Encoding Strategies for Learning Geographical Concepts via Analogy

Matthew D. McLure & Kenneth D. Forbus

Qualitative Reasoning Group, Northwestern University,
2133 Sheridan Rd, Evanston, IL 60208 USA
mclure@u.northwestern.edu, forbus@northwestern.edu

Abstract

Creating systems that can work with people as apprentices, learning from them in natural ways, is a long-standing goal of AI. Being able to teach systems new concepts via sketching is an important step towards this goal. This paper explores the role of qualitative representations in learning geographic concepts indicated on 2D maps. We propose three principles for representation bias when learning via analogical generalization, and describe two dimensions of variation in qualitative encoding schemes for 2D maps. An experiment with multiple encoding schemes, using Freeciv, an open source strategy game, is described.

Introduction

Analogical learning over qualitative representations has been used to effectively learn spatial concepts (e.g. Lockwood et al 2008). A central problem in learning is how to choose (or construct) an appropriate representation scheme for learning new concepts. This must be done automatically, in order to support learning the broadest range of concepts. Analogical generalization can separate characteristic structure from irrelevant noise in the input, but this assumes that the qualitative representations (1) are rich enough to include the important information and (2) do not include so much noise that the matching process is derailed. This trade-off in analogical retrieval has been called the *Goldilocks problem* (Finlayson & Winston, 2005).

Previous work on learning spatial concepts from sketched input (e.g. Lovett, Dehghani and Forbus 2006; McLure, Friedman and Forbus 2010; Veselova 2003) relied on hand-tuned, fixed encoding schemes. Unfortunately, different concepts can require different encoding strategies. For example, on tasks involving comparison between simple closed shapes it is beneficial to attend to the edges that constitute each shape, because this is the level of abstraction at which a handful of entities exhibit a number of informative relationships. However, when encoding a scene with many disconnected objects, it is often useful to begin by attending to the relationships between the objects and ignoring their internal structure (Lovett and Forbus 2011). There is psychological evidence that even though people organize perceptual entities bottom up, they seem to attend to them top down (Hochstein and Ahissar 2002). Thus quickly determining what strategies might be reasonable for a new concept is an important problem.

This paper investigates what are the properties of good encoding strategies for analogical learning, and how can they be selected automatically based on the properties of a given example. We use a rich simulated environment, the open-source strategy game Freeciv. Freeciv uses a 2D map of Earth-like terrain, making it a useful domain for exploring learning of geographical concepts. The game

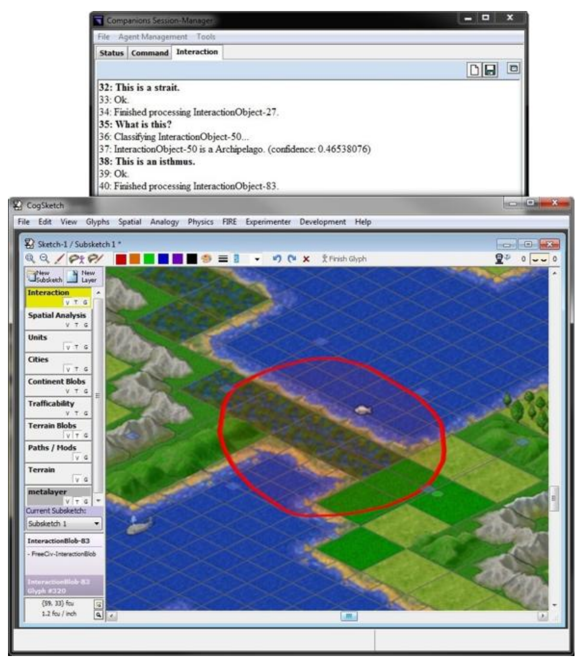

Figure 1. Examples are introduced to the system by circling them in the CogSketch/Freeciv interface.

McLure, M. & Forbus, K. (2012). Encoding Strategies for Learning Geographical Concepts via Analogy. Proceedings of the 26th International Workshop on Qualitative Reasoning. Los Angeles, CA.

world map interfaces with CogSketch (Forbus et al 2011), our sketch understanding system, to enable human trainers to introduce examples by circling them on the map, as illustrated in Figure 1. This fluid style of interaction provides a natural way to communicate spatial ideas.

In the machine learning community, learning bias is sometimes considered a combination of *representational bias*, i.e. a definition of the hypotheses space, and *procedural bias*, i.e. methods for searching the hypothesis space (Gordon and des Jardins 1995). Our goal is to find a representational bias that is synergistic with the procedural biases of analogical learning, and understand how this representational bias can be operationalized for geographical concepts in a 2D map.

