Data enrichment of spatial databases using ontologies and Bayesian networks

Alexander Melchior,
3040585

August 31, 2013

Abstract

In the *map generalization process* the supplied data often lacks the explicit information for a proper automated approach. This is a problem that is very apparent when we want to generalize to a *social construct*, a description of something made by a society, both for crisp entities like "a detached house" and *vernacular geographies* such as "a suburban area" or "the high street". In this thesis we will explore a new way of *data enrichment* in spatial datasets for the use of such generalization. We will model the entity that we want to enrich the data with as an *ontology* using OWL, try to exploit the hierarchical nature of these entities for modeling and find these entities with the use of *Bayesian networks* that are generated from the ontology. We have created a Protégé plug-in called BNGen as tool to convert ontologies to Bayesian networks and a code blueprint for the enrichment process framework. We will show that this approach works with an illustrative use case where we will enrich a dataset with the *leafy residential area* concept. While the use case is successful in showing that our approach works, it will also be shown that OWL is not good at modelling vague relationships where a relation might hold or only partially holds. To counter some of these problems, and the fact that Bayesian networks are not dynamic in structure, we introduce *summary nodes* in the Bayesian network and *staged classification*. It will also be shown that we can exploit the enriched data to deal with the vagueness of spatial concepts as social constructs in our approach.

Thesis for the degree of Master of Science from University Utrecht (UU) for the completion of the master Computing Science in collaboration with the University of Edinburgh (UoE).

Supervisors:
Prof. Dr. Marc van Kreveld (UU), Dr. Silja Renooij (UU) and
Dr. William Mackaness (UoE)

Contents

1 Introduction

In geographical information science there is a big research field on map generalisation. Much of the research has been focused at the *geometric aspects* of map generalisations and the concept of MRDB's (MultiResolution DataBases). An aspect that has been studied less is the use of *data enrichment* for map generalisation. If one, for example, would want to generalise a residential area consisting of multiple residential buildings, the desired generalisation could differ depending on the type of residential buildings. If the area mainly consists of terraced houses one might want to generalise this differently from an area that is mainly made up of flats. If such information is not available this can be added by a data enrichment process that classifies the buildings with type information and adds this information to the data. Such additional information is also useful for the construction of thematic maps and choremes, data mining and other analysis of the data as it can improve the results.

Usually the data enrichment process is one of manual labour where experts have to add information of their expertise to the data. Such work can be tedious, expensive and time consuming, making it very promising for automation. Most efforts to automate such classifications rely on pattern recognition algorithms that are tailor made for their specific task and input data, thus having poor re-usability value. The poor re-usability of such data enrichment approaches warrant a look at a more generic approach in both modelling and enriching the data.

1.1 Data enriching

Often datasets do not contain the explicit information needed for a decision making process [34]. Generalization is a type of decision making process as it needs to be decided how, where and when to generalize. In [32] multiple ways of generalization are presented, Figure 1 illustrates generalization by aggregation by taking the data on the left and returning the data on the right, which is done by a GIS (Geographical Information System(s)). The data enrichment process works in the opposite direction as we add more detail to the data instead of aggregating it.

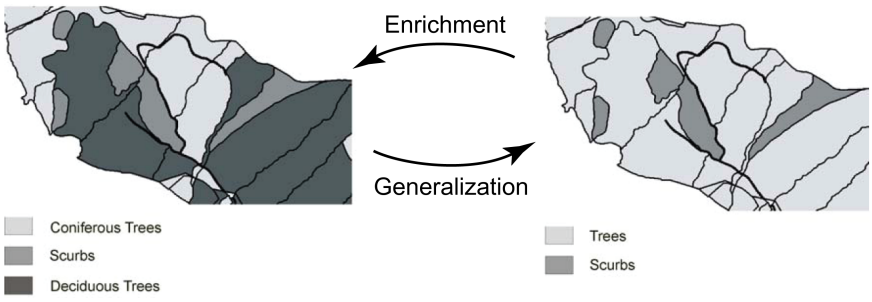

Figure 1: Figure adapted from [32] to show relationship between generalization and enrichment. *OS MasterMap data Ordnance Survey ©Crown Copyright*

A stronger cartographic example is to enrich a detailed dataset and to generalize the enriched data by symbolization. In the enrichment process we can add information about groups of objects, topology, proximities and many other relationships [34]. Usually this information is implicitly available in the data, but by making it explicit in the data enrichment process we can make use of it in generalizations. As a result we can create a MRDB with more pre-computed data which offers great potential to improve computational performance of applications [5]. One could say that we "specialize to generalize", which can be seen in Figure 2. The original

dataset is enriched with information about what type of areas are present in the mapped area. After the data has been enriched the generalization process takes the additional information into account and, in this case, generalizes this information using symbology to add symbols of trees and factories to the generalized map.

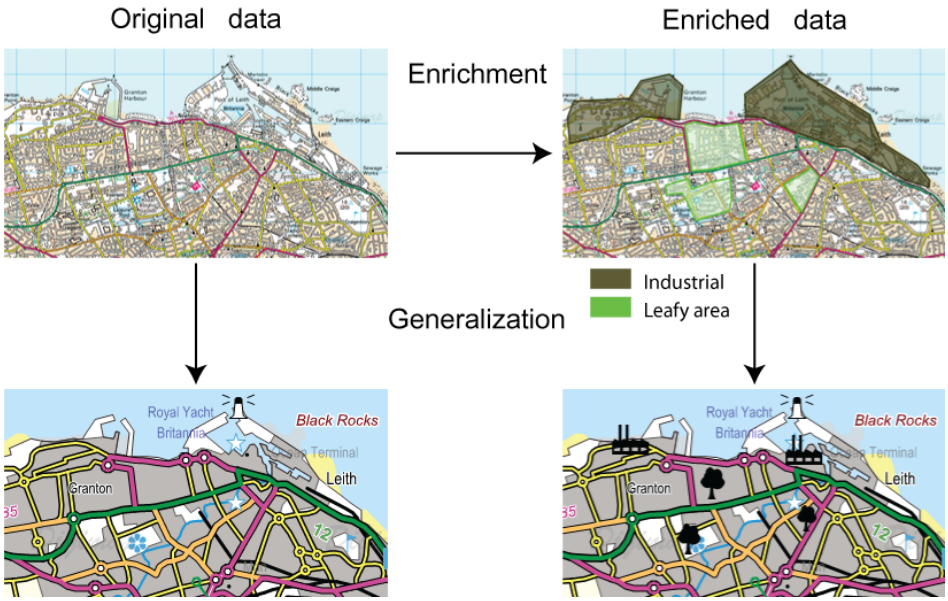

Figure 2: Original data VS enriched data where industrial and leafy areas have been classified. The enriched data is generalized by symbolizing to a factory or a tree. *OS MasterMap data Ordnance Survey ©Crown Copyright*

1.2 Research questions and outline

Before we can classify entities (elements in the data) we first need to create a model of what we want to classify them as. Due to our interest in GIS we will be focusing our efforts on entities in a spatial context. Modelling entities in a spatial context is not something new and is often not a formalized process. In efforts to formalize this process the concept of ontologies is introduced [1, 13]. Despite the use of a more formalized model for entities we notice that a model of a spatial entity will never be as crisp as one would want, e.g. what would a model of vernacular geography [39] like a *business district* be? As such we expect that there will be some vagueness and uncertainty in our model and classifications, thus making *Bayesian network classifiers* a likely choice as classifier as they seem to be able to cope with this [16, 35]. They also make an interesting research subject as their application in a geographical context is relatively new [29, 30].

This leads to the general research question of this thesis:

1. Is it possible to use a framework based on ontologies and Bayesian classifiers for data enrichment?

To get a good look at the possibilities, strengths and weaknesses of the approach and application to generalization we have devised three sub questions:

2. How well are ontologies usable in a spatial context?
3. Can we use ontologies for hierarchical modelling of entities?

4. Can we use Bayesian classifiers to classify entities modelled within the ontological framework?

With these three sub research questions we take a closer look at each step in the data enrichment process. Question number three is specifically aimed at generalization for multiple scales as these different generalizations can be stored in a hierarchical manner in MRDB's [3, 6]. Ontologies and Bayesian network classifiers will be explained in separate sections (respectively section 2 and 3) that will treat the preliminaries, give a theoretical insight in our questions and show how we might answer them. After this we will create a prototype based on our theoretical ideas in section 4 and put this to the test in the following use case in section 5. The thesis will be concluded by a discussion of our findings and conclusion in sections 6 and 7.

2 Ontologies

In order to be able to classify entities we need a way to describe these things in our world. Our world can *be* and *contain* many different things: a world that we perceive as being around us, a world that we see if we use our computer, the world that we can imagine when we look at data in a database, anything that we might or might not be able to think of. All these different representations show us different worlds in different ways. When we create a model of the world we usually want this model to be as close to reality as possible. A problem is that we often end up making a model that we don't understand of a world that we don't understand.

While we might not be able make a perfect model of reality we are often able to agree on a proper way to *reason about* and *model* reality within a group. For this problem of modelling and reasoning we can use *ontologies*, which is loosely translated from ancient Greek as the verb 'to be' or as 'being; that which is'[1]. With an ontology one can describe something in a more coherent, structured and consistent way than by "just giving a description" of the entity. One of the main motivations for this type of approach is the re-usability of and ability to share ontologies. Due to this more formal way of modelling, ontologies are used in the semantic web with standards such as XML (Extensible Markup Language) and RDF (Resource Description Framework) in the form of OWL (Web Ontology Language) and OWL 2 [19, 21, 31]. In the next sections we will take a look at what ontologies exactly are, how they can model spatial entities and how we can use this for data enrichment.

2.1 GIS & Ontologies

During the past two decades ontologies have been given more and more attention in different research fields such as Artificial Intelligence, Semantic Web, Biology and Geo Sciences. In [1] Bateman et al. give an overview of the use of ontologies for spatial representation and reasoning. Especially their overview on how geographical ontologies are used in GIS gives us a good view on multiple possible approaches of using ontologies in GIS. Other examples can be found in [28].

