Generating Explanatory Captions for Information Graphics

Vibhu O. Mittal* **Steven Roth†** **Johanna D. Moore*** **Joe Mattis†** **Giuseppe Carenini***

*Learning Research & Dev. Center and
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15260

†Robotics Institute
School of Computer Science
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract

Graphical presentations can be used to communicate information in relational data sets succinctly and effectively. However, novel graphical presentations about numerous attributes and their relationships are often difficult to understand completely until explained. Automatically generated graphical presentations must therefore either be limited to simple, conventional ones, or risk incomprehensibility. One way of alleviating this problem is to design graphical presentation systems that can work in conjunction with a natural language generator to produce "explanatory captions." This paper presents three strategies for generating explanatory captions to accompany information graphics based on: (1) a representation of the structure of the graphical presentation (2) a framework for identifying the perceptual complexity of graphical elements, and (3) the structure of the data expressed in the graphic. We describe an implemented system and illustrate how it is used to generate explanatory captions for a range of graphics from a data set about real estate transactions in Pittsburgh.

1 Introduction

Graphical presentations can be an effective method for succinctly communicating information about multiple, diverse data attributes and their interrelationships. A number of research groups are attempting to develop systems that automatically generate such presentations [Casner, 1991; Mackinlay, 1986; Roth *et al.*, 1994]. When a display includes only a small number of data attributes or only makes use of conventional graphical styles (e.g., spreadsheet graphics), it is easy for a viewer to understand how to interpret it. However, one of the main goals for automatic presentation systems is to allow users to see complex relationships between domains and perform problem-solving tasks (e.g., summarizing, finding correlations or groupings, and analyzing trends in data) that involve many data attributes at the same time. These visualizations are often both novel and complex. They can only be fully effective for supporting such analysis tasks if accompanied by explanations designed to enable users to understand how the graphics express the information they contain. Furthermore, studies (e.g., [Nugent, 1983]) have shown that the presentation of captions with pictures can significantly improve both recall and comprehension, compared to either pictures or captions alone.

In this paper, we propose a framework for automatically generating explanatory captions to accompany graphical presentations of combinations of diverse data sets. The graphical displays are designed by an automatic presentation system, SAGE [Roth *et al.*, 1994], and are often complex for several reasons. First, they typically display many data attributes at once. The mapping of many different data attributes to multiple graphical objects can be difficult to determine from the graphics alone. Second, integrating multiple data attributes in a display requires designing graphics that are unfamiliar to most users accustomed to simple spreadsheet displays of individual data attributes. While these integrated displays can be very useful once they are explained, it is difficult to understand them completely without accompanying explanations. Finally, the nature of the data with which we are concerned is inherently abstract and does not have an obvious or natural visual representation. Unlike depictions of real world objects or processes (e.g., radios [Feiner and McKeown, 1993], coffee makers [Wahlster *et al.*, 1993], network diagrams [Marks, 1991]) and visualizations of scientific data (e.g., weather, medical images), visualizations of abstract information lack an obvious geometric analog.

As an example of the type of data we are concerned with, consider the graphic shown in Fig. 1. This is a SAGE generated version of the famous graphic drawn by Minard in 1861 depicting Napoleon's march of 1812 [Roth *et al.*, 1994]. The graphic relates many different variables: position (latitude and longitude), troop size, direction of movement, temperature, and dates and locations of battles. Unless one has seen this graphic (or a very similar one) before, it cannot be understood and used to its fullest extent.

Consider how the following human-generated caption for the graphic in Fig. 1 explains the picture as well as the underlying data:

> *This graphic shows march segments and battles from Napoleon's 1812 campaign. The map shows the relation between the geographic locations, temperature and number of troops for each segment. Each line shows the start and end locations for the march segment. Its color*[1] *shows the tem-*

This work was supported by grant number DAA-1593K0005 from the Advanced Research Projects Agency (ARPA).