We begin summarizing relevant background about sketch understanding and analogical processing. Then we discuss some proposed principles for representational bias for analogical learning. We discuss several encoding schemes and a simple discrimination tree for choosing between them. The effectiveness of this model on learning a set of six geographic concepts is evaluated by an experiment. Finally, the results are discussed, along with related and future work.

Background

We start by describing CogSketch and its interface with Freeciv, which provides a means of introducing perceptual examples to the learning system through a shared sketching interface. We then briefly discuss our models for analogical learning and classification since representational bias inherently interacts with procedural bias.

Perceptual Encoding Models

CogSketch

CogSketch (Forbus et al. 2011) is an open-domain sketch understanding system. It takes digital ink as input and produces structured, qualitative representations. The basic level of organization for ink in CogSketch is the *glyph*. A user may segment hand-drawn ink into glyphs, but glyphs are also generated by the system. CogSketch is capable of computing various spatial relationships between glyphs, including adjacency, relative position (e.g. 2D occlusion), topological relationships and relative size. CogSketch can also compute shape attributes of glyphs, such as roundness.

CogSketch can operate on glyphs to construct new entities on the fly. It can create glyphs from the intersection, difference, or union of two existing glyphs. It can also create new glyphs by intersecting visible glyphs with the drawing pane. CogSketch can decompose a glyph into a network of edges that represent either its ink or its medial-axis transform (2D skeleton). Just like at the glyph level, there is a collection of qualitative spatial relationships and attributes that can be automatically computed at the edge level, including various flavors of connectedness, relative length, concavity, axis alignment, and curvature. See (Lovett and Forbus 2011) for a full catalog of edge-level descriptors.

The organization of a glyph's ink into edges is based on intersections and discontinuities in the curvature of the ink, using a modification of the Curvature Scale Space corner detector (Mokhtarian and Suomela 1998). For edges in a medial-axis transform (MAT), we instead separate edges at the forks in the MAT and points where there is a qualitative change in the radius function, i.e. the distance between the points along the MAT and their respective closest points on the exterior of the glyph, i.e. their *generating points*. For example, in Figure 2, the section of the MAT corresponding to each finger of the hand is carved into three pieces as it extends outward from the palm: (1) a segment entering the base of the finger where the radius function is decreasing, (2) a segment where the radius function is approximately constant, and (3) a segment at the tip of the finger where the radius function is again decreasing. There are some extensions to the basic edge-level vocabulary for describing attributes of and relationships between MAT edges. MAT edges may be directed (oriented in the direction of decreasing radius function) or undirected (along segments where the radius function holds constant at a local minimum or maximum). `weaklyDirectedConnection` is an asymmetric relationship between a directed edge and an undirected one that connects to it, whereas `stronglyDirectedConnection` relates two directed edges that are connected head-to-tail. The

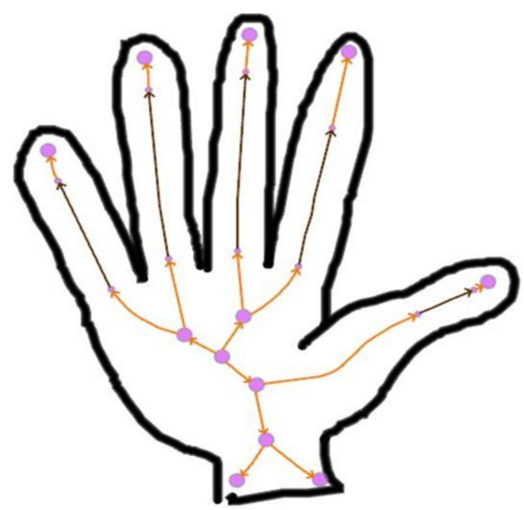

Figure 2. A user-drawn hand and its medial-axis transform (MAT) as generated by CogSketch. Edges are segmented at forks in the MAT as well as qualitative changes in the radius function. Edges in orange are segments with a decreasing radius function, whereas those in brown have a roughly constant radius function.

`sourceConnection` relation holds between two connected edges, neither of which is directed toward the other. Inversely, `sinkConnection` relates two connected edges, neither of which is directed away from the other.

The Freeciv/CogSketch Interface

CogSketch is capable of interfacing with the Freeciv world map by communicating through a Lisp-based Freeciv AI player (Houk 2004). A base layer is created in the sketch to render the terrain tiles of the map. Other layers in the sketch are populated with glyphs corresponding to entities from the Freeciv world, such as units and cities.