Fonseca et al. argue in [13] that the increasing amount of available data in the world, and mainly the integration of this data, calls for thinking about ways to improve the *interoperation* of data. In their work they address the *semantic aspects of geographic information integration*: "...the meaning of the entities that compose the ontologies representing concepts of the real world...". By making ontologies explicit one can prevent conflicts between ontological concepts and implementations since a lot of ontological concepts are easily understood by humans due to their

[1] Taken from the Wikipedia page on Ontology (5-12-2012)

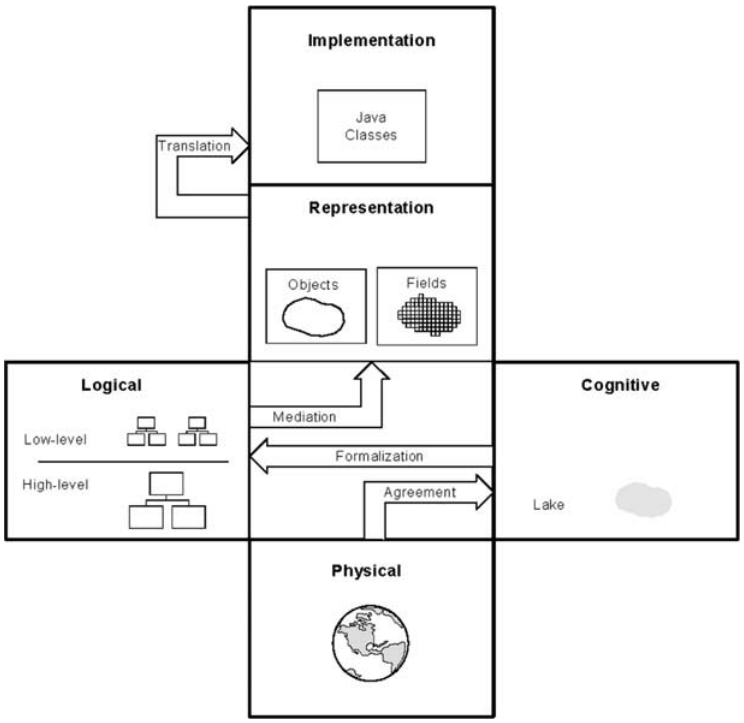

Figure 3: The five universe paradigm [13].

"common sense reasoning", something that's hard, if not impossible, to express in code. In the paper they base their *five universes paradigm* model on the four-universes paradigm [18, 4] and the work by Frank [14] who presents a five tiered system: *human-independent reality*, *observation of physical world*, *object with properties*, *spacial reality* and *subjective knowledge*. In the five universes paradigm there are the following five universes: the *physical*, the *cognitive*, the *logical*, the *representation* and the *implementation* universe. The five universes paradigm is illustrated in Figure 3.

Entities in the physical universe are interpreted by the human mind when they perceive these entities and the entity enters the cognitive universe. Once we try to formalize these cognitive entities we enter the logical universe and construct ontologies. Now that we have ontologies we can make choices on how to represent these: we enter the representation universe. As a final layer we have to construct this representation by implementing it: we have reached the implementation universe. This paradigm has given us an example of how ontologies can be used in GIS by implementing an entity from the physical universe and modelling it as an ontology.

For the implementation of ODGIS (Ontology-Driven Geographic Information Systems) Fenseca et al. [13] introduce the concept of roles: one entity can fulfil different roles. While roles are generally used to represent changes of an entity over time the main goal is to use roles to deal with *different points of view* of the same entity. Due to a loose interpretation of roles it is possible to stick to the same entities having different roles in different ontologies. Fonseca et al. state that "Each community has a right to its own point of view and information must be integrated on that basis, hence the use of a flexible specification of role". In a discussion in [31] it is also pointed out that the development of an ontology should be firmly driven by the intended usage.

We illustrate this idea by taking a look at a *lake*. While all points of view agree

that a lake is a water body, they might have different interpretations about the rest of the lake. A tourist board might see a lake as a tourist and recreation hot spot, a society of bird spotters might see it as a bird reserve and a transportation company sees it as a node in its water transportation network.

The problem of different conceptualisations of the world also comes forth in [27]: different classifications arise when different classification systems are used. Sometimes classifications simply don't exist in other communities. One example that is given is the definition of "river" in English, which overlaps with both the words "fleuve" and "rivière" in French while not representing the exact same concept. Other examples are the Dutch words "gezellig" (cosy, nice, good, friendly, ...) and "strooien" (sprinkling, gritting, throwing, ...), both are only translatable to English if one knows the context of the word while the words by themselves have proper meaning without any context in their own language.

To overcome these problems the usage of ontologies is suggested, as Lüscher et al. point out in their works [27, 28, 30]. Despite ontologies assuming only one reality in the classical interpretations, and thus having only one ontology for one entity, we can adjust an ontology to fit a particular point of view. This can, for example, be done by creating a basic ontology of an entity, give this ontology to different societies that each have a different point of view and let them change elements to fit their taste. The resulting ontologies will be alike as they had the same starting point, but the changes will be different as they don't share the same point of view.

Taking this reasoning, together with ideas the of Fonseca et al. in [13], we can think of an ontology as a *social construct* and not as the definitive truth: it is constructed by social interaction. A social construct is a model of an entity (an ontology) that is being approved by a society, where a society could range from an individual or a small group of people to all the inhabitants of a country or even all the people on Earth. The idea of social constructs is also supported by the ideas of Janowicz in [24] where he points out that there are many misconceptions about the construction of ontologies and argues that it is impossible to make a global ontology with a common agreement. The social construct concept is related to vernacular geographies which have been related to using fuzzy logic to define ambiguous geographic data [23, 39]. In [39] Waters and Evans state that "These [vernacular geographies] are not simply indicative - they often represent psychogeographical areas in which we constrain our activities, and they convey to members of our socio-linguistic community that this constraint should be added to their shared knowledge and acted upon.". As such vernacular geographies are a subset of our social constructs, we can both model crisp and ambiguous geographies using ontologies.

There are many more reasons and examples to give to support the notion of ontologies being a social construct, but the point to make here is that we can never construct an ontology that *everyone* agrees upon. Ontologies are usually created by a subset of the world's population, thus an ontology for the same thing (e.g. a residential area) can be created by two totally different groups of people. While both groups might fully agree on what their own ontology for a residential area is, work with it properly and get good results, the two ontologies might differ a lot from each other.

2.2 Hierarchies

We are able to make a hierarchy based on every relationship that can give us some sort of ordering. For example we could make a hierarchy on "medical service level" where we could have a person trained in first aid at the bottom, going up through entities such as nurse, general practitioner, small sized hospital up to a large university hospital. In this example we have a large mix of different types

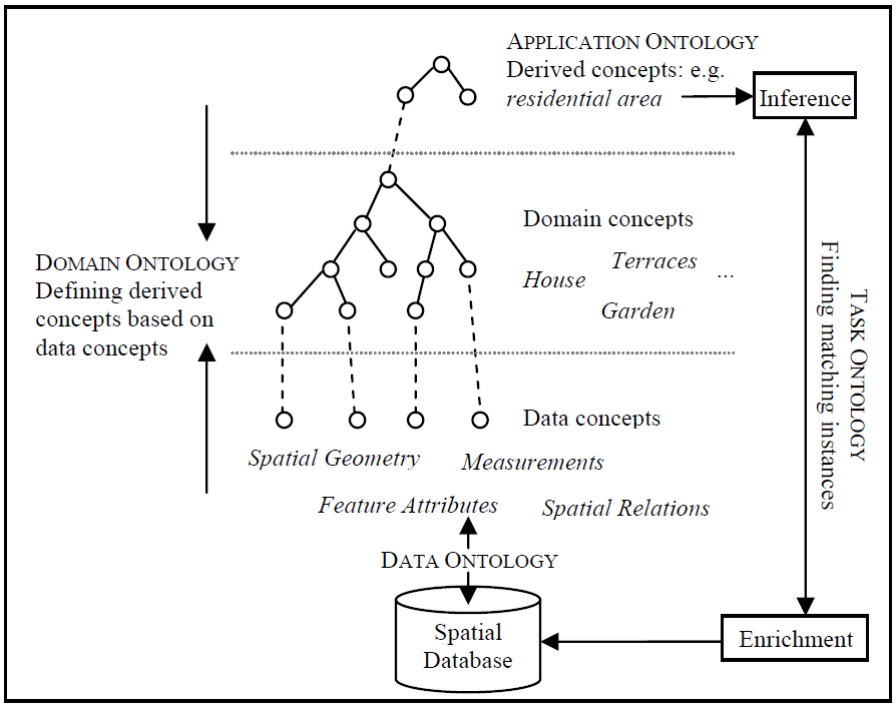

Figure 4: Relating higher-level concepts to the data structure and fitting this into the data enrichment process [38].

of entities, from spatial entities (hospital) to people (nurse) and many different explanations (like *size* or *level of training*) why the entities are ordered as they are.

One of the interesting things of spatial ontologies is that they have very apparent hierarchies. The most familiar to many people is the use of hierarchies as different *scales* as each entity is usually represented differently on different scales. On the smallest scale we see a lot of detail, which is usually aggregated, or displayed differently, on higher scales. This type of hierarchy is a partonomy ("part of" relationship) where aggregation is used for the generalization process and entities on the finest scale are part of aggregated entities. Along with partonomy we have taxonomy ("is a" relationship) [32] as another very apparent spatial hierarchy, e.g. a house *is a* building. Both these hierarchies are easy to model in OWL using the subclass properties of an entity. Interesting here is that the partonomy hierarchy is effectively the inverse of the taxonomy hierarchy. This shows that hierarchies in spatial data can go in different "directions", something Thomson also pointed out for these abstraction hierarchies [38]. The thing to realize here is that a hierarchy is a relationship that orders entities.

If we want to be able to construct ontologies and take advantage of the hierarchies we have to take into account what sort of entities we put in our ontology. Once this has been done we can construct an ontology by combining multiple sub ontologies to form the final ontology. Figure 4 illustrates the ontological hierarchies and how they could be used in a data enrichment process. These sub ontologies start at *primitives* (see [24]) and build up to bigger and more elaborate ontologies. For example an ontology of a house could contain the ontologies of building and residents. By changing the sub ontologies of an ontology we can adjust it to another social definition for the same entity while not having to rebuild the total ontology. We could also go into depth on one aspect in an ontology if, for example, there is a lot of data present on a particular aspect of the ontology or we have a question

that we want to answer that only focuses on a specific part of the ontology. If we had a full fledged ontology of a *business district* and we are interested in the transportation in this district we could take this part for the ontology and use this to answer our question. This gives us a structured way to create ontologies and an easy way to make small changes to particular details of the ontology, thus increasing the re-usability of the ontology.

3 Bayesian classifiers

Once we have made an ontological model of the entity with which we wish to enrich our data we need to create a way to find these entities. We can do this with classification: classify each entity in our data as being part of the class of entities that match the ontological model or as part of the class of entities that don't. In our use case, for example, we are classifying buildings into specific classes like terraced house and enrich our dataset with this classification data.

3.1 Preliminaries

While there are many different classifiers available we have chosen one of the most effective approaches: Bayesian network classifiers [16]. These classifiers are based on Bayesian networks [16, 35] and will give us the probability, or *belief*, that a certain entity belongs to a class.