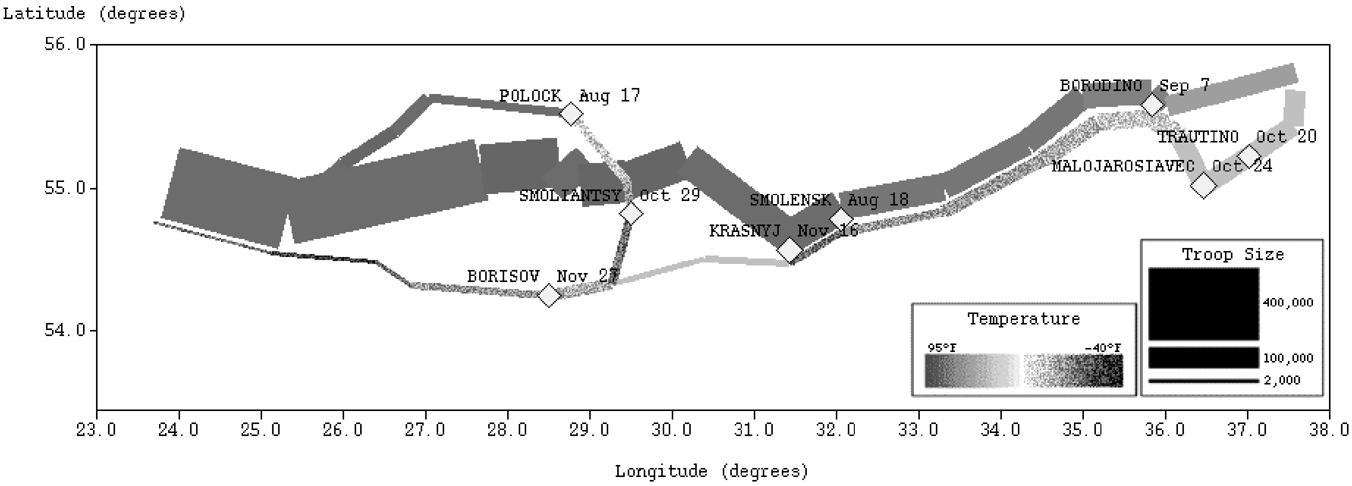


Figure 1: A SAGE generated version of the well known Minard graphic.[1]

perature, and the thickness shows the number of troops. The temperature was about 100 degrees for the initial segments in the west (the wide, dark red lines on the left), about 60 degrees in later segments in the east (the narrower, light red lines on the right) and about -40 degrees in the last segments, also in the west (the narrowest, dark blue lines on the left). The number of troops was 400,000 in the earliest segments, 100,000 in the later segments, and 10,000 in the last segments. The city and date of each battle is shown by the labels of a yellow diamond, which shows the battle's location.

With this caption, this complex, integrated graphic can be fully appreciated for the amount of information it conveys so succinctly and clearly.

Although several projects have focused on the question of how such intelligent graphical presentations can be automatically generated [Casner, 1991; Mackinlay, 1986; Roth and Hefley, 1993], they have not addressed the problem of generating the accompanying textual explanations. In this paper, we describe an implemented system designed to generate explanatory captions by integrating two robust systems: the SAGE intelligent graphics presentation system [Roth *et al.*, 1994], and a natural language generation framework consisting of a text planner [Moore and Paris, 1993] and a sentence realizer [Elhadad, 1992]. This system generates explanatory captions for complex graphical presentations that integrate multiple data attributes by: (1) aligning sequences of multiple charts and tables, (2) composing graphical objects (e.g., points, bars, lines, gauges, text strings) within each chart, and (3) using multiple graphical properties of each object to encode data attributes (e.g., shape, color, size, line thickness, etc.). This is an important application domain, since these types of graphics represent a significant portion of data that must be presented for many business applications [Schmid, 1983].

The system selects data to be presented in the caption by taking into account both the characteristics of the data (e.g., the range and type of attributes, as well as the relationships among them) and the way data objects and attributes have been mapped to graphical objects in the presentation generated by SAGE. The system organizes its explanations using one of three explanation strategies and determines how much to say about any individual data to graphic mapping based on heuristics about the complexity of understanding that particular type of mapping.

2 Requirements for Generating Explanatory Captions

Previous efforts in intelligent multimedia presentation have focused on coordinating NL and graphical depictions of real world devices (e.g., military radios [Feiner and McKeown, 1993], coffee makers [Wahlster *et al.*, 1993]) for generating instructions about their repair or proper use. In these projects, researchers have tackled problems such as the apportionment of content to media and generating cross references. Our work differs from previous efforts in two ways. First is the type of data that our system deals with. We are concerned with presentations of abstract or relational information (e.g., census reports, logistics data)–data that does not have an obvious or familiar analogy in the graphical medium. Second, although our long term goal is in generating coordinated multimedia explanations using information graphics and natural language, our current focus is generating explanatory captions. The current system presents all of the data graphically, and then generates captions based on the data as well as the perceptual complexity of the graphic. Therefore, the graphical display itself is the object of explanation.

Generating textual captions for visualizations of abstract or relational information requires the following:

- a representation of the *syntax* of graphical displays, that is, the structural, spatial and other relations among graphical objects and their properties.
- a representation of the *semantics* of graphical displays, that is, the way they map from data objects and their attributes to graphical ones.
- a mechanism for determining which aspects of graphical displays must be explained based on their perceptual complexity or the complexity of the data attributes they express.

[1] SAGE generates color graphics (see [Roth *et al.*, 1994]), but due to printing limitations, greyscale versions are shown here.

- strategies for generating coherent multi-sentential captions.

We describe the representations and reasoning strategies we have designed to meet each of these criteria in the following sections.

3 SAGE Overview

SAGE is a knowledge-based presentation system that designs graphical displays of combinations of diverse information (e.g., quantitative, relational, temporal, hierarchical, categorical, geographic) [Roth *et al.*, 1994]. The inputs to SAGE include: (1) sets of data, (2) a characterization of the properties of the data that are relevant to graphic design, (3) a characterization of the tasks that the graphics should be designed to support (e.g., determining the correlation among variables; finding subsets of data with extreme values for some attributes and then looking up values of other attributes for these; detecting differences between pairs of attributes, etc.), and (4) an optional set of design specifications, expressing a user's preferences for visualizing the data set.

SAGE's output consists of one or more coordinated sets of 2D information graphics that use a variety of graphical techniques to integrate multiple data attributes in a single display. SAGE integrates multiple attributes in three ways. First, it represents them as different properties of the same set of graphical objects (e.g., the thickness and shade of line segments in Fig. 1). Second, it assembles graphical objects into groups that function as units to express data (e.g., the pair of text strings associated with the diamond shaped marks at different positions on the line segments in Fig. 1). Third, it coordinates multiple charts and tables by aligning them with respect to a common axis (e.g., the aligned charts in Figs. 2 and 7).

