario-based question answering in the geography domain at high school level. *arXiv preprint arXiv:1908.07855*.

Kapanipathi, P., *et al.*, 2021. Leveraging abstract meaning representation for knowledge base question answering. In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 3884–3894.

Kraak, M. and Ormeling, F., 2013. *Cartography: visualization of spatial data*. Boca Raton, FL: CRC Press.

Kruiger, J.F., *et al.*, 2021. Loose programming of GIS workflows with geo-analytical concepts. *Transactions in GIS*, 25 (1), 424–449.

Kuhn, W., 2012. Core concepts of spatial information for transdisciplinary research. *International Journal of Geographical Information Science*, 26 (12), 2267–2276.

Kuhn, W. and Ballatore, A., 2015. Designing a language for spatial computing. *In*: *Agile 2015*. Cham: Springer, 309–326.

Lukovnikov, D., *et al.*, 2017. Neural network-based question answering over knowledge graphs on word and character level. *In*: *Proceedings of the 26th international conference on world wide web*, 1211–1220.

Mador-Haim, S., Winter, Y., and Braun, A., 2006. Controlled language for geographical information system queries. *In*: *Proceedings of the fifth international workshop on inference in computational semantics (ICoS-5)*.

Mai, G., *et al.*, 2021. Geographic question answering: challenges, uniqueness, classification, and future directions. *AGILE: GIScience Series*, 2, 1–21. Available from: https://agile-giss.copernicus.org/articles/2/8/2021/.

Mai, G., *et al.*, 2019a. Deeply integrating linked data with geographic information systems. *Transactions in GIS*, 23 (3), 579–600.

Mai, G., *et al.*, 2019b. Relaxing unanswerable geographic questions using a spatially explicit knowledge graph embedding model. In: *The annual international conference on geographic information science*. Cham: Springer International Publishing, 21–39.

Mohit, B., 2014. Named entity recognition. *In*: *Natural language processing of semitic languages*. Berlin, Heidelberg: Springer, 221–245.

Nyerges, T.L., 1995. Cognitive issues in the evolution of GIS user knowledge. *In*: *Cognitive aspects of human-computer interaction for geographic information systems*. Dordrecht: Springer, 61–74.

O'Looney, J., 2000. *Beyond maps: GIS and decision making in local government*. Redlands: ESRI Press.

Punjani, D., *et al.*, 2018. Template-based question answering over linked geospatial data. *In*: *Proceedings of the 12th workshop on geographic information retrieval*, 1–10.

Sagara, T. and Hagiwara, M., 2014. Natural language neural network and its application to question-answering system. *Neurocomputing*, 142, 201–208.

Scheider, S. and Huisjes, M.D., 2019. Distinguishing extensive and intensive properties for meaningful geocomputation and mapping. *International Journal of Geographical Information Science*, 33 (1), 28–54.

Scheider, S. and Lemmens, R., 2017. Using SPARQL to describe GIS methods in terms of the questions they answer. In: *Societal geo-innovation: short papers, posters and poster abstracts of the 20th AGILE conference on geographic information science*. LEI Wageningen University and Research Centre.

Scheider, S., *et al.*, 2020a. Ontology of core concept data types for answering geo-analytical questions. *Journal of Spatial Information Science*, 2020 (20), 167–201.

Scheider, S., *et al.*, 2020b. Geo-analytical question-answering with GIS. *International Journal of Digital Earth*, 14 (1), 1–14.

Schwitter, R., 2005. A controlled natural language layer for the semantic web. *In*: *Australasian joint conference on artificial intelligence*. Springer, 425–434.

Sinton, D., 1978. The inherent structure of information as a constraint to analysis: mapped thematic data as a case study. *Harvard papers on geographic information systems*.

Ten Berge, T. and Van Hezewijk, R., 1999. Procedural and declarative knowledge: an evolutionary perspective. *Theory & Psychology*, 9 (5), 605–624.

Wielemann, J., 2019. *The semantic structure of spatial questions in human geography*. Thesis (Masters). Utrecht University. Available from: https://dspace.library.uu.nl/bitstream/handle/1874/384695/Thesis_Report_Joris_Wieleman.pdf.

Xu, H., *et al.*, 2020. Extracting interrogative intents and concepts from geo-analytic questions. *AGILE: GIScience Series*, 1, 1–21.

Yang, W., *et al.*, 2019. End-to-end open-domain question answering with bertserini. *arXiv preprint arXiv:1902.01718*.

Yin, X., Gromann, D., and Rudolph, S., 2021. Neural machine translating from natural language to SPARQL. *Future Generation Computer Systems*, 117, 510–519.

Zuva, K. and Zuva, T., 2012. Evaluation of information retrieval systems. *International Journal of Computer Science and Information Technology*, 4 (3), 35–43.