Additional layers are created for various types of *blob* representations. A blob is a region in the sketch that demarks a cluster of adjacent tiles belonging to some category. The most coarse-grained types of blobs are continent blobs, which divide the terrain into regions of land and water. At a finer granularity are trafficability blobs, which subdivide continent blobs into regions based on the degree of trafficability of tile types. There are three degrees of trafficability for land tiles and two for water tiles. At the finest granularity are terrain blobs, which subdivide trafficability blobs based on the literal type of the tile (e.g. forest, desert, mountain, ocean, lake). Importantly, blobs are an entirely ink-based representation of clusters of tiles in the Freeciv world; CogSketch treats them as domain-independent ink polygons (possibly with holes). Therefore the encoding and learning processes described in the rest of this paper are applicable to any 2D terrain for which blobs can be constructed.

At the top of the layer stack is the *interaction layer* where the user can draw additional glyphs by hand, called *interaction glyphs*. In the simulation discussed later, examples are introduced to the system by circling regions of the map on the interaction layer and providing a textual label. Figure 1 provides a snapshot of one such interaction. To avoid excessive computation on large maps, the glyphs that represent blobs are pruned down to the portion that is visible in the sketching pane at the time the stimulus is introduced. To maintain consistency during simulation, the same zoom (relative to the interaction glyph) and aspect ratio are used across all examples.

Analogical Models for Learning and Classification

This subsection describes the analogical models used to learn from examples and to classify them.

The Structure Mapping Engine

SME, the Structure Mapping Engine (Falkenhainer et al. 1989), is a domain-general computational model of Gentner's (1983) structure mapping theory of analogy and similarity. It takes two structured, relational cases as input: a *base* and a *target*. SME computes up to three *mappings* between the base and target. Mappings include a list of *correspondences* between entities and expressions in the base and target, candidate inferences which are suggested by the mapping, and a numerical *similarity score*. We normalize the similarity score by dividing it by the score computed for mapping the base with itself.

Structure mapping theory includes a *systematicity bias*, which states that higher-order relations (relations whose arguments are relations) have a more significant impact on similarity judgments than do lower-order relations, and lower-order relations have more impact than surface-level attributes. Structure mapping also requires that mappings are one-to-one: i.e., an entity or expression in the base can correspond with at most one entity or expression in the target, and vice-versa.

MAC/FAC

During learning and classification our system makes repeated use of MAC/FAC (Forbus et al. 1995), a domain-general computational model of similarity-based retrieval. It takes as input a *probe* case, which is again a structured, relational representation, along with a case library of other such representations. In a two-stage process, MAC/FAC retrieves up to three *remindings* from the case library that are similar to the probe. The first stage is an efficient, coarse filter that computes dot products between the content vectors of each case in the library and that of the probe. The three most similar cases from the first stage advance to the second, where SME is used to compute a mapping between each one and the probe. The best mapping, or up to three if they are all within 10% of the best, is returned as MAC/FAC's output. The returned remindings include the associated similarity scores and SME mappings.

Learning with SAGE

SAGE is a computational model of analogical generalization descended from SEQL (Kuehne et al. 2000), extended with probability (Halstead and Forbus 2005). The approach presented here uses SAGE for automatically clustering and merging positive examples into generalizations. Each user-provided category (label) is assigned a *generalization context*, where positive examples are assimilated incrementally and clustered into *generalizations* based on similarity. Each incoming example is first subjected to a *selection* phase, in which MAC/FAC is used to retrieve remindings from the generalization context given the incoming example as the probe. If the top reminding has a similarity score over the *similarity threshold*, then the new example is assimilated into it. That is, if the reminding is a generalization, the new example is merged into it, and if the reminding is a previous example, the old and new examples are combined to form a new generalization. Otherwise, the new example is added to the context as an ungeneralized example. The merge process updates the probabilities of each statement in a generalization, based on their frequency of occurrence in the examples in that generalization. Non-identical entities that correspond are replaced by new abstract entities, not logical variables.

As a generalization assimilates more examples, incidental features fade into low-probability and eventually disappear when below a probability cut-off, whereas common features retain high probability. When the generalization participates in future analogical matches, the probabilities of statements influence the matching process.

Classification

Our classification approach uses MAC/FAC to fetch remindings across the union of the SAGE generalization contexts for all learned concepts. The label associated with each returned reminding is treated as a vote for a concept, and each vote is weighted with the similarity score of the reminding. The system assumes that the concepts are mutually exclusive and thus assigns the highest-voted label to the example.