SAGE's representation serves three functions in explanation generation. It helps define what a viewer must understand about a graphic in order to obtain useful information from it. It does this by defining the elements of a graphic and the functions they serve as they come together to express facts (i.e., how they map to data). The representation also describes the structure of both graphics and the data they present so that they can be explained coherently. Finally, the representation helps derive judgments of *complexity* for specifying graphical elements needing textual explanation. To understand these three functions, we briefly review the representation.

First, SAGE has knowledge of the characteristics of data relevant to graphic design [Roth and Mattis, 1990; Roth and Hefley, 1993], including knowledge of data types and scales of measurement (e.g., quantitative, interval, ordinal, or nominal data sets), structural relationships among data (e.g., the relation between the end-points of ranges or between the two coordinates of a 2D geographic location), and the functional dependencies among attributes in database relations (e.g., one:one, one:many, many:many, etc.).

Second, SAGE has a library of graphical techniques, knowledge of the appropriateness of the techniques for different data and tasks, and design knowledge for assembling these techniques into composites that can integrate information in a single display. SAGE uses this graphic design knowledge together with the data characterization knowledge to generate displays of information.

The portion of SAGE's knowledge base that is most relevant for generating explanatory captions is its graphical syntax and semantics. The syntax includes a definition of the graphical constituents that convey information (e.g., graphical objects called *graphemes*, their properties, the frames of reference that enable their properties to be interpreted/translated back to data values), and the ways graphemes can be combined to form composites that integrate multiple data attributes. The syntactic structure of a graphical display, like the linguistic structure of text, can provide guidance for creating structurally coherent explanations.

The representation of the *semantics* of graphics conveys the way data is mapped to the syntactic elements of displays. It also provides guidance for organizing explanatory captions by grouping graphical elements that express data attributes that form a coherent group. The data characterization provides information about the structure of the data and therefore also influences the structure of the explanation.

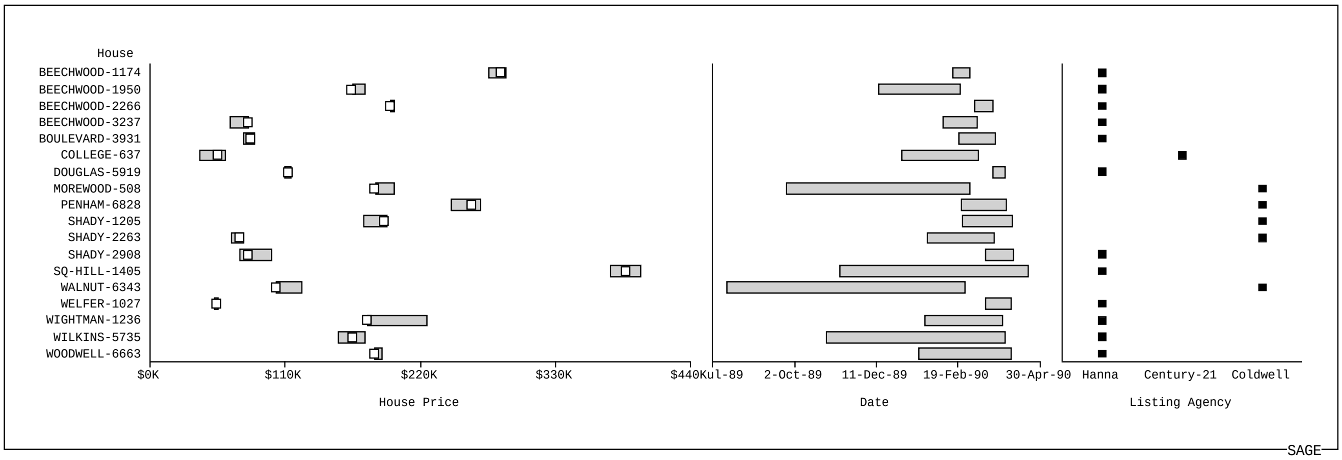
4 Understanding Elements of Graphical Presentations

SAGE's graphics convey assertions about the world that have been represented as tuples or facts in a relational database. In the example data set used here, the assertions are facts from a relational database of house sales (e.g., street address, date placed on market, date sold, asking and selling prices, agency estimate of selling price, listing agency, number of rooms, etc.). These are expressed by mapping them to graphical assertions, which are collections of graphemes and their properties.

Understanding a graphic is a process of determining how data attributes are related to graphical techniques (e.g., color, shape, horizontal and vertical position). This mapping is usually given by axes or keys which are called encoders (because they encode data values into graphical values). Encoders provide a frame of reference so that one can convert between values of graphic techniques (e.g., blue, horizontal pixel location 120) and data values (e.g., Century-21, $30,000). Knowing how a data attribute is expressed also requires knowing the graphical object (i.e., the grapheme) that is mapped to it. For example, in Fig. 2, it is not only necessary to know that prices are expressed with respect to the horizontal axis but also that each bar grapheme expresses the asking and selling prices, while the square grapheme expresses the agency estimate. Even more precisely, we need to know the properties of these graphemes that are mapped to the attributes: the horizontal position of the left and right edges of each bar convey selling and asking prices, while the horizontal position of the geometric center of each square conveys agency estimate.