An important issue that we have to note here is the difference between *vagueness* and *uncertainty*, two terms that are often misused [26, 11]. An example to show the difference, inspired by [11], is that of the classification of a greenish house. A greenish house is a house that has trees and foliage in the garden and is most likely to be found in a leafy residential area, see section 5 for the full definition. If we are *uncertain* and say "this is a greenish house" with 0.5 certainty we know that there is a 50% chance that it is a greenish house. But if we have to deal with *vagueness* and pose the same statement, "this is a greenish house" with a degree of truthfulness of 0.5, it might have 50% of the features and aspects of a greenish houses. In the former something is true or not (a boolean value) while in the latter there is room for different interpretations, a degree of correctness.

Both vagueness and uncertainty are present in our ontology and classification work flow. The hierarchical structure causes the existence of uncertainty or vagueness: something at the lowest level might be classified as belonging to a certain class, but this is most likely done with a certain belief. And as our model is a social construct there is no absolute truth about the entity that we want to enrich our data with either. For example we would have uncertainty in a dataset containing information about which houses have a shed in the garden if we know that 9 out of 10 values are correct. But if we interpret the same data knowing that it has 90% truthfulness we know that all the houses with a shed have something that is a lot like a shed and that the houses that don't have a shed have don't have a shed at all.

As such we have to take all of this into account in our classification. The problems of vagueness and uncertainty concerning data is not something new, the research fields of biology, semantic web, economics and artificial intelligence [35, 12] have coped with this problem for a long time. While uncertainty and vagueness are two different things and it is important that we understand that there is a difference, we are actually interested in the belief we have that something belongs to a certain class. So for this thesis we will work with *belief*, while it could be vagueness, uncertainty or a combination of both.

A Bayesian network describes a joint probability distribution over a set of variables. In its associated DAG (directed acyclic graph) each node represents a variable that we want to use and edges represent the conditional dependencies between these variables: associated with each node is a set of conditional probability distributions that together uniquely define the joint distribution [16, 35]. The probabilistic calculations in Bayesian networks are based on Bayes' Theorem:

$$Pr(h|e) = \frac{Pr(e|h)Pr(h)}{Pr(e)}$$

where h is a hypothesis, e is evidence and $Pr(h|e)$ is the probability that hypothesis h holds given the evidence e. Of course this result can only be significant if h is actually dependent on e, or put differently, influenced by e: $Pr(h|e) \neq Pr(h)$. This way Bayesian networks offer us a mathematically elegant and sound way to handle uncertainty [7], and more importantly, belief.

An often used example for Bayesian classifiers is the diagnosis of a patient by a doctor. Given the symptoms of a patient (evidence) an attending physician has to make a diagnosis (hypothesis). If there is enough evidence for a hypothesis the physician can start treatment, otherwise the gathering of more evidence is required. While this is usually a process in the human mind, this process has been modelled by Bayesian networks for classification purposes since the early 1990's [17]. In the same manner we can model a spatial entity from an ontology and use the ontological model to formulate the conditional dependencies between the hypothesis and the evidence.

A good observation in the light of uncertainty and vagueness is that our initial input *might* contain uncertainty and vagueness, but as soon as we start combining evidence for a hypothesis we will always introduce vagueness or uncertainty in the classification. This is due to the fact that the classifying node classifies something given its priors, which might not all be true or only hold a certain belief. So something can be classified as belonging to a class A while it doesn't fully fit the definition of the class.

3.2 Advancements

The idea of using Bayesian networks and ontologies together is not new and has been used by many, we refer to [2, 7, 12, 22, 25, 29, 33, 36] to name a few. While some of these approaches try to see how much different ontologies are alike [36] or incorporate uncertainty in their model [7], we want to combine them for classification. What is certainly striking is the structural similarity between a DAG of a Bayesian network and the RDF of an ontology [9]. This suggests that we might be able to use this similarity in the construction of the Bayesian networks that we are using for classification. If we use the ontology as starting point for our Bayesian network we can simplify the Bayesian network construction by using the relationships that are present in the ontology. The Bayesian network for the classifier would always have the same type of structure because the entity that we want to classify will be the *sink* of the network and every other node an *ancestor* of this sink, giving us a directed graph. Any other node is able to have multiple children and parents in the network if this entity has multiple relationships in the ontology as well.

Because we will be working in a spatial context there is an other issue that we have to take care of as well: dynamic relationships. Standard Bayesian networks are static in structure while entities in a spatial context might have one-to-many relationships. A building might, for example, be adjacent to multiple other buildings. Unfortunately we do not know how many other buildings it will be adjacent to. In our Bayesian classifiers we counter this problem by introducing *summary nodes* to our network. These nodes will represent if the entity that we are classifying satisfies

the constraint that has been related to this relationship, e.g. "is this entity adjacent to 2 or more other buildings?".

The static nature of Bayesian networks prevents us from creating one big Bayesian network for the classifier if the related entity is not explicitly available in the data. If we want to know if something is adjacent to 2 or more houses then we first have to classify entities as houses before they are explicitly available in our data. As such we might have a *staged classification* that makes a new type of belief explicit in the data during each stage. To create these sub Bayesian classifiers we can use the sub ontologies that make up the main ontology of the final entity that we want to classify.

4 Prototype implementation

In [30] Lüscher et al. describe the following approach for the use of ontologies in pattern recognition and data enrichment:

> [...]we pursued a top-down approach to cartographic pattern recognition of urban structures. The individual steps of this ontology-driven approach are illustrated in Figure 5: Based on textual descriptions of urban spaces extracted from the literature, we identify specific urban patterns (step 1); we then formalise these patterns, their context and hierarchical composition based on ontological descriptions (step 2). The ontological definitions of patterns are then used to deductively trigger appropriate low level pattern recognition algorithms (step 3) in order to detect them in spatial databases (step 4).

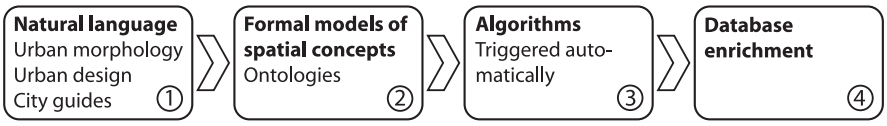

Figure 5: Steps in the processing chain of ontology-driven pattern recognition [30].

In this approach we see the basic steps towards both the creation of an ontology (step 1 and 2) and the use of the ontology (3 and 4) which we can interpret as two separate pieces of the puzzle. Fonseca et al. used a structure in the JAVA programming language for ODGIS [13] where they used classes for entities, which matches well since instantiated classes are objects in the world of programming. Most relationships can be given by operations on the object and subclasses can represent a specialisation of an entity. Both these works give us a good idea how to use ontologies in our prototype, but before we start constructing our prototype we look at what is currently available and if we can use this. After this we will discuss our own implementation for the enrichment process and use this for the use case in section 5.

4.1 Comparison of current implementations

With the development of OWL and the semantic web as a whole there have been many different types of attempts to combine Bayesian networks with ontologies and some prototype implementations. We will briefly discuss the largest categories of implementation to show the current state of the art and see how they might be relevant to data enrichment.

4.1.1 Annotation

Proposals in this category consist of taking an original OWL ontology and adding probabilistic information, like conditional probabilities, for the construction of Bayesian networks by annotating this information with additional markups. Examples of this type of work are, among others, OntoBayes [40], the work by Z. Ding et al. [9] and BayesOWL [10].

To construct a Bayesian network rules need to be created in a parser that takes the ontology and the additional markup and gives us the Bayesian network in the desired format. This is a simple and straightforward way of working that does its job well for basic networks, but lacks greater expressive power to work with first-order logic [7].

4.1.2 PR-OWL

In the motivation of PR-OWL, Costa et al. [7] try to overcome the lack of this greater expressive power. In basic annotated adaptations one can only express *attribute* knowledge and not *relational* knowledge. An example of relational knowledge would be to modify the probability of classification of all entities that adhere to a certain relationship, e.g. "All trucks that are within 100 kilometres of the distribution point are more suited for delivery x".

PR-OWL uses MEBN (Multi Entity Bayesian Network) as its first-order logical basis, the goal achieved here is found in the following quote taken from [7]:

> A major concept behind PR-OWL is that of probabilistic ontologies. Probabilistic ontologies go beyond simply annotating standard ontologies with probabilities, providing a logically sound formalism to express all relevant uncertainties about the entities and relationships that exist in a domain.

All the details can be found in the definition of the probabilistic ontology, but effectively they require "a proper representation of the statistical regularities and the uncertain evidence about entities in a domain of application". For this it is required that the semantics and abstract syntax of OWL is extended, a step further than adding annotation.

4.1.3 BNTab

BNTab[2] (Bayesian Network Tab) is a Protégé[3] plugin that translates an OWL ontology into a Bayesian network that can be used in Norsys Netica[4], a program to work with Bayesian networks. BNTab is a result of the practical need of Stefan Fenz to be able to model ontologies and convert them into Bayesian networks [12].

The approach Fenz takes is close to what we need but its overly complicated interface and lack of a relational structure in the interface make it hard to use. Also is Netica a commercial package which requires a full license if we want to create large networks or use all of its features, which limits the power of BNTab. Both these factors make us prefer constructing Bayesian networks manually with freely available software packages without the limitations that Netica has. Yet the idea of a plugin-in for Protégé to construct a Bayesian network from an ontology is very appealing.

[2]BNTab plug-in website: `http://protegewiki.stanford.edu/wiki/Bayesian_Network_Tab_(BNTab)`

[3]OWL editing tool, Protégé website: `http://protege.stanford.edu/`

[4]Norsys Netica: `http://www.norsys.com/netica`

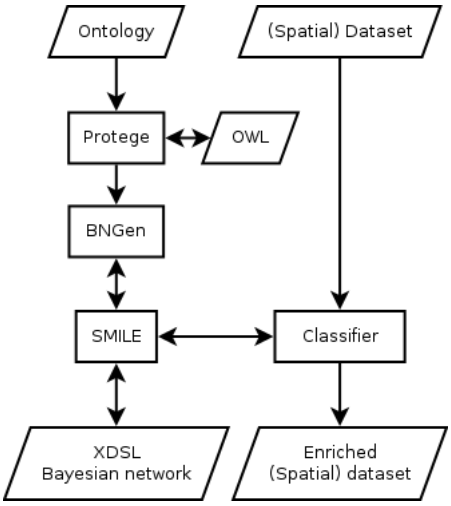

Figure 6: The work flow of the implementation, squares are processes and parallelograms are data.

4.2 Implementation overview

Now that we have described the theoretical background and looked at the current available solutions it is time to translate this into an actual prototype. A work flow diagram of the implemented prototype can be found in Figure 6. If we would compare Figure 6 with Figure 5 we could say that steps 1 and 2 are portrayed on the left of 6 while steps 3 and 4 are portrayed on the right. For our implementation we choose to use OWL to represent our ontology as it is the current standard for storing ontologies. The OWL Working Group states the following[5]:

> The OWL Web Ontology Language is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema (RDF-S) by providing additional expressive power along with a formal semantics.

With Protégé we can parse, edit and store ontologies from different types of OWL files and other ontology formats. Protégé also offers the option to add our own plug-in written in JAVA to Protégé. Our Protégé plug-in BNGen (Bayesian Network Generator) enables us to click a Bayesian network together and save it as a Bayesian network xdsl file using jSMILE[6].

Our classifier is also written in JAVA and supports shapefile datasets, using the GeoTools JAVA GIS toolkit[7], and the xdsl Bayesian networks as input. The classifier takes care of the connection with the Bayesian networks and deals with the shapefiles: the user has to specify how and which data is fed to the Bayesian networks. The output is freely specifiable by the user, but the use of shapefiles or other standards that are supported by GeoTools are the most straightforward to use.

[5]OWL Working Group website: `http://www.w3.org/2007/OWL/wiki/OWL_Working_Group`
[6]SMILE JAVA API: `http://genie.sis.pitt.edu/wiki/JSMILE_and_Smile.NET`
[7]`http://www.geotools.org/`

4.3 Protégé plug-in: BNGen

The BNGen Protégé plug-in is partly inspired by BNTab [12], but only on the visual aspects of the plug-in. Where BNTab uses *Netica* as Bayesian modeller we have chosen to use jSMILE, the JAVA API of SMILE. SMILE is a free C++ framework and supports multiple types of Bayesian networks, multiple file formats (including the Netica file format) and has a stand alone GUI (Genie[8]).

Once the user has selected an entity he/she wants to classify BNGen helps the user by only showing relevant parts of the ontology to the user based on the *entity hierarchy* (taxonomy) and *object properties* (including partonomy), as seen in Figure 7 on the left. Compared to BNTab we limit the amount of information that is provided to the user by not showing irrelevant information. When a variable and a dependency are selected in the interface the user can add the relationship to the Bayesian network by pressing the "add relationship" button, BNGen will add the relationship to the selected entities list on the right where the variables and their relationships for the Bayesian network are displayed. The plug-in also provides the option to create *reachable by* parts in the Bayesian network. These provide a way to link two entities together using an object property of the classifying node. Finally BNGen provides the option to select a state space for the added nodes. If no state space is selected the node will have two default states that SMILE initially provides for each node.

Once the user has created the Bayesian network it can be saved to the file system and viewed or edited by the user using editors like Genie. Currently BNGen lacks support for *data properties*[9] and the option to set the probability of the selected states. As such they have to be set in a Bayesian network editor after generating the Bayesian network.

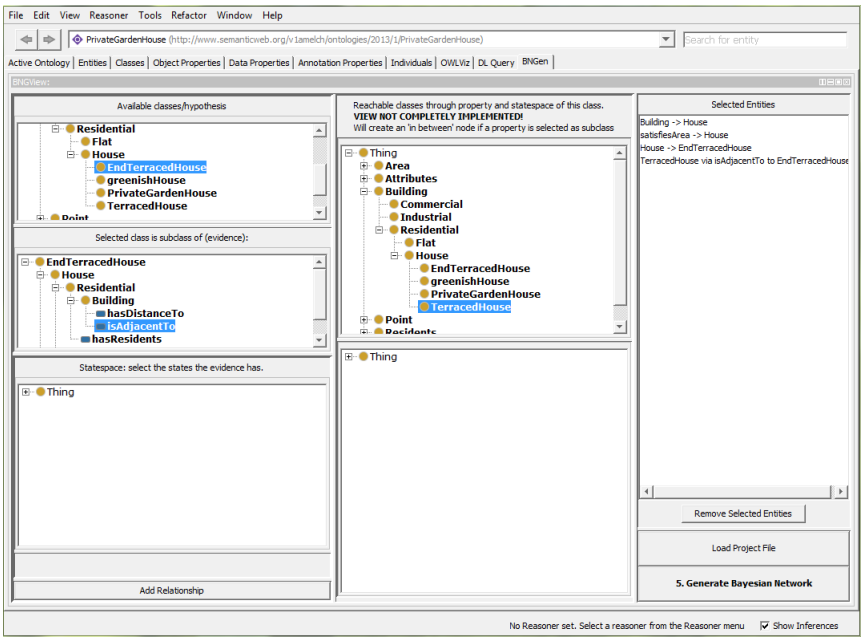

Figure 7: A screen shot of BNGen using the ontology from Figure 8.

[8] Genie website: `http://genie.sis.pitt.edu/`

[9] A way to assign data values to entities in Protégé.

4.4 Classifier

The classifier is the working horse of our implementation as it does, once fully implemented by the user, the following things:

- Connect with the Bayesian networks using jSMILE.
- Convert the data to a preferred data structure for the classification process.
- Classify entities in the dataset.
- Return enriched data in desired output.

In the current setup the classifier is compatible with all GeoTools features when working in Eclipse with Maven[10]. The initial classifier provides a code blueprint for the user to read shapefile data, create cases for the data enrichment process, execute the actual data enrichment and return this data as shapefile. This does require that the user knows JAVA but offers the most flexibility in return.

5 Use case: the leafy residential area

In order to put our framework into a geographic context and to see how well it works with real applications we have created a use case. When looking at available data from OS (Ordinance Survey) MasterMap[11] we notice there is explicit data available on small areas like buildings and roads. For map generalization purposes one might want to use such information, as shown in Figure 2 as a possible example. We have chosen to enrich the data with information about *leafy residential areas*, but one could also think of entities such as "the high street", "industrial areas" or "the business district". The real challenge here is to classify a leafy residential area with the limited data that is available in OS MasterMap. We also face a difference in scale: the OS MasterMap data has high scale information like exact geometric details on the footprint of each building while the leafy residential area concept works on a lower scale.

This use case suits our research questions well as we have to work with various hierarchies and try to model an entity that is not explicitly represented on the map in any way. What is also important is that while leafy residential areas are not explicitly mentioned on a map, humans are able, with some work, to Figure out where they are. With the use of ontologies and Bayesian networks our classifier should be able to do the same job, if not better. We have to stress that while this use case will try to classify leafy residential areas and return an enriched dataset, using our best understanding of the concept leafy residential area without extra verification by experts, it is an illustrative example to show how we can apply the described approach in this thesis.

5.1 Definition

We need to properly define leafy residential area, a term that is mainly used in common day speech, before we can create an ontology. If we look for a formal definition from the Oxford British & World English dictionary[12] we find the following:

Leafy *adjective* (leafier, leafiest) having many leaves or much foliage:

[10] Apache Maven is a software project management and comprehension tool, website: `http://maven.apache.org/`

[11] See section 5.2 for more information, website: `http://www.ordnancesurvey.co.uk/products/os-mastermap/index.html`

[12] `http://oxforddictionaries.com`

a leafy glade

leafy bushes

the leafy suburbs

Residential *adjective* designed for people to live in:

private residential and nursing homes

providing accommodation in addition to other services:
a residential sixth-form college

occupied by private houses:
quieter traffic in residential areas

concerning or relating to residence:
land has been diverted from residential use

Area *noun* a region or part of a town, a country, or the world:

rural areas of Britain

people living in the area are at risk

[with modifier] a space allocated for a specific use:
the dining area

a part of an object or surface:
areas of the body

If we combine these definitions we can describe a leafy residential area as "a part of the world designed for people to live in with much foliage".

A search on the web for "leafy residential areas" confirms this idea: the main results are sites that sell homes in or holidays to regions that are being described as being "a leafy residential area" while the corresponding pictures show residences that are being surrounded by foliage and trees.

But what is an *area* exactly? In [20], area is mostly translated as region or place: "In a generic sense, a place is a geographical locale of any size or configuration, comparable to equally generic meanings of *area*, *region* or *location*.". As such we find for region [20]:

> **region** Most commonly used to designate: (a) an area or zone of indeterminate size on the surface of the Earth, whose diverse elements form a functional association; (b) one such region as part of a system of regions covering the globe; or (c) a portion of one feature of the Earth, as in a particular climate region or economic region.

And in the light of our goal to create an ontology the following from [20] is particularly interesting:

> The region has been subject to much examination as to its epistemological and ontological status (see *epistemology*; *ontology*). How are regions to be known and represented? Do regions exist in actuality? It is probably safe to say that most geographers who have dealt with these questions agree that regions are based on socially constructed generalizations about the world, that their delimitation and representation are artefactual but not purely fictions.
>
> [...]
>
> Grigg took these criticisms to mean that the region, especially its use by geography, needs always to be understood as a means to an end and not

> an end in itself (cf. Hartshorne, 1939). The point of 'doing' the region is not ultimately to divide the world into regions and rest content. It is rather, if one wishes, to engage in classifying and modelling geographical phenomena so as to generate questions about their variability and functioning with respect to other phenomena.

As such we can generalize one part of the world as an *area* and define it as a region between certain points or coordinates in the world. Both *residential* and *leafy* are modifiers of what type the area is. If an area is classified as a certain type it satisfies certain constraints, in this case on the type of buildings and the amount of foliage. (Un)fortunately there are many ways to describe these constraints, some of these ways are:

- Minimum amount of leafy residential entities in the area.
- Maximal distance between leafy residential entities in the area.
- Minimum ratio of leafy residential entities compared to all the other entities in the area.

Where leafy residential entities are entities that have a positive influence to an area being a leafy residential area. One has to choose a type of constraint depending on what one wants to define, but this could also depend upon aspects such as available data or possible models. As such we will have to make this choice during the implementation process.

5.2 Data

The data that we will be using for this use case is from the OS Mastermap (MasterMap is the academic and commercial digital portal of the OS to access their data via the internet) Topography Layer. This dataset contains all of the United Kingdom in vector format with some basic attribute information. Of each entity (called *feature* in OS MasterMap) we are given, among others, a theme attribute, a description group attribute and the geometry in WKT (Well-known text). The themes that we are interested in are the themes "buildings", "land" and "road or pathway" which tell us what a feature is. The dataset contains 3 different scales: 1:1250 for urban, 1:2500 for rural and 1:10000 for mountains and moorland, all with a 99% accuracy confidence. Our selected dataset contains about 300.000 features in the urban part of Edinburgh as this is the most practical for our fieldwork later on, as such our scale will be at least 1:2500. The data is regularly updated and the most recent publication of new data was in April 2013. More information can be found in the OS Mastermap Topographic Layer manual[13].

5.3 Ontology

We have modelled the leafy residential area ontology in Protégé while taking the available data (if a feature is a building, land or road) into account as this defines our initial building blocks for the ontology. As such, buildings are our starting point in the model which we can extend into residential, commercial and industrial buildings with our model. Next we extend residential into houses and other types of residential buildings and extend houses into different types of houses such as terraced houses. This gives us, amongst others, the following hierarchy: terraced house *IsA* house *IsA* residential building *IsA* building. In the same manner a *leafy residential area* extends a *residential area* which extends *area* in turn.