Encoding Strategies

Here we describe some new encoding strategies for learning spatial concepts via analogical generalization. We begin by describing a set of principles that we believe should guide the encoding process, thus defining (at least conceptually) a representational bias for the system. Next, we describe a set of encoding schemes designed using these principles. Finally, we describe a dynamic encoding strategy (referred to as the Selective Strategy) in which the system chooses an appropriate encoding scheme for a given example based on the proposed representational biases.

Representational Biases

To investigate appropriate representational bias for a learning system, it is useful to consider its procedural bias. The representational bias and procedural bias in a learning system may interact synergistically or adversely (Guyon and Elisseef 2003; Cardie 1993). In the present system, the procedural bias is fixed as we attempt to find a synergistic representational bias. We use background knowledge about the procedural bias of the analogical learner and past research on qualitative representations of space to propose a set of three principles that should lead to a beneficial representational bias for learning spatial concepts via analogy: sparsity, structure and locality.

Sparsity

In light of the Goldilocks problem, reasonably sparse representations are preferred to large ones. Representations that are too sparse will lack discriminatory power. Bloated representations can derail the analogical matching process, because they make good matches harder to find. In CogSketch, spatial representations that describe relationships between more entities tend to result in more facts, due to the combinatorial nature of relations, and the availability of multi-argument relations. Therefore, all else being equal, an encoding scheme that produces fewer entities is preferred to one that produces more entities. This rule breaks down for encoding schemes that produce fewer than two entities, as discussed next.

Structure

Because systematicity is a fundamental bias in SME, it is also a fundamental procedural bias in a learning system that uses SAGE and MAC/FAC. It follows that the learning system should perform better on input that includes a higher degree of structure (i.e. representations containing relations are preferred to those containing only attributes, and those containing higher order relations are preferred to those containing only low order relations). In the Selective Strategy below we operationalize the structure bias by seeking encoding schemes that operate on multiple blobs rather than just one. Attending to more than one entity will allow for at least some relations to be encoded, thereby avoiding structure-less cases (i.e., only containing attributes). There is an inherent trade-off between structure and sparsity.

Locality

When the user selects an item in the sketch and introduces it as an example, encoding should prioritize information about the items in the sketch that are near the selected item. Locality is already built into several of CogSketch's algorithms, e.g. positional relationships are only computed between glyphs that are adjacent. In our task, the selected interaction glyph is a closed shape overlaid on top of a map, so the region it encloses is considered the stimulus. At least one of the entities that participate in the representation should lie completely within the region of focus. Thus the lack of a containment relationship between the interaction glyph and at least one participating entity is taken as a signal that the current encoding strategy is not sufficiently localized.

Encoding Schemes

We define an *encoding scheme* as a partially ordered list of queries that are issued to CogSketch. The results of these queries are compiled into a structured, propositional description used as a case for analogical learning. The queries are partially ordered because some queries are constrained by the results of previous queries in the scheme. The following three scheme definitions are functional; they take a blob-level (continent, trafficability or terrain) as input, which determines the exact partial order of queries that is instantiated. Since there are three functional schemes and three blob levels, there are nine possible encoding schemes.