Understanding a graphic may also require determining which graphemes and properties function together as units to express semantically coherent groups of attributes. For example, in Fig. 5, the shade, vertical position and horizontal positions of the edges of each bar combine with the mark graphemes to convey five data attributes.

To summarize, SAGE's representation defines components of graphics that must be understood for one to interpret how a graphic conveys information. In order to understand how a graphical technique is used, one must locate and understand the encoders that are frames of reference for converting



(1) These three charts show information about houses from data set PGH-23. (2) The Y-axis identifies the houses in the three charts. (3) In the left chart, house prices are shown by the X-axis. (4) The house's selling price is shown by the left edge of a bar, (5) whereas the house's asking price is shown by the right edge of the bar. (6) The horizontal position of the square mark shows the house's agency estimate. (7) For example, the asking price of 1950 Beechwood is $175K, (8) its selling price is $165K, and (9) the agency estimate is $164K. (10) In the middle chart, the house's date on the market is shown by the left edge of a bar, whereas (11) date sold is shown by the right edge of the bar. (12) The right chart shows the house's listing agency.

Figure 2: Graphic with caption generated using strategy 1.

graphical values for that technique into data values. One must also identify the graphemes and their properties that represent particular data attributes with respect to the encoder techniques, and the clusters of graphemes that come together as units to express multiple attributes.

5 Graphical Complexity: The Need for Clarification

While the representation identifies all the elements that one must understand to interpret a graphic, generating explanations requires focusing only on those elements that are not apparent. Indeed, an explanation that includes all the elements needed to understand a graphic would be extremely verbose. Therefore it is necessary for SAGE to identify those elements that might be difficult for a viewer to understand in answering the question: What data attributes are shown and how are they expressed in the graphic?

There are five types of complexity that can make it difficult for a user to answer this question and they correspond to graphical elements discussed previously.

• **Encoding technique complexity:** One factor that determines the difficulty of understanding encoding techniques is the number of dimensions involved. For example, in Fig. 1, saturation and color[2] are combined in a single encoding technique to express temperature. Dark red indicates 100 degrees and dark blue indicates -40 degrees. As the color gets paler (less saturated) it indicates a less extreme temperature. For example, pale red (pink) indicates 65 degrees, while pale blue indicates -5 degrees. White indicates a special transition point (e.g., 32 degrees).

Another example of technique-encoder complexity is the use of truncated scales for quantitative attributes. For example, the X-axis in Figure 3 does not have zero origins so the data can be distributed across a wider area. Therefore, it is erroneous to conclude that a point that is twice the distance from the origin as another point encodes a value twice as large as the first one. This is a common occurrence in charts so it is unlikely to be misinterpreted. However, this error can occur in less familiar techniques (e.g., the area of circles, etc.) as well.

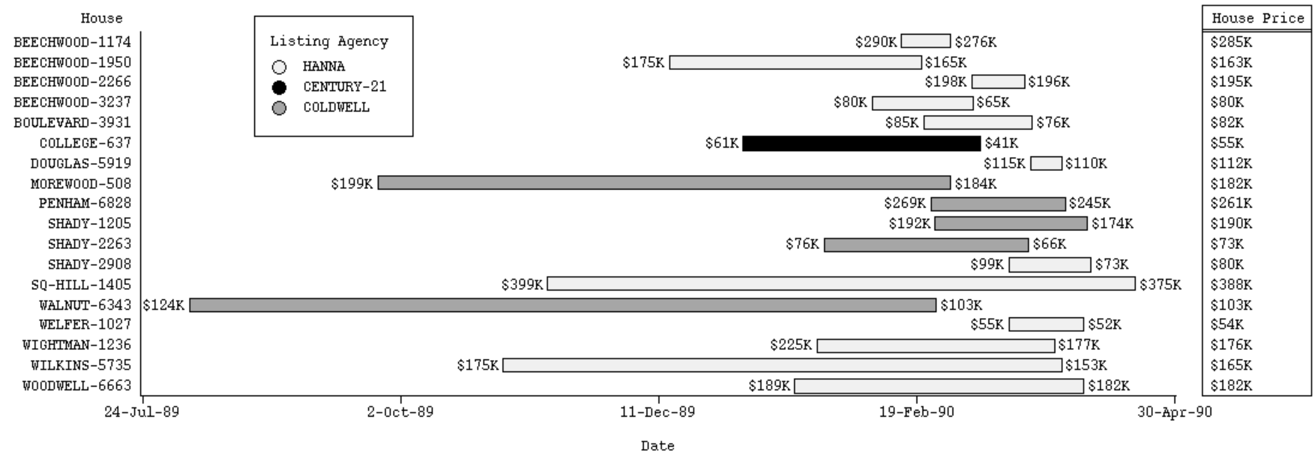
• **Grapheme complexity:** Although the technique and encoder may be simple, it is possible that a grapheme that uses that technique and encoder is geometrically complex, and so it is difficult to determine how it should be interpreted with respect to the encoder/technique. For example, while the axis in the middle chart of Fig. 2 is simple to interpret, the interval bars are complex if one has never seen them. Unlike the simple marks (squares) in the rightmost chart, the parts of the bar that map to the axis are the left and right edges. Similar problems occur for other types of graphemes (like the lines in Fig. 1).