[13]OS Mastermap Topographic Layer manual: `http://www.ordnancesurvey.co.uk/oswebsite/docs/user-guides/os-mastermap-topography-layer-user-guide.pdf`

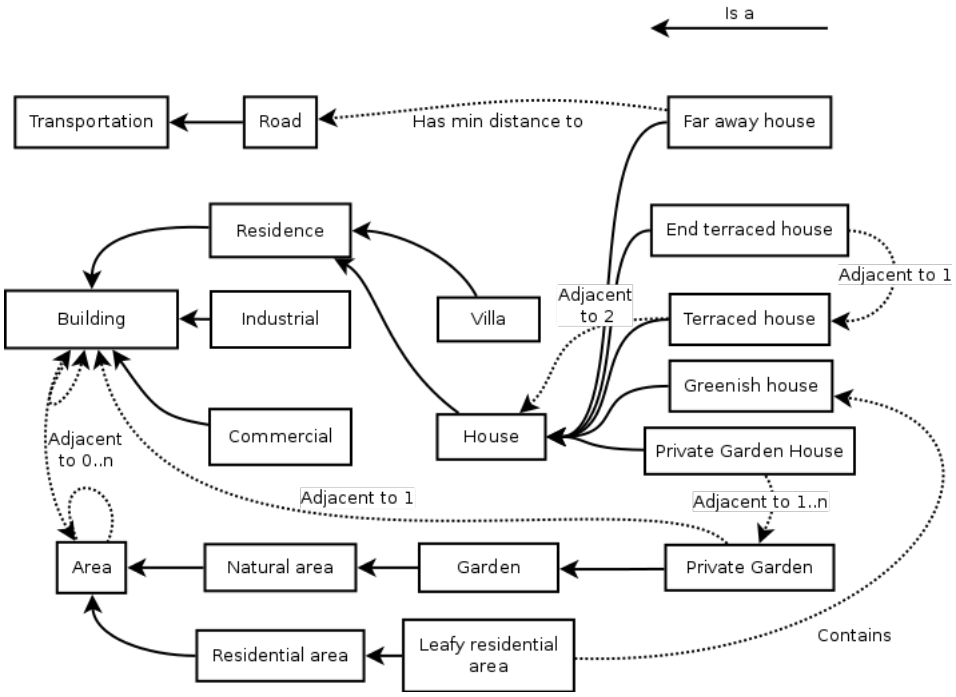

Figure 8: A graphical representation of the ontology.

A graphical model of the ontology can be seen in Figure 8. The solid arrows represent *IsA* relationships while the dotted lines represent relationships (object properties) with other entities denoted by the text near them. Some information that is in the ontology is not visible in the model in Figure 8 to improve the comprehensibility. One of these is the *disjoint* property: is an entity disjoint with another entity or could it be multiple entities at the same time? E.g. a pizza could be both a *spicy pizza* and a *meaty pizza*, but not a *vegetarian pizza* and a *meaty pizza*.

To find a leafy residential area we need to be able to classify the elements that give us an indication of where such an area is. In section 5.1 we called these elements leafy residential entities and will now model *greenish houses* as a leafy residential area. We have chosen to only classify greenish houses to focus on a single type of entity and because this (a type of house) seems to have good distinctive power for our use case. While we have multiple options on how to model a greenish house in the ontology we are limited by the available data and don't want to make a very complex model for the ease of analysis of our approach. We define a greenish house as:

- It is most likely a house.
- It is most likely not a terraced house.
- It most likely has a private garden.
- It is most likely not close to the road.

Where a terraced house is defined as a house adjacent to exactly 2 other houses, a private garden is defined as a garden adjacent to exactly 1 building and "not close" is a distance defined by a parameter set by the user. The use of *most likely* shows one of the interesting bits of geographical entities and reasoning with it. There are examples of terraced houses that are very leafy, but generally they aren't. The same

goes for all the other criteria, there are no clear cut boundaries on when a building is or isn't a greenish house.

Another choice that we made is not to force as much as possible into the attributes of the greenish house. Thus we have created a *private garden house* and a *far away house*, where the former indicates if a building is a house and has a private garden and the latter indicates if it's a house that is far away from the road. This enables us to have a better insight in the classification process later on as we can clearly see the in-between results of the classifications. We also notice that there is no hierarchical relationship between a greenish house and the other types of houses because they are all a type of house.

While it was a choice for the private garden and far away houses we are forced to do the same thing for terraced houses. A row of terraced houses is made up of terraced houses and two *end terraced houses* which mark the endpoints of a row of houses. End terraced houses are defined as being a house and being adjacent to exactly 1 terraced house. To be able to do this we first need to classify all the terraced houses before we can classify the end terraced houses. This order of classification is depicted by the ontological object properties such as *adjacentTo*.

5.4 Bayesian networks

Now that our ontology is created we can create the Bayesian network with the help of the BNGen plugin in Protégé. With the plugin we select the needed entities shown in Figure 8 and use the relationships depicted by the arrows connecting them. The Bayesian network for a terraced house (on the right in Figure 9) has one node as classifier, *terraced house*, and two nodes that influence the probability of this node: *house* and *adjacentTo*. House represents the belief that our entity is a house, which is calculated with the Bayesian network on the left in Figure 9, and *adjacentTo*, a summary node, the belief that it is adjacent to two houses. The belief of adjacentTo is constructed by checking if our current entity is adjacent to two other entities that have been classified as house with a belief of more than 50%. If this is the case the node gets the value *true* by assigning a 100% probability to the state that represents true.

The two Bayesian networks in Figure 9 could have been one network if *isAdjacentTo* did not need to know if entities, that are adjacent to the entity that we are classifying, are houses. This forces us to use the staged classification as discussed in the theory: first classify each entity with a house Bayesian network and then use this information for further classification in the next stage. The dashed line indicates the belief that an entity is a house being propagated from the first stage (house classification) to the next stage (terraced house classification). It also illustrates the need for the summary nodes: without this we would need to construct a Bayesian network in which each of the adjacent entities is represented by at least one node. As the number of adjacent entities is unknown in advance we would have to construct new Bayesian networks on the fly for each single case.

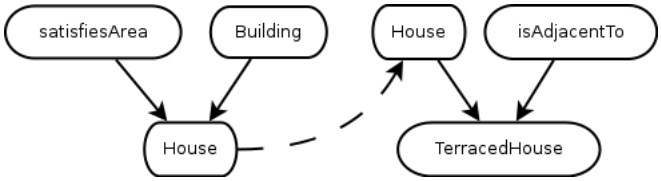

Figure 9: Two Bayesian networks, House(l) and TerracedHouse(r) where the belief of a house classification is propagated to the TerracedHouse network for the next stage of classification.

We have the same type of approach for all our other types of house, except for the greenish house. The greenish house doesn't have any relationships with the other types of house other than that they aren't disjoint. As mentioned before it could be one, or more, of the other types of house next to being a greenish house. But we don't know if it is, and if so, how much we actually believe it's one of the other types. This makes it impossible for us to create the Bayesian network for a greenish house with BNGen since we can't select relationships, including the taxonomy and partonomy, that are not *explicitly* modelled in the ontology as BNGen only shows relationships that are modelled in the ontology. Thus we have hand crafted this network ourselves in Genie so that we can still use it in our classifier as seen in Figure 10.

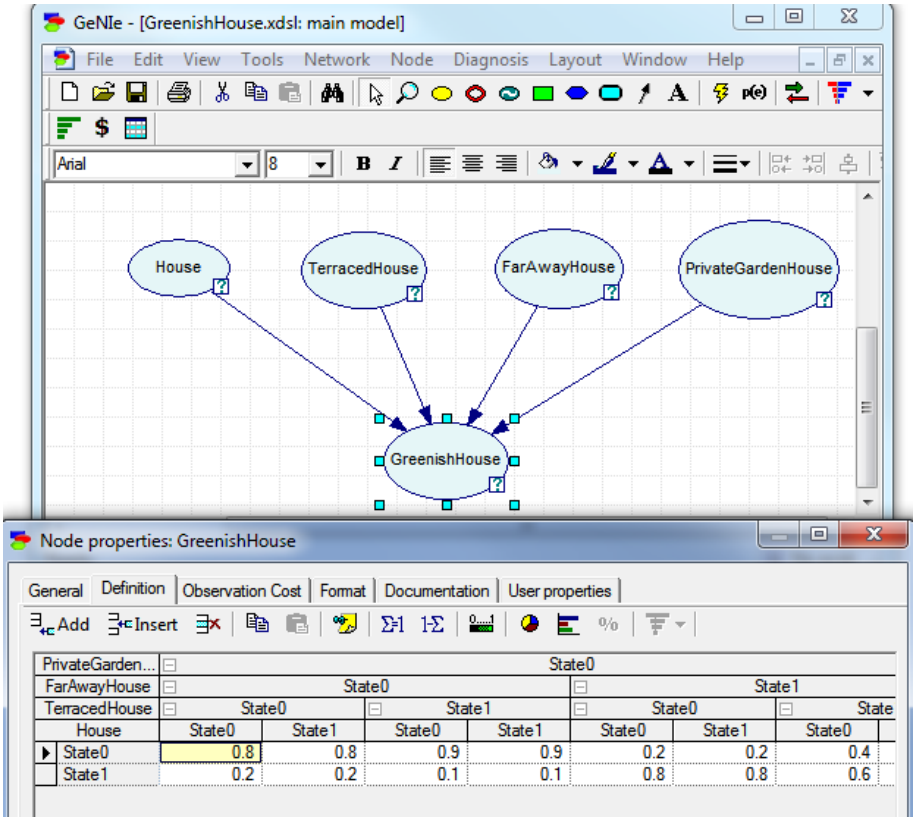

Figure 10: Screen shot from Genie showing the greenish house Bayesian network.

As BNGen doesn't provide us with a way to set probabilities yet we have to do so with SMILE or Genie manually. SMILE both offers the options for learning algorithms to learn the probabilities from sample data and for the user to set the probabilities manually. We've chosen for the latter because we want to have a clear view on the effects of the probabilities on the resulting output. In the final versions of the Bayesian networks we have chosen probabilities for the enrichment process that reflect our best understanding of the entities, but we stress that they are not chosen by experts in the field of residential entities.

5.5 Classifier

Our classifier has certain expectations of and assumptions on our dataset. As buildings can not intersect other buildings the dataset can be represented as a single

planar graph, e.g. there are no edges that cross each other. If this constraint holds the creation of a topological data structure becomes a lot easier, saving us a lot of time and effort. We expect the constraint to hold because our data is cartographic in nature and has to be printed on a two-dimensional plane without creating overlapping features. Unfortunately the dataset supplied by the OS doesn't satisfy this constraint due to small geometric errors causing the need to clean the dataset first. We load the data into ESRI's ArcGIS[14] and remove all the features in the data that overlap with other features in the data. An advantage is that Arc map is able to store the resulting dataset as shapefile, the format the classifier prefers and saving us an extra data conversion.

Removing parts of our dataset affects our final result as we can't return these in our enriched dataset or take them into account during the classification. Fortunately the removed features are relatively unimportant to our use case: roughly 2% of the features are removed and most of these are natural features that are ignored in the enrichment process like stretches of grass between a road and the pavement.

5.5.1 Classification

Once we have loaded the cleaned shapefile into the classifier we need to create a data structure with *topological* information next to the geometrical information that is available in the shapefile. We have chosen to create a DCEL (doubly connected edge list) data structure [8] of the dataset using a sweepline algorithm [8]. Because we have the guarantee that our input is a planar graph we don't have to deal with cases where edges would intersect each other, which would be the most complex cases for the algorithm to deal with. A DCEL gives us access to all the m neighbours of a feature in $O(m)$ time which is as good as it's going to get. Notice that we create a DCEL specifically for this use case and is not part of our general approach, but the creation of a topological data structure will most likely be required for most applications that work with our approach if such data is not readily supplied.

For each classification step we create cases that we let the Bayesian network classify, where every individual entity (called feature in OS MasterMap) in the dataset is a case. If, for example, we would want to classify a feature as terraced house we need to know how much belief we have that this feature is a house and if it is adjacent to two other houses. In the current implementation this last constraint is a hard constraint: if it is adjacent to two entities of which we have more than 50% belief that they are a house we have 100% belief that the adjacency constraint is fulfilled. When all the cases are created we give them to the Bayesian network and add the classifications returned by the Bayesian network to the dataset for further use. These classification are the belief that we have that something is a certain entity between 1 (100%) and 0 (0%), so an entity could have a classification `Private garden (0.000024)`. An example of the difference between the initial data and the enriched data can be seen in Figure 11.

5.5.2 Area creation

When the classification process is finished and we know for each feature how strong our belief is that it is a greenish house we start with the construction of the leafy residential area. We do this with a buffer algorithm and "grow" our leafy residential area: we pick a random element that satisfies a minimum belief constraint in our dataset, find all elements within a buffer created around this element that also satisfy the minimum belief constraint, create a buffer around the newly found elements and search for more elements that satisfy the constraint and repeat this for

[14] A GIS for working with maps and geographic information. Website: `http://www.esri.com/software/arcgis/`

Toid	123456
Theme	Buildings
DescGroup	Building
Make	Manmade
⋮	⋮

→

Toid	123456
Theme	Buildings, House (0.875), Terraced House (0.765)
DescGroup	Building
Make	Manmade
⋮	⋮

Figure 11: Example of initial data and resulting enriched data where the belief has a value between 1 and 0.

all the newly found elements. Once we can't find any more new elements that satisfy the minimum belief constraint we put all the found elements in a single set and remove them from the initial dataset. We continue this process until there are no more elements left in the initial dataset that satisfy the minimum belief constraint. Our result will be a set L of n sets consisting of greenish houses that are clustered together. The algorithm is illustrated by the pseudo code in figure 12.

areagrower(S, t)
Input: A set of features S that have been classified and a threshold t.
Output: A set of features L that represent the leafy residential areas.
G = the set of features in S of which we have more than t belief that they are greenish houses.
L = new arrayList[FeatureSet] ▷ L is an arraylist of FeatureSets
FQ = new FeatureQueue() ▷ Implemented as FIFO queue
while G.isEmpty() != True **do**
    LRA = new FeatureSet()
    $cf = G$.pop() ▷ A random element from G as current feature
    LRA.add(cf)
    $cfb = cf$.buffer()
    for All elements tf in G that intersect cfb **do**
        FQ.push(tf)
        G.remove(tf)
    end for
    while FQ.isEmpty() != True **do**
        $pf = FQ$.pop() ▷ Element to search around
        LRA.add(pf)
        $pfb = tf$.buffer()
        for All elements tf in G that intersect pfb **do**
            FQ.push(tf) ▷ Indirectly add element to this area and make sure we search around this one as well
            G.remove(tf)
        end for
    end while
    L.add(LRA)
end while
Return L

Figure 12: The growing algorithm for an area in pseudo code

A set L_i where $0 < i \leq n$ represents a leafy residential area which can be visualized by computing the buffer of all the elements in a set L_i, creating the union of all these buffers and creating a new feature (in this case a leafy residential area) that uses the result of the union as its geometry. If we do this for all n sets in L

we will have n features representing all the leafy residential areas in our data which can be visualized by tools such as ArcGIS. In Figure 13 we see hatched regions in the area of Morningside in Edinburgh which are leafy residential area features with a raster backdrop of the initial dataset.

We do all the geometric calculations (unions and buffers) with the use of standard functions of JTS[15], an API of 2D spatial predicates and functions that is included with GeoTools, and have chosen the parameters of the geometric buffers such that residential areas get grouped together and isolated entities stay isolated.

5.6 Results & discussion

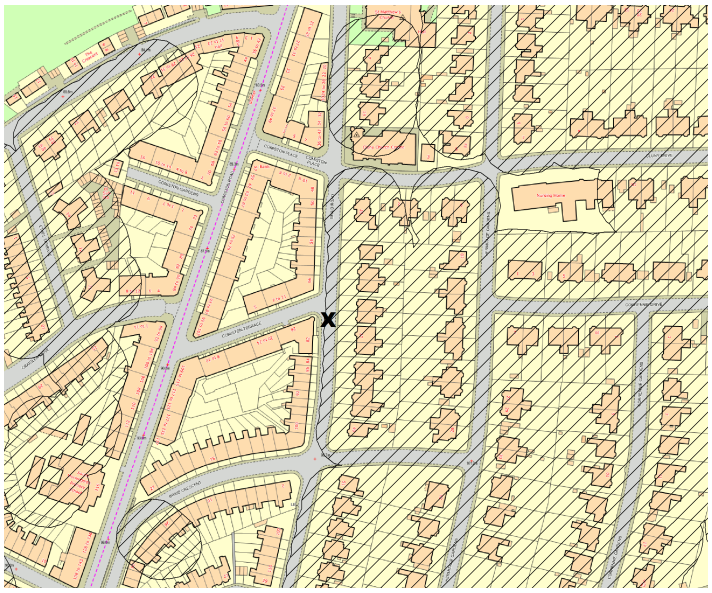

Figure 13: Part of the Morningside area in Edinburgh depicting leafy residential areas (hatched). The picture of Figure 14 is taken at the ×.*OS MasterMap data Ordnance Survey* ©*Crown Copyright*

In Figure 13 we can see a result of the classifier. The larger hatched areas depicted in the map mark a clear area that is supposedly a leafy residential area, but is this truly the case? When we look at the borders between areas that are leafy residential and that aren't we see difference such as in Figure 14 depicting a clear difference between classified areas and non-classified areas. In Figure 15 we see two examples of areas that are not classified as leafy residential areas and two examples of areas that are.

In Figures 16 and 17 we have the results of the classification of a larger part of Edinburgh. In Figure 16 we see all the areas that have been classified as leafy residential with more than 70% confidence while we see the subset with only large areas in Figure 17. In both Figures we see that the city centre has a very low number of leafy residential like spots. If we only look at the set of areas that are larger than 70.000m^2 we see one clear pocket just south of the city centre. We also see some areas that meet the size constraint but are very much stretched in length while others are more compact.

We have visited a number of points in our dataset to see how well our classifications fit, if we make errors and if so, what kind of errors these are. The areas that

[15]Website: `http://www.vividsolutions.com/jts/JTSHome.htm`

Figure 14: On the left we see houses that are in a leafy residential area while the houses on the right aren't. This picture is taken at the cross in Figure 13 facing south and shows a border between a leafy residential area and an area that isn't. Streetview image from Google Maps.

we would classify as real leafy residential areas during a visit had a real natural atmosphere with, being there during the summer, birds singing and the smell of grass and plants in the air. What was very apparent during the trip is that there are a lot of classified areas "quite like a" or "a potential" leafy residential area, or stated differently: "it is a leafy residential area with 70% truthfulness". We say this because these areas have the potential, which is partially being utilized, to be a real leafy residential area. The potential is in the fact that the houses have the space to plant trees and bushes in their gardens instead of the current stone walls, concrete ornaments, grass or car parks. Examples of this can be seen in the pictures in Figure 18 where the house on the left isn't very leafy but could be made like the house on right by planting more leafy elements in the garden.

We have also seen that the location and shape of the area is important as some areas are stretched along main access roads. Due to their proximity to such roads the houses have a big front garden with lots of trees and foliage but one would never classify these as (being in) a leafy residential area as they are next to a busy road. Another important and summarized effect is that of property value. In areas where the property value would appear to be lower[16] we noticed that there is less attention to foliage and trees while there is the same, or even more, space available than in areas where property values appears to be higher. Currently we see no clear relationship between the expected property value and the classified areas in our results.

The numerical results from the trip can be found in Figure 19. While we can count the areas that we have classified as being leafy residential areas (see Figure 17) we can't count the areas that we haven't classified. As such we have manually defined residential areas that are devoid of greenish houses as non-leafy residential areas. We have to stress that the selection of these areas are educated guesses looking at the density of greenish houses in an area given the classified data. The numbers indicate that we have about 56% accuracy, where accuracy indicates how many classifications are correct, if we only take the completely correct classifications into account and about 90% accuracy if we mark the potential leafy residential areas as correct as well. This last step is reasonable since we returned all areas of which

[16]Which can be found on websites such as Zoopla: `http://www.zoopla.co.uk/heatmaps/`

Figure 15: On the left we see two pictures of areas that the are not classified as leafy residential areas and on the right two that are.

we had 70% or more belief in, so some might only fit the ideal picture for 70%. When we compared our results with the real world we found that the real world is even more continuous than we expected. If we try to understand this further we have to look at vernacular geographies and fiat boundaries. Vernacular geographies argue that entities like a leafy residential area don't have hard boundaries [39] and that their "presence" gradually changes. These vague boundaries are also called fiat boundaries [37]: boundaries between entities that are not clearly definable. For example, where would the boundary between the English Channel and the North Sea be? While it is defined as "a line joining the Walde Lighthouse (France, 1°55'E) and Leathercoat Point (England, 51°10'N)"[17] this is an "arbitrary choice which part of all the water on the earth's surface we mark off and elect to call the "North Sea" " [15]. So unlike its boundaries with land, which are clearly demarcated by beaches and shores (also called bona fide boundaries), it is not clear where the North Sea abuts the English Channel [37]. So instead of having a hard constraint with hard boundaries to show areas that are classified with a certain amount of belief, it would also be interesting to create a heat map showing how much belief we have at a certain point. With such a heat map we could display the vague(fiat) boundaries and continuity of the areas in a more natural way.