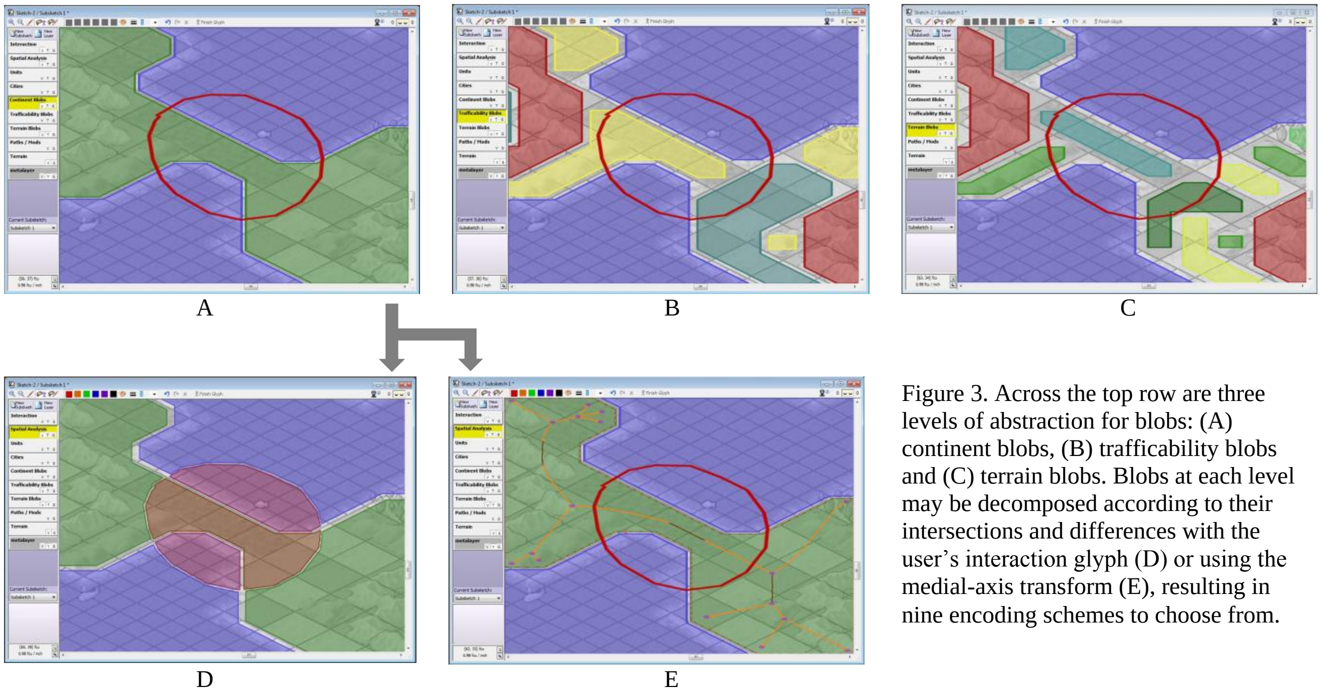

Figure 3. Across the top row are three levels of abstraction for blobs: (A) continent blobs, (B) trafficability blobs and (C) terrain blobs. Blobs at each level may be decomposed according to their intersections and differences with the user's interaction glyph (D) or using the medial-axis transform (E), resulting in nine encoding schemes to choose from.

Whole blobs

The Whole Blob encoding scheme first queries for every blob glyph at the given blob level that overlaps the interaction glyph, and it encodes each overlap as a binary relation. Next, all attributes of the overlapping blob glyphs are encoded, including the blob type (e.g. land/water, or forest/grassland/mountain/etc.) as well as glyph shape information such as the degree of roundness and elongation. Next all pairs of adjacent blob glyphs are encoded, where adjacency is determined using the Voronoi diagram. For triples that exhibit a transitive adjacency relationship (A is adjacent to B and B is adjacent to C), we check for 2-D occlusion (whether B occludes A from C) and encode it in the case. This captures a sense of between-ness. We also encode a relative size relationship for any pair of adjacent glyphs in which one is more than twice the size of the other. Finally, for every pair of adjacent glyphs, A and B, we encode any topological relationships that hold between A and the convex hull of B, as well as those between B and the convex hull of A. This captures when one blob is nested within another blob's concavity.

Severed blobs

As with the Whole Blobs scheme, the Severed Blob encoding scheme first queries for every blob glyph at the given blob-level that overlaps the interaction glyph, and the type information for each of these blobs in encoded. Every overlapping blob glyph is then decomposed into two pieces, one from the intersection between the blob glyph and the interaction glyph and another from the difference between the blob glyph and the interaction glyph, each of which forms a new glyph in the sketch. Figure 3(D) demonstrates how the intersections and differences between three blob glyphs and the interaction glyph result in seven new (severed) glyphs. The intersection and difference relationships are encoded. From this point forward, the queries closely imitate the Whole Blobs scheme, except that they operate on the severed glyphs. The shape attributes of the severed glyphs are encoded. Then their pairwise adjacency relationships between the glyphs are encoded, and used to constrain the subsequent encoding of occlusion, relative size, and convex-hull topological relationships.

Blob Skeletons

Like the other two encoding schemes, the Skeleton Blobs scheme begins by encoding which blob glyphs overlap with the interaction blob. Next it queries for a more strict type of overlap relationship: symmetric bisection. There is a symmetric bisection between two glyphs A and B if taking the difference between A and B results in at least two new glyphs *and* taking the difference between B and A also results in at least two new glyphs. In Figure 3(E), the land blob has a symmetric bisection relationship with the interaction glyph whereas neither water blob does. The symmetric bisections between blob glyphs and the interaction glyph are encoded, and any blob glyph for which a symmetric bisection holds is then decomposed using the medial-axis transform. The `hasMedialAxisEdge` relationship is encoded to link each

MAT edge to its parent glyph. To respect sparsity and locality, only the MAT edges that overlap the interaction glyph are included. All of the standard qualitative MAT edge attributes (length, curvature, axis alignment and directed vs. undirected) are encoded for these edges, plus any MAT edge relationships that hold for any pair of them. (e.g. `sinkConnection`, `weaklyDirectedConnection`).