• **Multiple grapheme properties:** A user's ability to identify the mapping of even simple techniques can be hindered when multiple properties of a grapheme are used at one time (e.g., shape, color and size of a point in a chart). For instance, it may not be clear that both the left and right edges of the horizontal interval bars in Fig. 2 map to different domains. Or, that two similar graphemes may map to very different domains: for instance, in Fig. 1, the two text graphemes (the labels next to the diamonds) map to battle-sites and dates, respectively. It is also possible that some properties can be overlooked because others are more salient. For instance, if all the houses in the data set in Fig. 5 were listed by the same agency, then all the interval bars would be the same shade. In such a case, the fact that the shade of the bar is used to communicate a data value could be overlooked.

• **Multiple graphemes:** When multiple graphemes occur in a space,[3] they can be confusing at first until their relation

[2]In this greyscale version, though the colors cannot be seen, the variation in saturation can be observed.

[3]Spaces are groupings of graphical elements that are positioned



This chart and table show information about house sales from data set PGH-23. The Y-axis identifies the houses in the two spaces. In the chart, dates are shown along the X-axis. The house's date on the market is shown by the left edge of a bar, whereas the house's date sold is shown by the right edge of the bar. The shade indicates the listing agency. The label to the left of a bar indicates the asking price of a house, whereas the label to the right of a bar indicates the selling price. For example, the asking price of 6343 Walnut is $124K, its selling price is $103K, its date on the market is August 10th and the date sold is Feb 20th. The table shows the agency estimate.

Figure 3: Caption for an alternative presentation of the dataset used in Figure 2.

to each other is understood by the viewer. For example, in the leftmost chart of Fig. 2, one must recognize the relation between the square and the bars, i.e., that the horizontal positions of the squares and the edges of the bars are all relevant (and related). Our current framework categorizes multiple graphemes to be either *cooperating*, or *interfering*, based on their position and effect on the user. For instance, the mark and the horizontal bar in Fig. 2 are considered interfering, whereas the labels and the mark in Fig. 1 are considered cooperating.

- **Complex alignments:** As illustrated in Figs. 2, 3, and 7, alignment can be a useful technique for supporting comparisons, rapid lookups for many attributes of the same object, and for maintaining consistent scales. Whenever an alignment of multiple charts and/or tables occurs, all but one become separated from the axis label and the relation between the aligned axis and all the charts may not be obvious.

The primary function of an explanatory caption is to make it easier for a user to determine the data that the graphic contains and how the data is expressed. Understanding the latter requires identifying the property of the grapheme (e.g., left edge of a bar) and its relation to the encoding technique (e.g., position along an axis). Therefore, for each data attribute, an explanation must describe those elements of the picture that are complex. The complexity assessment module identifies the graphical elements that require clarification in the caption. For example, the result of the complexity assessment module for the graphic in Fig. 2 is shown in Fig. 4 (the "i" after "multiple graphemes" is to indicate that these are *interfering*).

Data attribute	Graphical Element	Complexity Type
house	axis	alignment complexity
asking price	grapheme, property	complex grapheme, multiple properties, multiple graphemes (i)
selling price	grapheme, property	complex grapheme, multiple properties, multiple graphemes (i)
agency estimate	grapheme	multiple graphemes (i)
date on market	grapheme, property	complex grapheme
date sold	grapheme, property	complex grapheme
listing agency		

Figure 4: Complexity assessment for Fig. 2.

6 Generating Explanatory Captions

To generate explanatory captions, we use a natural language generation system consisting of a text planner [Moore and Paris, 1993] and a sentence realizer [Elhadad, 1992]. The system's knowledge about how to produce explanatory captions is encoded into plan operators and control strategies for the text planner. The operators can be viewed as recipes for achieving a given explanatory goal. Two of the plan operators used by the system are shown in Fig. 6. Each operator specifies the constraints under which the steps in the operator can achieve the specified effect. Constraints refer to knowledge sources to find information to include in the explanation and to check the appropriateness of a given strategy. They may refer to the SAGE representation of a picture, the complexity information, the user model, and the context created by the dialogue (including the evolving text plan.) Here, we confine our discussion of the text planner to how it is used in generating explanatory captions. For a more complete discussion of the text planner, see [Moore, 1995; Moore and Paris, 1993].

To generate a caption, the system passes a goal of the form (`DESCRIBE (GRAPHICAL-MAPPINGS (PICTURE`

according to a single layout discipline, e.g., charts, maps, networks, tables, etc.