To test our beliefs we have created a heat map, as seen in Figure 20, by giving each entity with more than 20% belief a colour that depends on its belief. The red buildings have the least belief, yellow average, green the most and the entities with less than 20% belief are removed to reduce clutter from entities such as roads. If we compare this with Figure 13 we see that the entities in the leafy residential area are indeed green and those that aren't in the leafy residential area are red while yellow is in between those two different groups. Next we have created an outer glow effect in an image editing program[18] for each entity such that areas with one colour

[17] `http://en.wikipedia.org/wiki/North_Sea#Extent`
[18] Adobe Photoshop

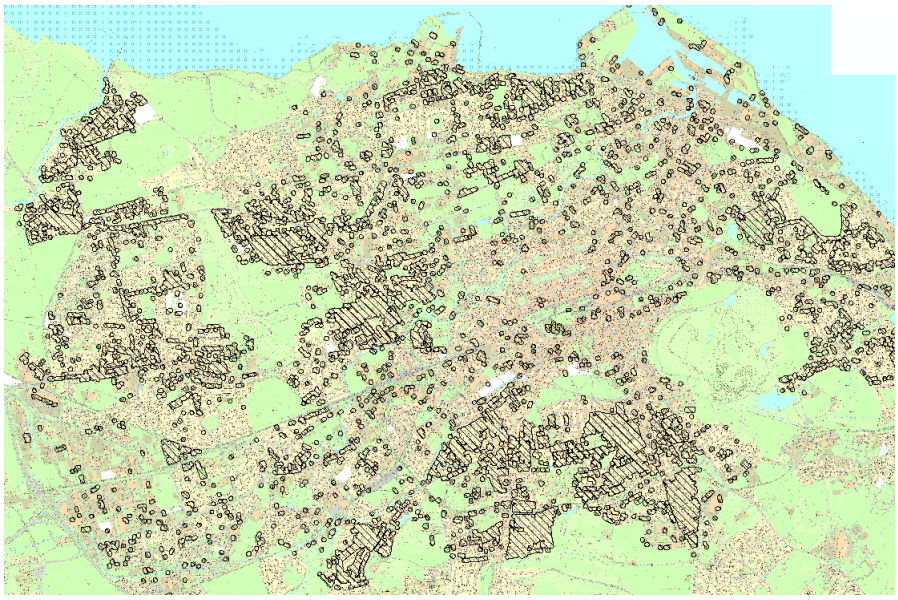

Figure 16: Map of Edinburgh where a large part has been classified. Hatched areas are classified as leafy residential. *OS MasterMap data Ordnance Survey ©Crown Copyright*

started to appear as the entities started to melt together. The effects were also configured to fade over into each other if they would overlap, creating most of the yellow areas between the red and the green areas. This fading effect is exactly what one would expect to see with fiat boundaries, vernacular geographies and entities that partly belong to a class. It also delivers the information in a clear and easy to interpret way for the human mind.

We have also found more things that could improve our classification after looking at the results and comparing these with the real world:

- Add address data for buildings: buildings with more than two addresses are most likely not a greenish house as they are apartment buildings or alike. These types of residence don't have a private garden as the garden is being shared by multiple dwellings in a single building. But due to the lack of such information they are wrongly classified as private garden houses.
- Add property value information to the dataset, buildings with a higher value are more likely to be in a leafy residential area. We expect this to be due to factors like:
  - Properties in a leafy residential area have a higher value because they are in a leafy residential area.
  - People living in higher valued properties are generally part of a social class that tends to care more about the leafiness of the surroundings in which they live.

  There are many more indirect relationships to point out, but properly specifying all of these would warrant another thesis.
- Take the proximity to the city center into account. Leafy residential areas that are currently classified that are in the city and not in the rural areas are more likely to be a real leafy residential area.

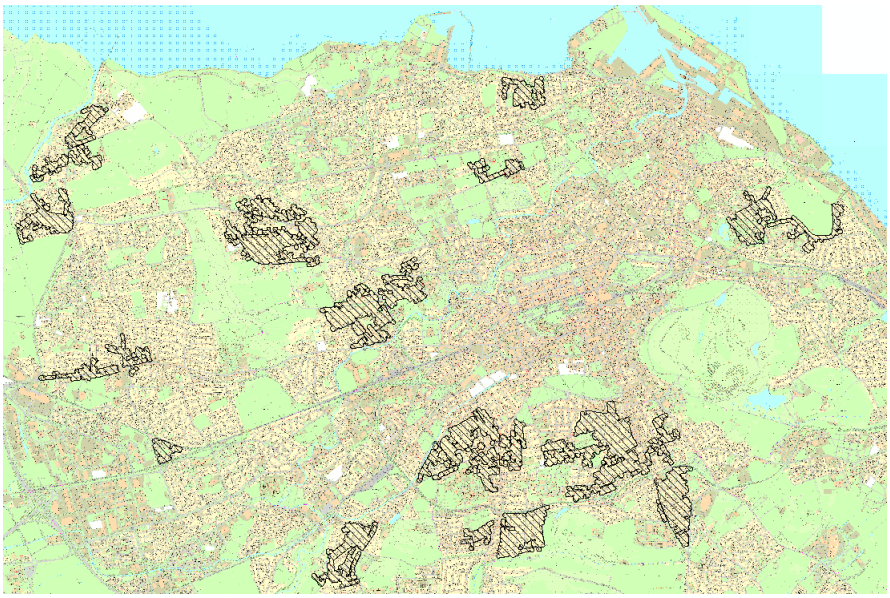

Figure 17: The areas from Figure 16 that are larger than 70.000m^2. *OS MasterMap data Ordnance Survey ©Crown Copyright*

Figure 18: A potential greenish house and a greenish house.

- The greenish house density also tells us how likely something is a leafy residential area. If there are a lot of greenish houses in a small area they are packed together and leave less room for gardens or other leafy elements in the area and make it less leafy.
- Take the geometry of the area into account. If it is a long stretch of houses it's along a single road and likely to not be a leafy residential area.
- Make use of remote sensing to get information like building height or the composition of foliage in an area.
- Add more detailed theme information to our initial dataset. This would make our basic building blocks less uncertain and vague. For example information if something is a house or not could be added removing the need to add this by classification.

All these improvements could be added to the classifier and be incorporated by extending the ontology and Bayesian networks. With these improvements the classifier would potentially be more accurate both in belief and how correct this

	Reality		
	clear LRA	potential LRA	non-LRA
LRA in output	13	11	1
non-LRA in output	0	2	5

Figure 19: Comparing the output of our classification with the real world situation where LRA stands for Leafy Residential Area.

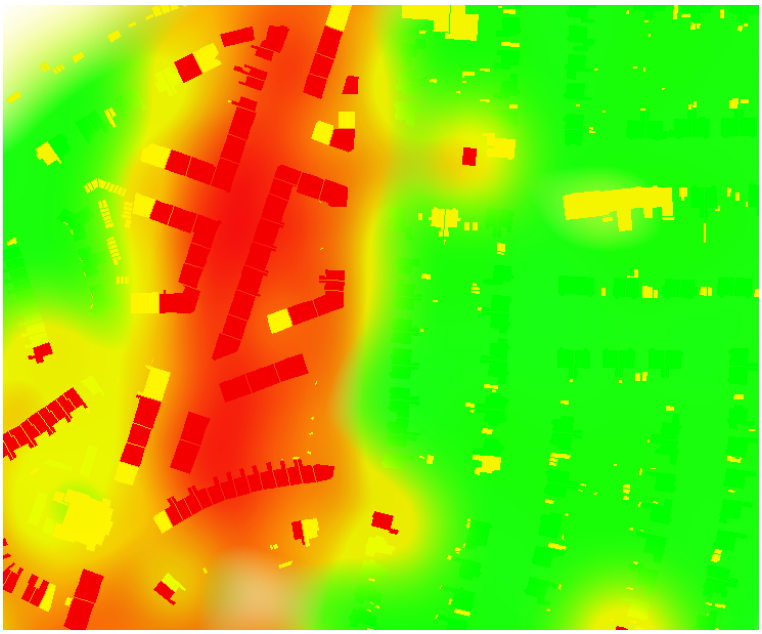

Figure 20: Heat map of the same area as in Figure 13 using real belief values.

belief would be. However, one drawback of these improvements is that they require more (manual) work and thus reducing the amount of profit that we have from the automation. It will be important to find a good balance between improvement and the required effort to do this improvement.

5.7 Conclusion

Even without the proposed improvements we are able to point out certain areas in the city that contain leafy residential areas using only basic spatial data and a flexible ontological model. The numbers show that our framework is fairly accurate with the data that we have at our disposal. Also has our heat map shown that we can get useful results without exactly knowing where the boundaries are by showing transitions between areas. The biggest drawback of the approach that one could point out is the fact that we have been forced to create one Bayesian network manually.

Nevertheless this use case has taken a real geographic and spatial data enrichment problem and managed to enrich our data in a useful way, thus showing that our approach works. Given the framework and the tools (BNGen and the classifier code blueprint) we have spent most of our efforts and time in creating the ontology and extending the classifier for our use case. Especially the creation of the DCEL took a lot of time since there were no algorithms that we could use readily available. As such we are currently looking into making our implementation for the construction of a DCEL from a planar graph available via GeoTools.

6 Discussion

In section 1.2 we have formulated the following general question:

1. Is it possible to use a framework based on ontologies and Bayesian classifiers for data enrichment?

With the following sub questions:

2. How well are ontologies usable in a spatial context?
3. Can we use ontologies for hierarchical modelling of entities?
4. Can we use Bayesian classifiers to classify entities modelled within the ontological framework?

In our use case we have shown that we can use ontologies and Bayesian networks for data enrichment. To take a better look at the whole process we will discuss the three sub questions here.

Spatial entities can be modelled by adhering to the structure of an ontology: typical spatial things such as geometry, topology and location can be added as attributes of the entity and its relationships with other entities can be depicted by object properties. While ontologies offer us a structured way to model entities and to reason with them they also offer a lot of flexibility. This flexibility enables us to create a model that fits a particular point of view and can be easily changed to a different one by changing its sub ontologies. In light of the social constructs, vernacular geographies and fiat boundaries this is a good thing and helps us with the problems associated with these types of entities. But this freedom also offers the option to create horrific ontological models of poor quality with ease.

To counter this potential pitfall we can use two types of hierarchies which are very strong in spatial entities: partonomy and taxonomy. These relationships are crisp and usually well defined, both by scientists and non-scientists, and guide us in the creation of a good ontology. But while these hierarchies help creating an ontology they also limit us in the application as we have seen in the use case. We have made the Protégé plug-in BNGen to create Bayesian networks for classification in a semi-automated way using the modelled hierarchies as this is a more intuitive and structured way. In most cases this works very well when the entities that we want to classify adhere to the hierarchies, but once we want to classify something by using other entities that are on the same hierarchical level BNGen isn't able to help us with this because there are no explicit relationships modelled between them for BNGen to use. In the use case this can be seen with the greenish house that *could be* both a greenish house and a house of a different type at the same time. BNGen could be extended so the user could add relationships between entities that are on the same hierarchical level, but this will most likely make it less intuitive.

Another aspect of the creation of the Bayesian networks is that we might have to resort to staged classification and classify entities with a sub ontology first. This is caused by the incompatibility of Bayesian networks with dynamic situations where we don't know the exact type or size of the input beforehand. To solve this we have introduced summary nodes and propagate classifications from sub ontologies to higher level ontologies. This approach gives us satisfactory results, but the important thing is to find a good way to implement such summary nodes as how they should work totally depends on their context. The difference could range from simple minimum or maximum constraints to specific domain knowledge with complicated formulas from that domain.

Once we have implemented the full classification process and have classified our data we have a classified dataset in which we know the belief of every entity in the

data being the entity described in the ontology. This belief indicates how certain we are that it is such entity or how truthful our classification is. Especially with vernacular geographies we see the emergence of fiat boundaries and other effects that would make one say that a classification is "pretty good" or "almost correct". How good or correct is expressed in the belief that we have in the classification, which is visually communicated to the user with a heat map.

In our use case we have only modelled one specific spatial entity, or put more precisely, we have only modelled an area consisting of one type of entity: the greenish house. But how does our approach fare with other types of spatial entities that are not an area? For our use case we initially planned to find railway stations on maps as they are usually only indicated by a name or symbol and do not tell you what the extent of the station is. This type of entity would have been a set of different features that together make up a train station, like the railway and platforms. Unfortunately the data in the available dataset was so contradictory that we were unable to create an ontology that built upon the basic entities. But if the available data would have been of a higher quality and fidelity we would most likely have been able to classify these types of entities. So the approach would work for more precise entities such as railway stations and harbours or more vague entities such as wilderness or mountainous area, as long as our data gives us the possibility to create an ontology using its basic buildings blocks. But we have to keep in mind that there is a trade off between the effort required for high quality data capture and the capacity to infer higher level entities from data. For an optimal result one should have data that is good enough for the data enrichment process to produce satisfactory results, which differs for each and every problem or question.

7 Conclusion

In this thesis we have shown that it is possible to enrich a dataset using a framework of ontologies and Bayesian networks for classification. In this classification process we have exploited the hierarchical nature of spatial concepts for the creation of Bayesian networks by basing our Protégé plug-in BNGen on this and used staged classification in our classifier to deal with the static structure of Bayesian networks. However, in the use case we saw that not every relationship is hierarchical or clear in nature. As such we had to manually create a Bayesian network for these relationships after we could use this Bayesian network with our classifier along with Bayesian networks that we made with BNGen. Despite this drawback the approach seems to be promising, especially with our interpretation of an entity as a social construct. For vernacular geographies this could be useful as we can show how strong the belief is that something is the entity that we want to find according to some social group of people that made the ontology. With the use of heat maps one can show the vagueness of the vernacular geographies with fiat boundaries in our resulting dataset. This type of information will give the users also more insight in the concept they have created as they can see how much something matches instead of just knowing if it does or doesn't.

While we have mainly focussed on geographic entities in this thesis the approach would work in any area that works with entities, both crisp and vague, that need classification. Hierarchies are not required, but have to be clearly defined if we want to use them in BNGen. BNGen can also be extended for further work to include the option to set probabilities along with the state space of the vertices in the network. The code blueprint of the classifier also supplies a great extensibility as it can be adapted to any domain by the user.

References

[1] J. Bateman and S. Farrar. Spatial ontology baseline. *Collaborative Research Center for Spatial Cognition. I1-[OntoSpace] D*, 1, 2004.

[2] G. Bucci, V. Sandrucci, and E. Vicario. Ontologies and Bayesian Networks in Medical Diagnosis. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, pages 1–8, 2011.

[3] B. Buttenfield and E. Wolf. "The Road and the River Should Cross at the Bridge" Problem: Establishing Internal and Relative Topology in an MRDB. In *10th ICA Workshop on Generalisation and Multiple Representation, Moscow*, 2007.

[4] G. Câmara, A. M. V. Monteiro, J. A. Paiva, J. Gomes, and L. Velho. Towards a unified framework for spatial data models. *Journal of the Brazilian Computer Society*, 7:17 – 25, 2000.

[5] A. Cecconi. *Integration of cartographic generalization and multi-scale databases for enhanced web mapping*. PhD thesis, Universität Zürich, 2003.

[6] O. Chaudhry and W. A. Mackaness. Utilising partonomic information in the creation of hierarchical geographies. In *10th ICA Workshop on Generalisation and Multiple Representation. Moscow, Russia*, 2007.

[7] P. Costa, K. B. Laskey, and K. J. Laskey. PR-OWL: A Bayesian Ontology Language for the Semantic Web. In P. Costa, C. dAmato, N. Fanizzi, K. B. Laskey, K. J. Laskey, T. Lukasiewicz, M. Nickles, and M. Pool, editors, *Uncertainty Reasoning for the Semantic Web I*, volume 5327 of *Lecture Notes in Computer Science*, pages 88–107. Springer Berlin Heidelberg, 2008.

[8] M. De Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Computational geometry*. Springer, 2008.

[9] Z. Ding, Y. Peng, and R. Pan. A Bayesian Approach to Uncertainty Modeling in OWL Ontology. In *Proceedings of the International Conference on Advances in Intelligent Systems - Theory and Applications*, page 9, Luxembourg, November 2004.

[10] Z. Ding, Y. Peng, and R. Pan. BayesOWL: Uncertainty Modeling in Semantic Web Ontologies. In Zongmin Ma, editor, *Soft Computing in Ontologies and Semantic Web*, volume 204 of *Studies in Fuzziness and Soft Computing*, pages 3–29. Springer Berlin Heidelberg, 2006.

[11] D. Dubois and H. Prade. Possibility theory, probability theory and multiple-valued logics: A clarification. *Annals of Mathematics and Artificial Intelligence*, 32:35–66, 2001.

[12] S. Fenz. An ontology-based approach for constructing bayesian networks. *Data & Knowledge Engineering*, 73:73 – 88, 2012.

[13] F. Fonseca, M. Egenhofer, C. Davis, and G. Câmara. Semantic Granularity in Ontology-Driven Geographic Information Systems. *Annals of Mathematics and Artificial Intelligence*, 36:121–151, 2002.

[14] A. Frank. Tiers of ontology and consistency constraints in geographical information systems. *International Journal of Geographical Information Science*, 15(7):667–678, 2001.

[15] G. Frege. Die Grundlagen der Arithmetik. *Translated to English by JL Austin as The Foundations of Arithmetic, Oxford, 1959*, 1884.

[16] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian Network Classifiers. *Machine Learning*, 29(2-3):131–163, 1997.

[17] L. Van Der Gaag and M. Wessels. Selective evidence gathering for diagnostic belief networks. *AISB Quarterly*, 86:23–34, 1993.

[18] M. F. Goodchild. Geographical data modeling. *Computers & Geosciences*, 18(4):401 – 408, 1992.

[19] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider, and U. Sattler. OWL 2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):309 – 322, 2008.

[20] D. Gregory, R. Johnston, G. Pratt, M. Watts, and S. Whatmore. *The dictionary of human geography*. Wiley-Blackwell, 2009.

[21] W3C OWL Working Group. Owl 2 web ontology language document overview (second edition).

[22] E. M. Helsper and L. C. van der Gaag. A Case Study in Ontologies for Probabilistic Networks. In *Research and Development in Intelligent Systems XVIII: Proceedings of ES2001, the Twenty-first SGES International Conference on Knowledge Based Systems and Applied Artificial Intelligence, Cambridge, December 2001*, page 229. Springer, 2002.

[23] G. M. Jacquez, S. Maruca, and M-J. Fortin. From fields to objects: a review of geographic boundary analysis. *Journal of Geographical Systems*, 2(3):221–241, 2000.

[24] K. Janowicz. Observation-Driven Geo-Ontology Engineering. *Transactions in GIS*, 16(3):351–374, 2012.

[25] A.S. Larik and S. Haider. Efforts to blend ontology with Bayesian networks: An overview. In *Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on*, volume 2, pages V2–598–V2–602, 2010.

[26] T. Lukasiewicz and U. Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):291 – 308, 2008.

[27] P. Lüscher. *Characterising urban space from topographic databases: cartographic pattern recognition based on semantic modelling*. PhD thesis, Zürich University, 2011.

[28] P. Lüscher, D. Burghardt, and R. Weibel. Ontology-driven enrichment of spatial databases. In *10th ICA Workshop on Generalisation and Multiple Representation, Moscow*, 2007.

[29] P. Lüscher, D. Burghardt, and R. Weibel. Integrating ontological modelling and bayesian inference for pattern classification in topographic vector data. *Computers, Environment and Urban Systems*, 33(5):363 – 374, 2009.

[30] P. Lüscher, R. Weibel, and W.A. Mackaness. Where is the terraced house? on the use of ontologies for recognition of urban concepts in cartographic databases. *Headway in Spatial Data Handling*, pages 449–466, 2008.

[31] D. L. McGuinness M. K. Smith, C. Welty. OWL Web Ontology Language Guide, 2004.

[32] W. Mackaness and O. Chaudhry. Generalization and Symbolization. In Shashi Shekhar and Hui Xiong, editors, *Encyclopedia of GIS*, pages 330–339. Springer US, 2008.

[33] W. A. Mackaness and O. Z. Chaudhry. Automatic Classification of Retail Spaces from a Large Scale Topographic Database. *Transactions in GIS*, 15(3):291–307, 2011.

[34] M. Neun, R. Weibel, and D. Burghardt. Data enrichment for adaptive generalisation. In *ICA Workshop on Generalisation and Multiple Representation*, pages 20–21, 2004.

[35] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

[36] S. Sen and A. Krüger. Heuristics for Constructing Bayesian Network Based Geospatial Ontologies. In Robert Meersman and Zahir Tari, editors, *On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS*, volume 4803 of *Lecture Notes in Computer Science*, pages 953–970. Springer Berlin Heidelberg, 2007.

[37] B. Smith and A. Varzi. Fiat and bona fide boundaries. *Philosophy and phenomenological research*, 60(2):401–420, 2010.

[38] M.K. Thomson. *Dwelling on ontology-semantic reasoning over topographic maps*. PhD thesis, UCL (University College London), 2009.

[39] T. Waters and A. Evans. Tools for web-based GIS mapping of a fuzzy vernacular geography. In *Proceedings of the 7th International Conference on GeoComputation*. Citeseer, 2003.

[40] Y. Yang and J. Calmet. Ontobayes: An ontology-driven uncertainty model. In *Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on*, volume 1, pages 457–463, nov. 2005.