Scheme Selection with a Decision Tree

The system has nine possible encoding schemes to choose from for a given stimulus. With the intuition that the spatial properties of the stimulus can inform which of these schemes is most appropriate for striking a balance between sparsity, structure and locality, the Selective Strategy was designed to dynamically choose an encoding scheme based on the region enclosed by the interaction glyph. To accomplish this, it uses the simple decision tree shown in Figure 3. First, a decision is made about which blob-level at which to encode. The continent level is preferred because it is most sparse, but the system only proceeds to encode at this level if the interaction glyph overlaps multiple blobs here. Recall that this is how we operationalize the structure bias. If this criterion is not met, the search moves to progressively finer-grain blob levels until one is found that does meet the criterion or no lower levels are available.

Once a blob level is chosen, the system chooses between the Whole Blob, Severed Blob, and Blob Skeleton schemes. If an entire blob is contained within the interaction glyph, we can assume that the contained blob(s) is/are the intended focal point(s), so we encode using the Whole Blob scheme to capture relevant local relationships with neighboring blobs while respecting sparsity by not introducing extra entities. Otherwise, if at least one of the blobs has a symmetric bisection relationship with the interaction glyph, then it is assumed that a subsection of the blob's MAT overlaps with the interaction glyph (serving the locality bias), and we pursue the Blob Skeleton scheme. This heuristic may be overly aggressive for spatial concepts outside of our dataset but we leave that for future work. The system chooses the Severed Blobs scheme as a last resort. This scheme guarantees that some entities will be local to the interaction glyph because it generates intersections between the interaction glyph and overlapping blobs. However, it typically results in the least sparse representations.

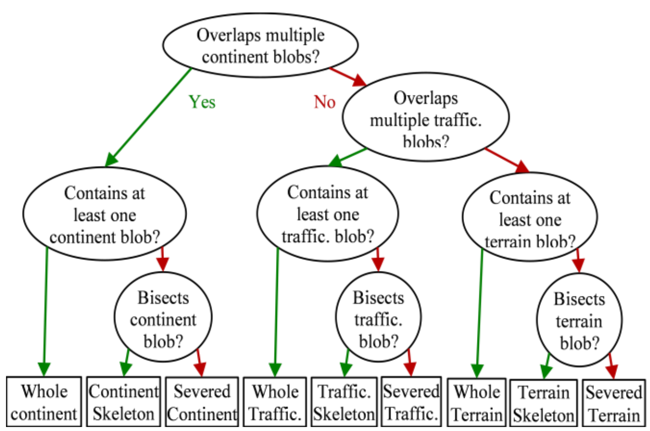

Figure 4. A decision tree for selecting an encoding scheme to use based on the properties of region circled by the user.

Experiment

We investigated the efficacy of the nine fixed encoding schemes and the Selective Strategy by evaluating their performance on a classification task. The learner was trained on a set of geographical concepts from the Freeciv world, and was then asked to classify a series of unlabeled examples. The dataset consisted of 60 examples spread evenly across 6 concepts: Isthmus, strait, peninsula, valley, island and archipelago. For each of the 10 encoding conditions (9 fixed encoding schemes plus the Selective Strategy), the learner was evaluated using a 10-fold cross-validation pattern where each fold included one example of each concept.

	Continents	Trafficability	Terrain
Whole	45%	65%	73.33%
Severed	61.67%	48.33%	40%
Skeleton	58.33%	40%	33.33%
Selective Strategy			76.67%

Table 1. Classification accuracies for the 9 baseline encoding schemes and the proposed Selective Strategy approach. The schemes shaded in gray are those that the Selective Strategy ended up selecting for at least one stimulus in our dataset.

Results

The accuracies for the 10 encoding strategies are shown in Table 1. The classification accuracy for the Selective Strategy (encoding scheme selection by decision tree) approach was 76.67%. Chance is at 16.67%. The Selective Strategy condition performed better than any of the individual schemes that it ended up selecting for any of the concepts in our dataset. A paired t-test over the 10 folds revealed that this difference was significant with $p < 0.1$. However, the selective strategy did not end up significantly outperforming the best-performing baseline, Whole Terrain Blobs, which demonstrated an accuracy of 73.33%.

We were interested in which pairs of concepts were most confusable for the top-performing schemes. For the Selective Strategy, confusion between isthmi and straits accounted for 62% of the misclassified examples – the most by far. The pair of concepts confused most by the Whole Terrain Blobs scheme was isthmi and peninsulas, which accounted for 25% of the misclassified examples. This scheme confused 7 distinct pairs of concepts, but it never confused straits with isthmi.