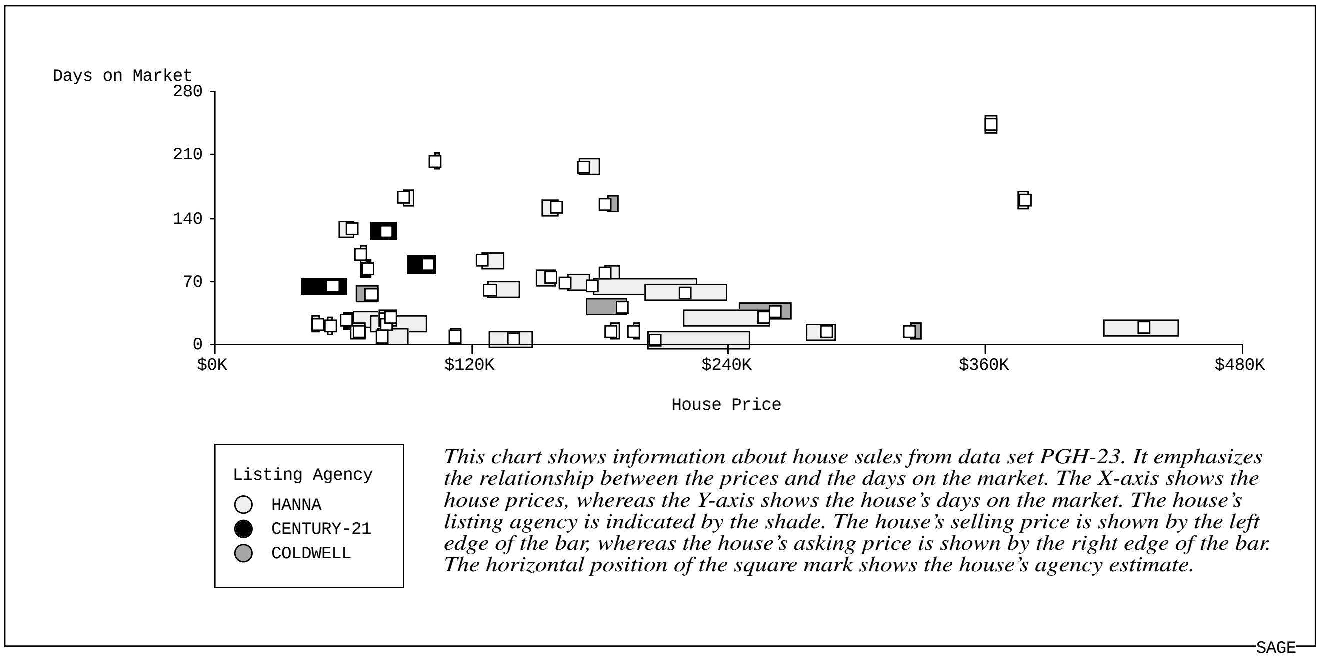


Figure 5: Graphic with caption generated using strategy 2.

?PICTURE)))[4] to the text planner. The planner then finds all operators capable of achieving this goal in the current context. Once a strategy is selected, it may in turn post subgoals for the planner to refine. For example, the operator shown in Fig. 6 posts several subgoals depending on the number of bindings for the variable ?space in the FORALL construct. Planning continues by refining subgoals in this fashion until the entire plan is refined into primitive actions that can be directly executed (in this case, speech acts such as INFORM). Once a text plan is completed, it is recorded in the dialogue history, and passed to the realization component. For each speech act in the text plan, the realization component performs lexical choice, constructs appropriate referring expressions using the algorithm proposed by Dale (1992), and selects passive vs. active forms to ensure local coherence based on the centering framework of Grosz *et al.* (1995). The resulting lexico-syntactic structures are then aggregated into complex sentences and passed to the syntactic realization grammar of the FUF system [Elhadad, 1992], which then generates the English text.

```
(define-text-plan-operator
  :name generate-caption-strategy-1
  :effect (describe (graphical-mappings (picture ?picture)))
  :constraints
    ((depicts-dataset ?picture ?dataset)
     (independent-domain ?dataset ?indep)
     (anchor ?indep ?picture)
     (all-spaces ?picture ?spaces)
     (> (number ?spaces) 1))
  :steps ((identify (depicts-indep ?picture ?indep ?dataset))
          (describe (anchor-mapping ?picture ?indep))
          (FOR-ALL ?space in ?spaces
               (describe (anchored-graphical-mappings
                              ?space ?indep))))

(define-text-plan-operator
  :name describe-grapheme
  :effect (describe (grapheme ?grapheme) (space ?space))
  :constraints
    ((grapheme? ?grapheme)
     (get-properties-mapped ?grapheme ?space ?properties)
     (> (number ?properties) 2))
  :steps (FOR-ALL ?property in ?properties
            (describe (property ?property)
                         (grapheme ?grapheme) ?space)))
```

Figure 6: Sample Plan Operators.

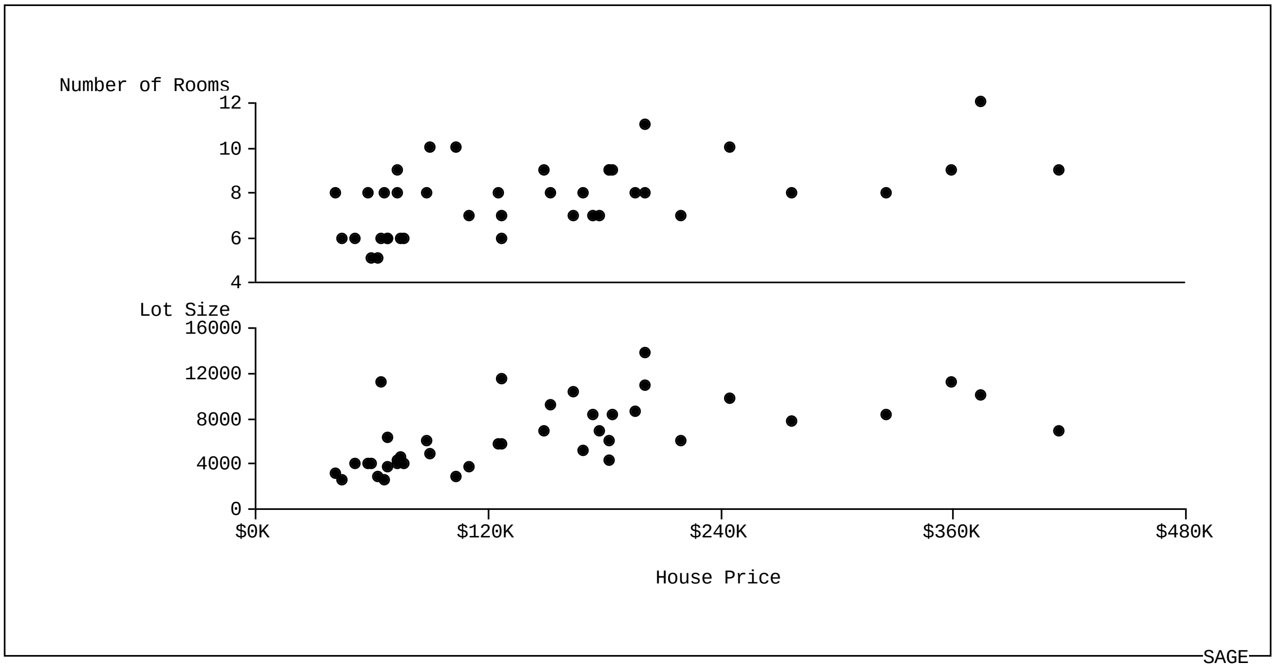
6.1 Strategies for Generating Captions

Explanations about information graphics can be classified into at least three categories based on the structural properties of the picture, as well as the structure of the underlying data attributes and their mapping to spaces and graphemes. These explanation strategies reflect the overall structure of the graphic presentation: the spaces are described left to right, and within each space, the explanation proceeds from the graphical clusters to the individual graphemes. In addition to these factors, the choice of explanation strategy also depends upon whether the spaces are aligned along a common axis, and around the functionally independent attribute. An attribute is functionally independent if it uniquely determines the values of all other attributes. For example, the house's street address uniquely determines the asking price, selling price, and so on. In contrast, the listing agency does not uniquely determine any of the other attributes in the relation. In our current system, an explanation strategy is selected as described below.

[4]The actual syntactic form used to represent the goals and speech acts in the implemented system is more complex; for the sake of clarity, we have shown simplified versions in this paper.

Graphic organized around the functionally independent attribute

Strategy 1: The first strategy is chosen when the data set contains a functionally independent attribute that is used as an organizing device or "anchor" for the entire graphic. This strategy applies when the graphic has only one space and the independent attribute is along one of the axes, or when there



These charts show information about house sales from data set PGH-23. In the two charts, the X-axis shows the selling prices. The top chart emphasizes the relationship between the number of rooms and the selling price. The bottom chart emphasizes the relationship between the lot size and the selling price.

Figure 7: Graphic with caption generated using strategy 3.

are multiple spaces and the independent attribute is mapped to the axis of alignment.

In this case, the explanation should reinforce the organizing role of the functionally independent attribute. One of the plan operators that implements this strategy is shown in Fig. 6. The first step in the body of this operator posts a subgoal to identify the data that is depicted in this picture as data about the independent attribute in a particular dataset. The second step posts a subgoal to describe the graphical structure that is the anchor and identify the independent attribute as being expressed by the anchor. The third step causes the planner to post a subgoal to describe each space in the picture relative to the anchor. A SAGE generated graphic and the associated explanation that illustrates this organizing principle is shown in Fig. 2. We go through this explanation in detail in the next section.

Graphic organized around dependent attributes

In cases where the graphic is organized around dependent attributes, the explanation cannot be structured around any of them. Instead the explanation emphasizes the *relation* between the dependent attribute(s) that serve as organizer(s). There are two strategies depending on whether or not the figure consists of multiple spaces.

- **Strategy 2:** When there is only a single space in the graphic, the explanation should emphasize the relation between the attributes encoded against the axes. A SAGE generated graphic and the associated explanation that illustrates this organizing principle is shown in Fig. 5.
- **Strategy 3:** If the graphic has multiple spaces and the axis of alignment encodes a dependent attribute, the explanation must describe each space independently, using the appropriate strategy for each space. That is, if a space is anchored on an independent attribute, use the strategy for describing anchored spaces. Otherwise, use the strategy for describing a space in which dependent attributes are expressed on the axes. Fig. 7 shows such a graphic and the corresponding explanation.

6.2 A Detailed Example

Now we consider in detail how our system generates the caption for the graphic in Fig. 2. This graphic consists of multiple spaces aligned with respect to an axis expressing a functionally independent data attribute. Therefore, the plan operator for strategy 1 shown in Fig. 6 is chosen. The first step of this operator posts a subgoal to identify the data that is depicted in this picture as data about the independent attribute (here, “house”) in a particular dataset (PGH-23). Satisfying this goal leads to sentence (1) in the caption.

The operator’s second step posts a subgoal to describe the graphical structure that serves as the anchor for the three spaces and identifies the independent attribute (“house”) as being expressed by the anchor (the Y-axis). This generates sentence (2). The clause “in the three charts” is included when describing the house attribute because the complexity assessment module indicated that this attribute is complex due to the alignment.

The third step causes the planner to post subgoals to describe each of the three spaces in the picture relative to the anchor, i.e., the Y-axis. Clauses (3)–(9) describe the first space, i.e., the left chart. Within the first space, there are two sets of graphemes: bars and square marks.

The operator for describing anchored spaces finds all of the attributes expressed in the space and groups those that have complexities by the grapheme that expresses them. The strategy for describing how an attribute is expressed depends on the types of complexity that exist for that attribute. As shown in Section 5, selling price and asking price were both associated with a complex grapheme. Clauses (4) and (5) clarify the mapping of these attributes to properties of the associated grapheme, i.e., the bar. In contrast, the agency estimate is rated complex solely because there are other graphemes in the same space, and therefore clause (6) simply identifies the square mark as the grapheme that expresses it.

Clauses (7)–(9) give examples of the attributes selling price, asking price and agency estimate. The strategy for clarifying the mapping of attributes to properties of the associated grapheme includes an optional step of giving examples. This optional step is expanded if the attribute has complexity of type "complex grapheme", "multiple graphemes", or "encoding technique complexity", unless this type of complexity was exemplified for a previous attribute (as recorded in the evolving text plan). As a result the description of the second space's attributes in clauses (10)–(11) does not include examples.

Finally, because there are no complexities associated with listing agency, and it is the only attribute expressed in the third space other than house (the anchoring independent attribute), clause (12) simply states that the chart expresses the attribute.

Notice that the caption generated for the graphic in Figure 3 differs from the caption in Figure 2, even though the dataset used for generating the graphics is the same. This is because the explanation reflects the different ways in which the graphic expresses the data in the two figures.

7 Conclusions and Future Work

The ability to generate captions to explain novel or creative information graphics is crucial for understanding how they express data. In this paper, we presented a general purpose method for generating explanatory captions for information graphics that employ a variety of graphical techniques to integrate multiple data attributes in a single display. The system generates captions based on: (1) a representation of the structure of the graphical presentation and its mapping to the data it depicts, (2) a framework for identifying the perceptual complexity of graphical elements, and (3) the structure of the data expressed in the graphic.

There are two parts to effectively using a graphic: (1) understanding how the graphic expresses its data, and (2) understanding how to use the graphic for a particular task. Thus far, we have addressed the first issue. Since a graphic may be used to support multiple tasks, the usefulness of explanatory captions would be increased if they included instructions for how to use the graphic for a given purpose. We plan to address this issue in future work.

Acknowledgments: We gratefully acknowledge John Kolojejchick's invaluable help in preparing the figures for this paper.

References

[Casner, 1991] Steven Casner. A task-analytic approach to the automated design of graphic presentations. *ACM Transactions on Graphics*, 10(2):111–151, April 1991.

[Dale, 1992] Robert Dale. *Generating Referring Expressions*. ACL-MIT Series in Natural Language Processing. The MIT Press, 1992.

[Elhadad, 1992] Michael Elhadad. *Using Argumentation to Control Lexical Choice: A Functional Unification Implementation*. PhD thesis, Columbia Univ., New York, NY, 1992.

[Feiner and McKeown, 1993] Steven K. Feiner and Kathleen R. McKeown. Automating the Generation of Coordinated Multimedia Explanations. In Maybury [1993], pages 117–138.

[Grosz *et al.*, 1995] Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. Centering: A Framework for Modelling the Local Coherence of Discourse. TR 95-01, IRCS, Univ. of Pennsylvania, Jan. 1995.

[Mackinlay, 1986] Jock D. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, 5(2):110–141, April 1986.

[Marks, 1991] J. Marks. *Automating the Design of Network Diagrams*. PhD thesis, Harvard Univ., Dept. of Computer Science, 1991.

[Maybury, 1993] Mark T. Maybury, editor. *Intelligent Multimedia Interfaces*. MIT Press, Menlo Park, CA, 1993.

[Moore and Paris, 1993] Johanna D. Moore and Cécile L. Paris. Planning Text for Advisory Dialogues: Capturing Intentional and Rhetorical Information. *Computational Linguistics*, 19(4):651–694, December 1993.

[Moore, 1995] Johanna D. Moore. *Participating in Explanatory Dialogues: Interpreting and Responding to Questions in Context*. MIT Press, Cambridge, MA, 1995.

[Nugent, 1983] Gwen C. Nugent. Deaf students' learning from captioned instruction: The relationship between the visual and caption display. *Journal of Special Education*, 17(2):227–234, 1983.

[Roth and Hefley, 1993] Steven F. Roth and William E. Hefley. Intelligent Multimedia Presentation Systems: Research and Principles. In Maybury [1993], pages 13–58.

[Roth and Mattis, 1990] Steven F. Roth and Joe Mattis. Data characterization for intelligent graphics presentation. In *Proc. CHI '90*, pages 193–200, New Orleans, LA., 1990. ACM/SIGCHI.

[Roth *et al.*, 1994] Steven F. Roth, John Kolojejchick, Joe Mattis, and Jade Goldstein. Interactive graphic design using automatic presentation knowledge. In *Proc. CHI'94*, Boston, MA, 1994. ACM/SIGCHI.

[Schmid, 1983] Calvin F. Schmid. *Statistical Graphics: Design Principles and Practices*. John Wiley and Sons, 1983.

[Wahlster *et al.*, 1993] Wolfgang Wahlster, Elisabeth Andre, W. Finkler, H. J. Profitlich, and Thomas Rist. Plan-based integration of natural-language and graphics generation. *Artificial Intelligence*, 63(12):387–427, October 1993.