The decision tree for selecting an encoding scheme ended up dividing the examples by conceptual label; all peninsulas ended up being encoded with the Severed Continent Blob scheme, all isthmi and straits with the Continent Blob Skeleton scheme, all archipelagoes and islands with the Whole Continent Blobs scheme, and all valleys with the Whole Trafficability Blob scheme. This leaves open the possibility that the decision tree to decide the encoding scheme is responsible for the classification accuracy. The accuracy at chance using the decision tree alone to classify the examples directly is 67%. We established using a within-sample bootstrap that the Selective Strategy condition outperformed this hypothetical "chance given decision tree" condition within a 90% confidence interval, thus ruling out this possible explanation for the results.

Discussion

We proposed a set of principles to guide representational bias in a system that learns spatial concepts via analogy. Based on these principles, we designed a set of encoding schemes for attending to various types of entities at different levels of abstraction. We tested each of these encoding schemes on a classification task, along with a strategy that selects between the available schemes based on the spatial properties of a given stimulus. The Selective Strategy outperformed the baseline accuracies of the four encoding schemes that it ended up selecting, but it did not outperform the best-performing baseline, the Whole Terrain Blobs scheme. This is interesting because the best performing scheme was deemed by the Selective Strategy to be too fine a granularity to use on any of the concepts in our dataset, despite the fact that it performed well on them overall. This may mean that our encoding schemes are too sparse, since the most detailed scheme performed best.

The similar performance between the Selective Strategy and the Whole Terrain Blobs scheme may result from a property of analogical generalization. For example, isthmi and straits have nearly identical relational structure when encoded by any continent-based scheme. This pair of concepts was by far the most confusable for the Selective Strategy. The distinguishing features between these two concepts are the type attributes (water/land) of the blobs, rather than their spatial relationships. Thus classifying based on structural similarity alone leads to confusability between these concepts at the continent level, despite the fact that the important difference is consistently present in the input representations. Even though highly structured representations are still necessary for properly mapping and generalizing these examples, structural similarity alone may be insufficient for high-reliability classification. We have developed techniques for using near-misses to detect subtle distinguishing features during analogical learning (McLure, Friedman and Forbus 2010). Applying that technique to this dataset would likely benefit continent-based schemes (and subsequently the Selective Strategy). Interestingly, the Whole Terrain Blobs Scheme never confuses straits and isthmi because the land continent blobs are always decomposed into configurations of specific terrain types whereas the water continent blobs rarely are. Thus the distinction that separates this concept pair is highly structural for that scheme. Instead, misclassifications for the Whole Terrain Blobs scheme are more uniformly distributed across concept pairs, and likely would not benefit as much from near-misses. We leave testing this hypothesis for future work.

Related Work

Qualitative representations of space have been used for learning spatial concepts in a number of tasks (Lovett, Dehghani and Forbus 2006; McLure, Friedman and Forbus 2010; Veselova 2003). Lovett et al. and McLure et al. used the same analogical models to learn from structured, qualitative representations of sketched input. In both cases, encoding schemes were held constant across the examples. Veselova created a system to learn and recognize qualitative descriptions of sketched symbols. The qualitative relationships and attributes were assigned adjustable relevance scores, with relevance being at least partially determined by global, psychologically inspired properties of the sketch. Veselova's approach focused exclusively on edge-level representations, whereas we use multiple levels of abstraction.

Lovett and Forbus (2011) model perceptual organization (separation into entities and relationships between them at different levels of abstraction) as a bottom-up process, and encoding as a top-down process, by which results of comparison at higher levels of abstraction guide comparisons at lower levels. In our case, blobs are likewise organized from the bottom-up, as terrain tiles are grouped into progressively more abstract clusters, and the encoding process begins at the top level and progresses downward as necessary.

Medial-axis transforms, while a widely used geometric technique for generating representative skeletons for closed shapes, often result in "hairy" skeletons for noisy contours as with any hand-drawn ink. Pruning methods for MATs have been an active research topic in the object recognition community for some time. MATs in CogSketch are pruned using a technique inspired by Bai, Latecki and Liu (2007), who use the exterior edge-decomposition of a shape to prune its MAT. They achieve impressive results by pruning any MAT branch whose generating points lie along the same exterior edge. While their exterior contours are segmented using Discrete Curvature Evolution, CogSketch uses a novel algorithm based on Curvature Scale Space methods. The pruning constraint is also loosened slightly; MAT edges are not pruned if the generating points lie on two different halves of the same exterior edge.

Our qualitative representations for MATs were heavily influenced by shock graphs (Siddiqi et al. 1993). Shock graphs also carve a MAT into segments based on qualitative changes in the relationship between medial-axis points and their generating points. Siddiqi et al. proposed a custom algorithm for matching shock graphs, whereas we generate a structured, propositional representation for input into SME. One key difference is that shock graphs distinguished between segments that formed as a result of zero-points in the derivative of the radius function based on whether they were located at a local maximum, a local minimum, or an extended constant region. In all of these cases, we create an edge to delimit an extended constant region in the radius function, no matter how short the resulting edge is. Preliminary trials revealed that this approach was more compatible with SME's one-to-one mapping constraint.

The idea of using a selection process to guide encoding towards relevant features has precedents. The machine learning community has been researching feature selection for some time (Guyon and Elisseeff 2003). In fact, a decision tree approach was explored early on (Cardie 1993). These endeavors have mostly dealt with feature-vector based learning systems. We believe we are the first to attempt something similar to feature selection for analogical learning in a spatial domain.

Future Work

As mentioned above, we may be able to improve our results by incorporating near-misses into the learning and classification processes. However, many geographical concepts combine spatial and functional aspects, i.e. hubs in transportation networks and formations in military operations. Our analogical models should be able to work with such information easily. More difficult is using metaknowledge to do fine-grained tuning of encoding strategies. The probabilistic generalizations produced by SAGE might be used to reveal which encoding queries add noise to the representations. Similarly, a lack of discriminability between particular concepts may suggest when an encoding scheme must be elaborated or adjusted at a finer granularity. We may also find that encoding schemes can be debugged via user interactions.

Acknowledgments

This work was funded by the Air Force Office of Scientific Research.

References

Bai, X., Latecki, L.J., and Liu, W. (2007). Skeleton Pruning by Contour Partitioning with Discrete Curve Evolution. *IEEE Trans. Pattern Analysis and Machine Intelligence, 29*(3): 449-462.

Cardie, C. (1993). Using Decision Trees to Improve Case-Based Learning. *Proceedings of the Tenth International Conference on Machine Learning*: 25-32.

Falkenhainer, B., Forbus, K. and Gentner, D. (1989). The Structure Mapping Engine: Algorithm and examples. *Artificial Intelligence, 41*: 1-63.

Finlayson, M.A. & Winston, P.H. Intermediate Features and Informational-level Constraint on Analogical Retrieval, in *Proceedings of the 27th Annual Meeting of the Cognitive Science Society*

Forbus, K. D., Gentner, D. and Law, K. (1995). MAC/FAC: A model of similarity-based retrieval. *Cognitive Science 19*(2): 141-205.

Forbus, K., Usher, J., Lovett, A., and Wetzel, J. (2011). CogSketch: Sketch understanding for Cognitive Science Research and for Education. *Topics in Cognitive Science*. pp 1-19.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science 7*(2): 155-170.

Gordon, D. F. and des Jardins, M. (1995). Evaluation and selection of biases in machine learning. *Machine Learning, 20*(1/2): 5-22.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research, 3*: 1157-1182.

Halstead, D. and Forbus, K. (2005). Transforming between propositions and features: Bridging the gap. *In Proceedings of AAA-05.*

Hochstein, S. and Ahissar, M. (2002). View from the top: hierarchies and reverse hierarchies in the visual system. *Neuron 36*(5): 791-804.

Houk, P. (2004). A Strategic Game Playing Agent for FreeCiv. Technical Report NWU-CS-04-29, Northwestern University, Computer Science Department. Evanston, IL.

Kuehne, S., Forbus, K., Gentner, D. and Quinn, B. (2000). SEQL: Category learning as progressive abstraction using structure mapping. *Proceedings of CogSci 2000*.

Lockwood, K., Lovett, A., and Forbus, K. (2008). Automatic Classification of Containment and Support Spatial Relations in English and Dutch. In the *Proceedings of Spatial Cognition.*

Lovett, A., Dehghani, M., and Forbus, K. (2006). Efficient Learning of Qualitative Descriptions for Sketch Recognition. *Proceedings of the 20th International Qualitative Reasoning Workshop.*

Lovett, A. and Forbus, K. (2011). Organizing and representing space for visual problem-solving. *Proceedings of QR '11*.

McLure, M., Friedman, S., and Forbus, K. (2010). Learning concepts from sketches via analogical generalization and near-misses. *Proceedings of CogSci 2010.*

Mokhtarian, F., & Suomela, R. (1998). Robust image corner detection through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(12): 1376-1381.

Siddiqi, K., Shokoufandeh, A., Dickinson, S. J., and Zucker, S. W. Shock graphs and shape matching. *International Journal of Computer Vision.*

Veselova, O. 2003. Perceptually based learning of shape descriptions. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA.