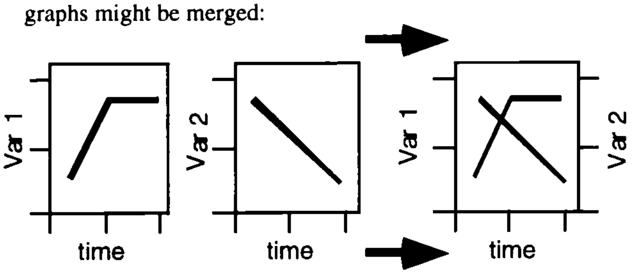
Summarization of Documents That Include Graphics

Robert P. Futrelle

Biological Knowledge Laboratory, College of Computer Science
161 Cullinane Hall, Northeastern University, Boston, MA 02115
futrelle@ccs.neu.edu http //www.ccs.neu.edu/home/futrelle/

Abstract

When documents include graphics such as diagrams, photos, and data plots, the graphics may also require summarization. This paper discusses essential differences in informational content and rhetorical structure between text and graphics, as well as their interplay. The three approaches to graphics summarization discussed are: *Selection*, in which a subset of figures is chosen; *Merging*, in which information in multiple figures is merged into one; and *Distillation*, in which a single diagram is reduced to a simpler form. These procedures have to consider the content and relations of the graphical elements within figures, the relations among a collection of figures, and the figure captions and discussions of figure content in the running text. We argue that for summarization to be successful, *metadata*, a manipulable representation of the content of figures, needs to be generated or included initially. Often, the textual references to figures are not very informative, so it will be necessary to generate metadata by diagram parsing, as we have done to develop *intelligent authoring systems* that will allow the author to easily include metadata. This paper introduces this new area of research with manual summarization examples and follows them with a discussion of automated techniques under development. For example, here is how two data graphs might be merged:



Introduction

The goal of automating the summarization of text has been pursued extensively, e.g., the work referenced in (ACL/EACL-Summarization 1997; Mani and Maybury 1998). But little has been done to produce summaries that select, merge, or distill the figures in documents to produce a smaller or simpler set of figures and captions, in spite of the fact that graphics plays a crucial role in so many documents. It is tempting to assume that one can index and summarize documents with graphics by focusing on captions or running text commentaries. Unfortunately, many important figures occur with little associated text – the figures "speak for themselves".

This paper lays out some of the problems and prospects for this new area of research on the graphical summarization of graphics. It first describes the basic characteristics of graphics, comparing it to text. A critical component of the approach is *metadata*, a formal description of graphics content that can be used by an automated summarization system. This is in part what diagram parsing produces (Futrelle and Nikolakis 1995). It then gives examples of manually constructed summaries using the strategies of *Selection*, *Merging*, and *Distillation*. Building on these, it then discusses the design of automated techniques – work in progress.

Basic Concepts and Terminology

Graphics and language are fundamentally distinct, because language builds logical structures from words, which are arbitrary signs. In contrast, graphics is *analogical*, using spatial structures as the information-carrying elements (Sloman 1995). But from another point of view, graphics and language are similar, since both arrange familiar and understood elements in various ways to state novel information. Documents can be characterized by their *genre* (journal article, newspaper article, book, etc.) and their *domain* (operating systems, microbiology, etc.). Text is essentially *propositional*, presenting statements in natural language that can be mapped into formalisms such as the predicate calculus. Latour has argued cogently (Latour 1990) that graphics has been critical for the growth of science and technology because authors can "show" a reader something remotely, without the author or the thing shown being present.

Graphics and language are different *modalities*, because they require distinct interpretation functions, whereas both may rendered in the same *medium*, e.g., on the printed page (Stenning and Inder 1995). Graphics can be divided into *veridical* entities, such as photographs and drawings of real-world structures, and *abstract diagrams* such as flow charts or data graphs. In graphics, there are also *conventional* classes of symbols, such as arrows, tick marks, and error bars, that are learned elements of the visual lexicon.

It is useful to distinguish the term, *graphics,* from more specific ones such as: *figures* which are instances of graphics in documents; *diagrams* which are line drawings; and *images* which are raster-based. Diagrams are built up from content-free items such as lines, curves, and closed regions such as polygons, called *vector data.* The text associated with figures can be in *captions* or in the *running text,* or included within the figures.

Metadata is normally propositional material that gives additional information about diagram structure and content but is not always made available for viewing by the reader.

Summarization of text or graphics can be *indicative* or *informative* (Paice 1990). The former presents material that indicates the subject domain, while the latter contains information that is a subset of the original document. The three types of graphics summarization we will describe are *selection* , *merging,* and *distillation.* In selection, a subset of the figures in a document is presented unchanged in the summary. Merging combines information from more than one figure into a single figure, whereas distillation, operates on a single figure to produce a simpler one. Both of the latter approaches involve *generation* of distinct new figures. All three methods may require parallel operations on the text, especially to generate captions.

Metadata

We define metadata as all information beyond the surface form of the document that a reader normally sees. Metadata is information in a form that can be reasoned about by automated procedures. Metadata for text can consist of word frequencies, collocational statistics, thesaural relations, sentence parses of varying sophistication, and discourse structure such as resolved anaphors. The description changes somewhat when we include diagrams, because it seems appropriate to consider text and graphics each serving as metadata for the other, e.g., a figure caption. But true graphic metadata consists of structural descriptions of figure content, which might be plain-text or a syntactic parse of the diagram (Futrelle and Nikolakis 1995) or a formal semantic structure. A simple example of graphics metadata is the numerical data behind a spreadsheet data plot.

In manual analysis, the characterization of a figure arises from the visual recognition and analysis done by the reader. In an electronic document, automated image understanding techniques could be applied to GIF and JPEG figures to develop metadata. This is a difficult problem, especially if the techniques are to be applicable across a variety of domains. If a figure is represented in vector form or is easily convertible to vector form, then there are techniques that we have developed to parse such figures to generate metadata (Futrelle and Nikolakis 1995).

Authors generate metadata as a matter of course. The superstructure of a document, represented in formal terms such as SGML markup, is metadata. Links to other documents such as citations or HTML hyperlinks [illegible] another. For graphics, text within figures as well as captions augment the purely visual material. HTML *image maps* allow regions within figures to be defined that link to locations in the same or different documents.

Graphics metadata is normally of little direct interest to a human reader, because it would contain knowledge that would be self-evident when viewing the images. Complex and formally structured metadata such as parse trees or FOPC representations would not be of interest to a reader either.

Intelligent Authoring Systems (IAS)

Using these systems we have proposed (Futrelle and Fridman 1995), an author could easily generate metadata, and most importantly, could generate structured graphics for which the metadata is developed as an integral and non-obtrusive part of the drawing process. (That earlier work focuses primarily on an IAS for text, but the principles apply to graphics.) Many drawing and writing tools exist, though most are not particularly intelligent and few produce useful metadata.

Probably the simplest approach to metadata would be for the author to provide a plain-text description of figure content, in a form that would go beyond the normal figure caption. It would be essentially the same as any verbal (textual) account of a scene, such as in a telephone conversation, a novel, or in a written news report lacking figures. The plain-text descriptions could then be used by conventional text-based summarization systems.

An alternative method would be to use a form-based interface in an IAS, possibly keyed to the domain, allowing the author's input to be turned into a more structured form such as knowledge frames. The system could allow the author to indicate links between form elements and objects or regions in the diagram being drawn.

To generate more structured graphics metadata, we first consider normal drawing applications. In these, predefined elements are available (lines, ovals, polygons, Bezier curves, etc.). In addition, constraints can be imposed such as positioning to a grid, or aligning or spacing horizontally or vertically. Transformations such as translation, scaling and rotation are available.

The key to building an IAS for graphics that builds the needed metadata is to map various predefined graphical elements onto semantically meaningful concepts. This is what is customarily done in CAD systems, e.g., a VLSI design system in which "lines" are interpreted by the system as electrical conductors and the designer selects various items defined by their function and their form, e.g., a NAND gate, and places and "connects" them in a drawing.

The above approach to diagram authoring can be described as using a "semantic construction kit". The final structures, beyond the purely visual form, constitute graphical metadata that could be used for indexing, search, and automated graphics summarization.

Summarization by Selection

In this example, we consider three figures, grouped as a single entity, Fig. 1, and formulate criteria for the selection of one of them as the summary (often called "extraction" in the text summary literature). The figures originally appeared together as color halftones in *Scientific American* (Diels et al. 1977) (pg. 53) but are represented here as line drawings.

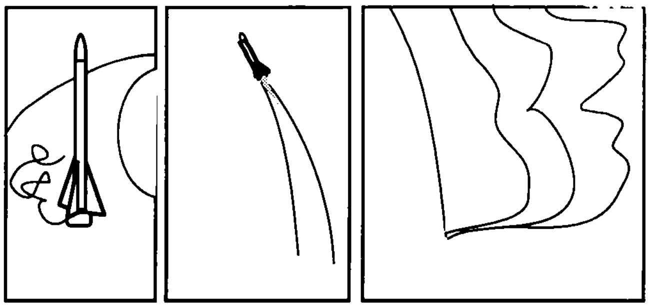


Figure 1. Original caption: "ROCKETS trigger lightning in various field experiments. The small, specially constructed missile (*left*) carries at its base, a spool of thin, grounded wire that unwinds in flight (*center*). The first stroke triggered in this way follows this copper filament and creates a conductive channel of ionized air; later strokes of the same flash event (which can occur repeatedly within a fraction of a second) travel along increasingly tortuous routes as the wind deforms the conductive path (*right*)." The line drawings are for discussion purposes, schematizing the original halftones.

The only mention of rockets in the running text of the article is: "... researchers ... can trigger lightning using small rockets that trail a thin, grounded wire." This statement parallels the first sentence in the caption. If we were to generate a summary related to rockets using the running text and caption by any of a number of text-based methods, the dominant content would be *rockets trigger lightning*, appearing in both places; it is the most *salient* (Boguraev and Kennedy 1997). Other approaches for discovering the most significant terms use the tf*idf measure (Salton 1989), or assume Bernoulli distributions of terms, which is useful for small samples (Dunning 1993). All of these methods compare the local frequencies of terms within an article with the frequencies in a broader corpus – simply choosing the highest frequency open-class terms is not adequate. In analyzing captions in this way, it may be useful to consider saliency measures across a set of occurrence contexts: within figures, in captions, in referring text, in the article, in a containing technical corpus, and in a larger corpus containing additional non-technical documents. It should be possible to exploit such a hierarchical sequence of contexts to generate a hierarchical set of descriptive text items.

Only the right-most figure illustrates *rockets trigger lightning*. The first two show aspects of the rockets with no explicit inclusion of lightning. The left figure shows the wire and spool, and the center shows a "generic" rocket launch with no distinguishing features. The selection procedure should choose the right-most figure in this case . The automation of this selection is not easy, because of the complexity of the caption. For example, the terms "thin grounded wire" and "copper filament" are co-referential. It is even more difficult to deduce from the caption that the rocket is involved in the third figure, but it is the proximate cause of the "copper filament" causing "a conductive channel of ionized air." This requires reasoning beyond syntax to world knowledge, which forces the analysis to focus on a specialized (sub-)domain. (To appreciate the difficulties of such an analysis, the reader should imagine making the figure selection based entirely on the text, without looking at the figures themselves.)

Another important point about the caption is the distribution of text devoted to the three figures. There is not a one-to-one correspondence between text spans in the caption and the three component figures, but an approximate division can be identified. The first caption sentence serves as the title (7 words). The segment, "The small, ... grounded wire" (16 words) corresponds to the *left* figure, the overlapping segment, "a spool ... in flight (*center*)" (11 words) corresponds to the *center* figure, and the longest span, "The first stroke .... path (*right*)." (49 words) corresponds to the *right* figure. Though this sort of analysis is attractive, there is a potential circularity to the argument, since we are relating the text to figure content, which may be know only through the text!

The salience measures are automated techniques, but the determination of the figure content in ways that can relate to the salient terms is the challenge. It is not even appropriate to assume that there will always be text from which the most salient content is determined, which would then control figure selection. Indeed, a figure may itself deliver important information by itself without much supporting text.

Metadata for selection.

It is worth discussing how the selection decision above could profit from the appropriate graphics metadata. Plain-text metadata for the figure might take the following form:

> "Figure, rocket (left): The rocket is sitting stationary on the ground on a sidewalk. Attached to the rocket base is a spool of copper wire. Some of the wire is shown unwrapped from the spool."
>
> "Figure, rocket (center): Shows the horizon and a portion of the sky that contains the rocket and the trailing wire. The rocket is ascending in flight. The rocket is trailing the copper wire (not visible). The lower end of the wire is attached to the ground so the wire can conduct electricity between the sky and the ground."
>
> "Figure, rocket (right): Shows the horizon and a portion of the sky that contains the rocket and the trailing wire and many lightening strokes. (Rocket and wire not visible). The first lightning stroke triggered by the wire

follows the wire to the ground, creating a conductive channel of ionized air. Thus, the first lightening path follows the smooth path of the wire. Later lightning strokes of the same flash event (which can occur repeatedly within a fraction of a second) travel along increasingly tortuous routes as the wind deforms the conductive path."

Note that the right-most figure metadata is similar to material in the original caption, since that was quite detailed. The metadata contains definite noun phrases but no pronouns, to facilitate natural language analysis. A text-based summarization system using the plain-text would presumably conclude that the third item is the most salient.

A more structured approach to metadata for the figures could be used for indexing and summarization. This could be generated by an IAS, a form-based interface, or parsing the plain-text metadata. An example is shown here in a hybrid frame-like structure:

```
Figure ID:  rocket, left
    Foreground:
            Item: rocket
                      State:  stationary
                      Location:  on ground
    Background:
            Items:  walkway, building, trees

Figure ID:  rocket, center
    Foreground:
            Item:  rocket
                      State:  flying
                      Location:  in air
    Background:
            Items: sky, clouds
    Implicit:
            trailing(rocket, wire)

Figure ID:  rocket, right
    Foreground:
            Item-set:  lightning
    Background:
            Items: sky, clouds
    Implicit:
            equal(path(wire),
              path(first(lightning)))
```

The "`Implicit`" slot holds information about the figures that is not necessarily visually obvious.

Rhetorical Structure Theory (RST)

Based on the figure content, it is possible to argue for the rhetorical structure shown in Fig. 2.

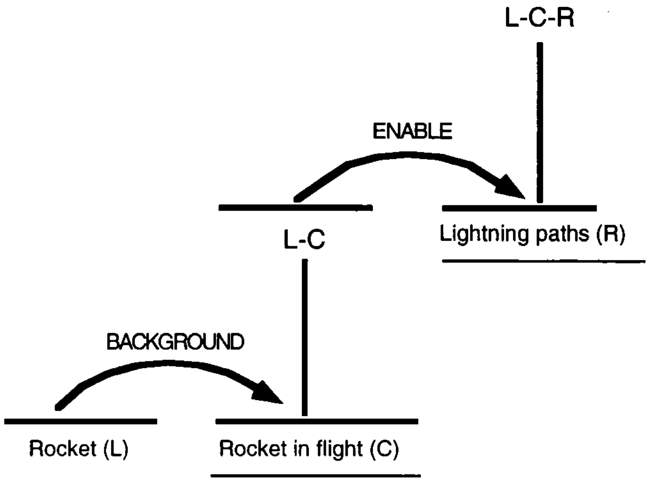


Figure 2. Rhetorical Structure Theory (RST) tree (Mann and Thompson 1987) for the three figures in Fig. 1. The nuclei are underlined, with satellites occupying the other branch of each relation. The nucleus that is attached to the root is the right-most figure (R), which is retained as the summary by selection. The left-most figure (L) is the *background* for the center one (C), and the left merged node (L-C) *enables* the final, right-most, figure.

Rhetorical structures such as the one in Fig. 2 are typically generated by hand, because automated generation depends on rather deep understanding of content. They are nonetheless useful for discussion purposes.

In the rhetorical structure of figures, as with text, it is important to keep in mind that there is an *informational* component that is focused on the content of the figures, as well as an *intentional* component that focuses on the author's goal of emphasizing certain points in the discourse. These are not entirely separate concepts and one does not necessarily dominate the other (Moore and Pollack 1992; Nicholas 1996).

Summarization by Merging

Merging the information from various figures into a single one could be used for the rocket example, but it is particularly appropriate for plots of data, especially for sets of similar or contrasting data. It is quite common to plot more than one set of data in a single data graph to effect this type of comparison. This merging within a single figure is already in an efficient form and difficult to reduce further. So we will concern ourselves with data in multiple figures, e.g., the data in the four subfigures shown in Fig. 3, taken from (Kim, Watrud, and Matin 1995).

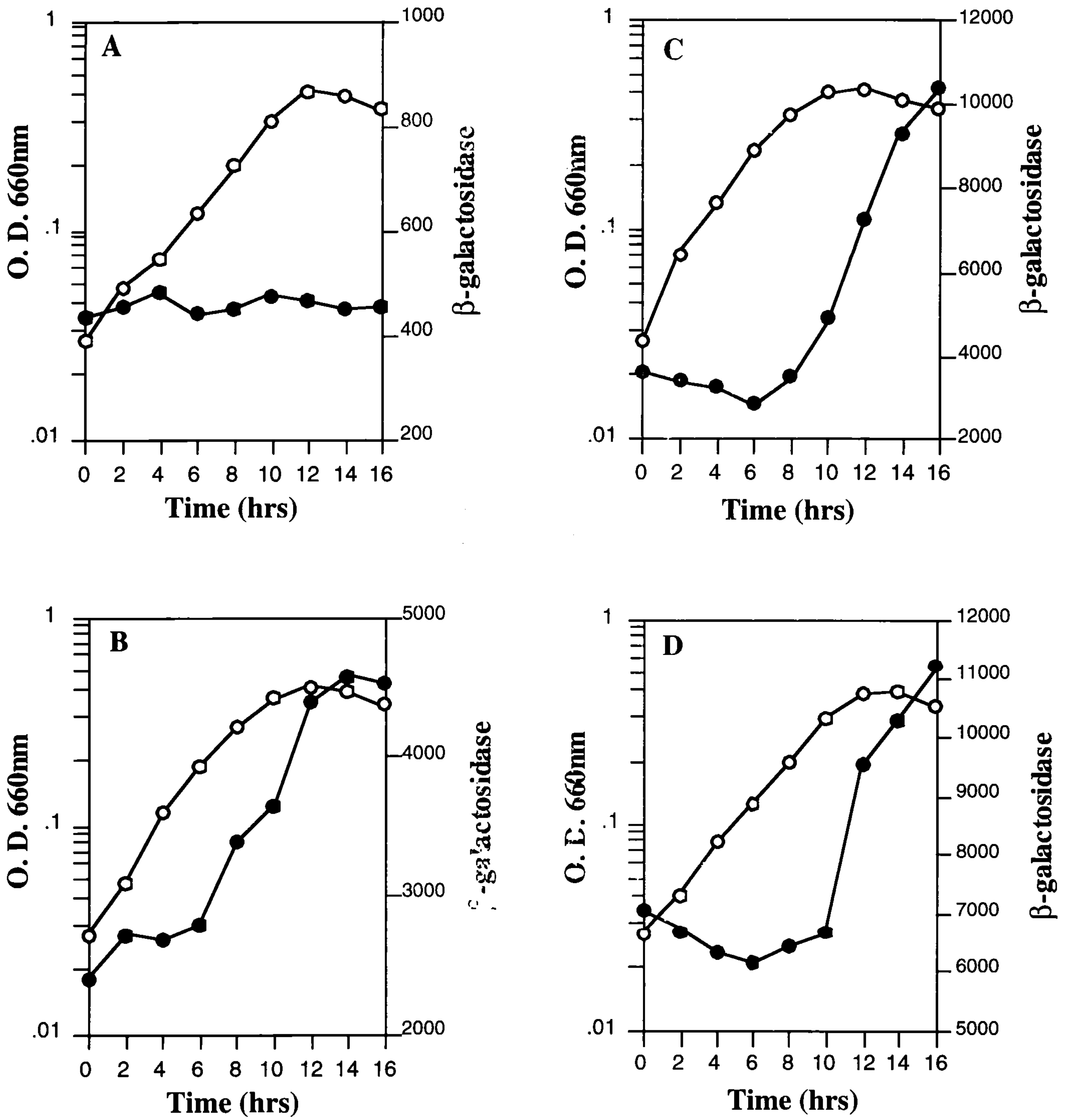


Fig. 3. Original caption: "β-Galactosidase synthesis during growth and stationary phase in the mini-Tn5::*lacZ*1 fusion mutants of *P. putida* MK1. (A to D) MK201, MK104, MK107, and MK114, respectively. Cultures were grown on M9 medium plus 0.1% glucose; other methods are described in the text. Open symbols, growth; solid symbols, b-galactosidase activity." This was Figure 2 in the referenced document. It is faithfully redrawn from the original.

In Fig. 3, there are four open circle datasets and four solid circle datasets. These, rather obviously, fall into three behavior classes. All the open circle datasets behave in essentially the same way; they form the first class. The solid circle datasets in B, C, and D are similar to one another; they form a second class. The solid circle dataset in A is the outlier; it behaves quite differently and forms a third, singleton class. Given that there are only three classes of behavior, it is not difficult to construct a rather succinct summary, using only one plot, as shown in Fig. 4.

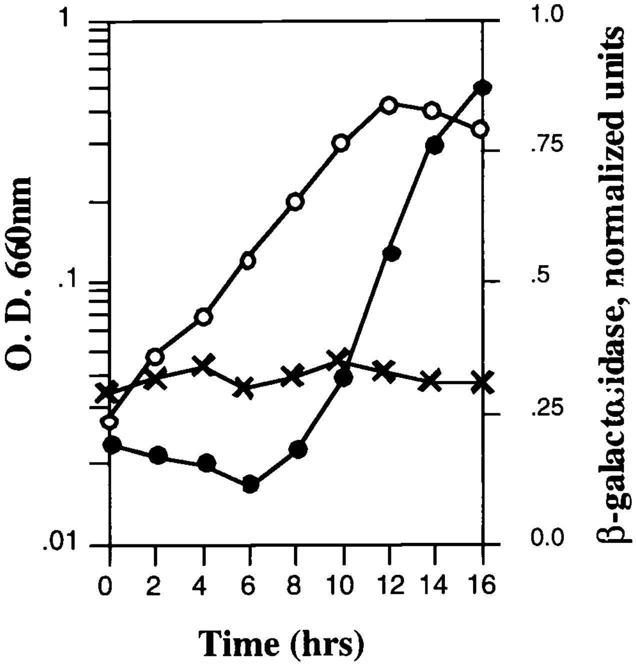


Figure 4. A merged graphic, a data plot, showing the three distinct classes of behavior evident in Fig. 8. The generation of this plot would also require the cogeneration of a caption which, at the very least, must identify the three data sets, i.e., that the solid circles arose from bacterial mutant strains MK104, MK107, and MK114, and the crosses from strain MK201. The open circle data applies to all four strains. Since the numerical values of the right-hand data scale differ in each of the four original data graphs, only normalized values are used in the merged plot. Each dataset group could be averaged, but here only example datasets selected from single plots are used, which shows the data variance better than an average would.

The automation of the generation of Fig. 4 is more straightforward than for the other examples, since statistical analysis of the datasets could reveal the appropriate equivalence classes. Once this was done, the generation of a revised caption would not be too difficult; it could be structured as an addendum to the original caption. Hovy has emphasized the production of summaries using analysis followed by *generation* of summarization text (Hovy and Lin 1997). All of our examples require generation of altered captions.

Much of the data published in data-intensive fields such as biology is not as clear-cut as in this example. The merging of data here is facilitated by the high degree of data redundancy in the four subfigures. To maximize information in a research article, authors try to include only figures that present non-redundant information. By their very nature, such sets of non-redundant data are difficult to merge. In such cases, selection and distillation techniques would have to be used, techniques that make decisions as to which information is the most important or germane to the document, discarding what is more peripheral.

Summarization by Distillation

In this example, we consider a single figure, 5, describing the series of operations from Java source code through to its execution on hardware (Hamilton 1996). The goal is to generate a simpler figure that omits and/or combines the objects in the original (Futrelle and Chambers 1998). The strategy is one that can be rather universally applied to flow charts: the omission of intermediate steps. Such a simple technique is inadequate for text summarization, because the most salient sentences are sometimes in a single paragraph, or may be the first sentence of each of a sequence of paragraphs. In addition, we take into account the diagram superstructure, its columnar organization.

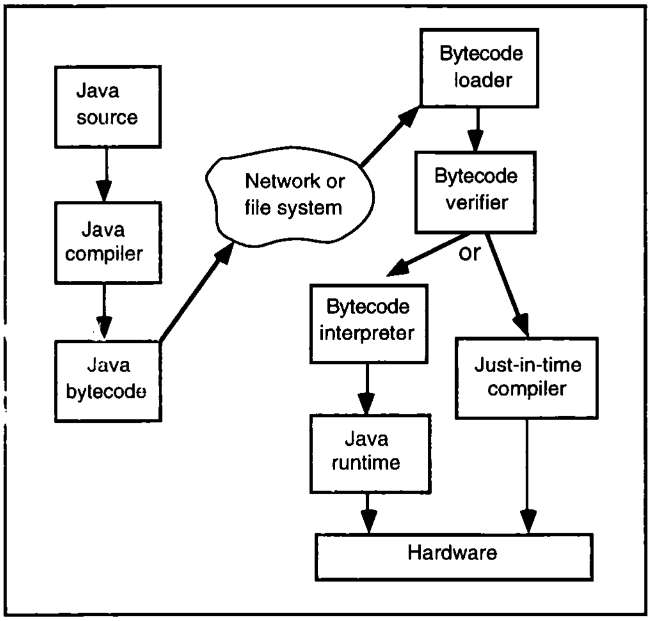


Figure 5. Original caption: "Executing a Java applet" This was Figure 1 in the referenced document. It is faithfully redrawn from the original.

Fig. 5 shows the full original figure from the document, with 10 stages and 10 arrows. The figure is a directed acyclic graph (DAG) organized as three vertical columns, with the right-most column containing a split and merge. There is no other figure in the original article that resembles Fig. 5, so we must either omit it or summarize it in some way. We show three successively simpler manual summaries below, which attempt to distill the most important structures from the original.

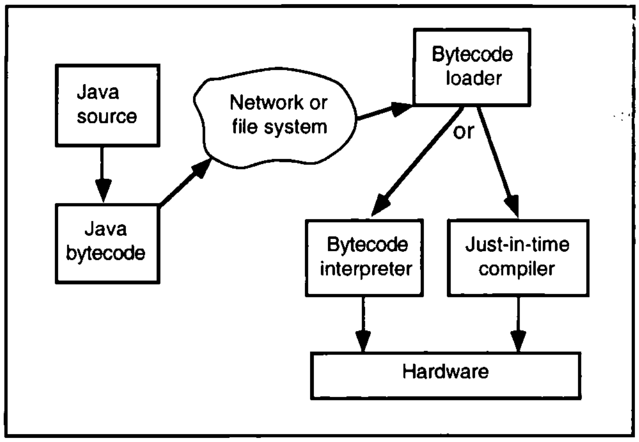
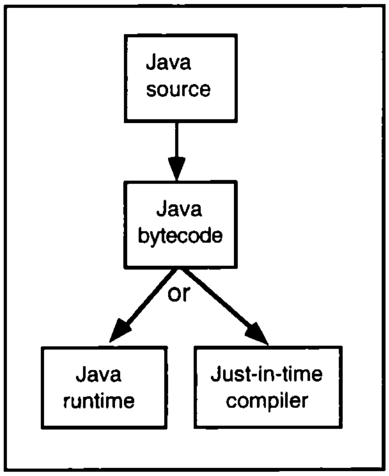


Figure 6. Executing a Java applet. Three internal nodes from Fig. 5 have been deleted.

Fig. 6, shows a moderate reduction of the figure to a summary with 7 stages and 7 arrows. The intermediate compiler step is omitted, as well as the Bytecode verifier and Bytecode interpreter steps. The distillation removes the intermediate stage in the left-most column, retains the only item in the middle column and deals with the right-most column by omitting the common intermediate verifier stage and the intermediate runtime stage on the left split section headed by the interpreter stage. It could be argued that every stage between the loader and the hardware could be omitted in the right-most column. One argument against this, is that it would give strong weight to the non-specific item, "Hardware".



Figure'7. Executing a Java applet. Additional internal nodes and final "Hardware" node deleted.

Fig. 7 is a further reduction to 4 stages and 3 arrows from the original 10 and 10. What is retained here is the initial stage from Fig. 6, the Java source, and the non-trivial end stages of the process, the interpreter and JIT compiler (Hardware stage omitted). The decision to include the end-stage of the left-most column, the bytecode, is a difficult one, but "bytecode" is prominent in the running text, whereas the loader is a generic stage that receives little discussion. Depending on its prominence in the text, it may be appropriate to also include the Network or file system stage. Viewing the figure in the large, this stage is intermediate between the processing the source, on the left, and execution, on the right.

There are two textual references to the figure in the original document. The first is the caption, only four words, but containing critical content. The only explicit reference to the figure in the running text is, "In this case, program integrity is guaranteed by the runtime, shown in Figure 5, which verifies that the bytecode corresponds to valid Java code and does not contain any viruses." This discussion is a bit confusing, since the verification is shown being done immediately after loading and the runtime is shown as a separate later stage. But the text does emphasize the verifier which we have omitted based on an analysis of the figure itself. This could argue for the inclusion of the verifier stage in even the simplest summary, as shown in Fig. 8.

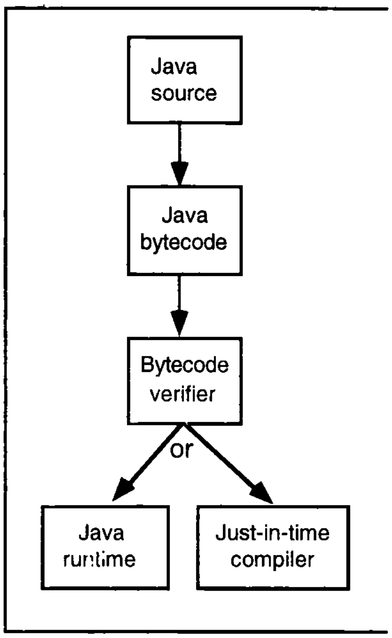


Figure 8. Executing a Java applet. Verifier node reinserted because of its mention in running text.

It is important to note that some of the items in the original Fig. 5 are not mentioned anywhere in the running text of the article. The most notable example is the lack of any discussion of the Just-in-time compiler.

It is not difficult to imagine a content representation for Fig. 4 that could be used by a distillation algorithm. The figure is in the form of a DAG, so the representation of the figure can be exactly that, a directed acyclic graph, with text contained in each node, Fig. 9. In addition, the columnar structure of the original figure is indicated by rectangles grouping the nodes.

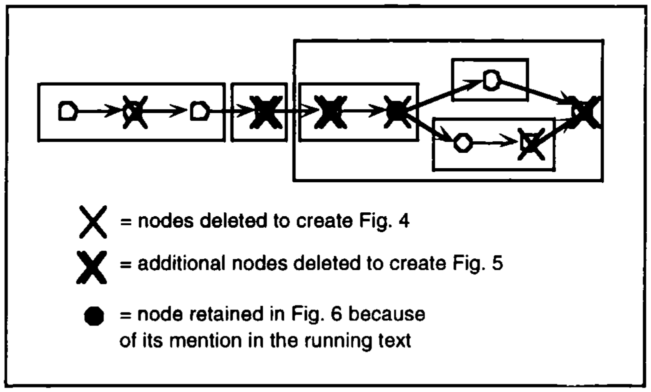

X = nodes deleted to create Fig. 4
X = additional nodes deleted to create Fig. 5
● = node retained in Fig. 6 because of its mention in the running text

Figure 9. Nodes deleted from Fig. 5 to create figures 6 and 7. Retention of node for Fig. 8.

The distillations are shown schematically in Fig. 9. The algorithm consists of first deleting interior nodes of triples and final nodes of pairs, excepting the final leaf node. The second set of deletions removes the node which is the interior of the triple at the highest level of organization of the DAG (Network or file system) and deletes the low-information (Hardware) node on the right, retaining the nodes that distinguish the two branches. Clearly, we are dealing with a *partitioned* DAG. The partitioning allows a hierarchical approach to be used. A similar argument can be made for the generation of Fig. 8, except that the verifier node is retained rather than deleted, because of the reference to it in the running text. This is indicated by the filled circle in Fig. 9. To summarize, the uncrossed nodes in Fig. 9 are all that are retained in the simplest summary, Fig. 7, while the filled node is added to those to create the five nodes in Fig. 8. The form shown in Fig. 9 emphasizes the topology of the graph, its *topovisual* structure (Harel 1995), since node labels are not included and the metric layout of the original figure is not relevant.

An alternate view is to write the partitioned DAG as a series of indented structures analogous to sections and subsections of a text. Then the algorithm could be thought of as retaining the initial "sentence" of each paragraph and deleting interior or final sentences. This puts the approach more on a par with text summary algorithms. An RST tree could be built for this example also.

Approaches to Automation

We will assume for this discussion that the metadata for figures already exists. This assumption must be tempered, i.e., it isn't appropriate to assume that the metadata contains all high-level data about the graphics and text and all their interrelations. That would essentially assume the result, allowing us in principle to read off the summary from the metadata. Instead, we'll assume that only the metadata for the diagram structure itself is available. A number of the operations described below can be viewed as building the additional metadata needed to construct a summary. The goal assumed will be the distillation of a single figure, as in the last example. We will further assume the most common mixture of text and graphics which is the following:

- The graphics contain only a modest amount of text.
- The caption discusses only limited aspects of the figure, perhaps not including its significance.
- The running text discusses many aspects of the topic illustrated in the figure but not all of it in direct reference to the items displayed in the figure.

Using these assumptions, the following computational strategy could be used to distill a diagram. Steps 1-4 focus on text salience, 5 on graphics structure and salience, and 6-10 on the distillation process:

1. Compute the statistical linkages between the running text, caption, and text within the diagram.
2. Find cue constructs in text that point to the figure to pinpoint relevant text, e.g., "is shown", "the figure demonstrates....", "Fig. 15 shows....", etc.
3. Locate definite noun phrases that appear to reference the figure content.
4. Using 1-3, determine the most salient text-related portions of the figure.
5. Using genre- and domain-specific graphics routines, tabulate all topological and layout structures of the diagram, e.g., looking at initial and final elements, topology, alignments, graphically and textually emphasized elements, etc.
6. Pick the graphical elements to be retained or emphasized in the distilled version, using measures from 1-5.
7. If elements are to be omitted, use topological or layout containment to "seal the gaps". This can be aided by coalescing low-scoring elements into single units.
8. If elements are to be emphasized, use properties such as size, color, and central location to emphasize them.
9. Restructure text to match the distilled graphics using text summarization procedures focused on the text most closely related to the summary diagram.
10. Generate the final summary graphics and text.

Though the steps above are not concise directions on how to build summarization algorithms, they are adequate to guide the implementation development that we are currently pursuing.

Discussion

The major aspects of the rhetorical role of graphics are that graphics can present complex information succinctly and that the reader can retain and use information in the visual modality, rather than merely having "read" or "heard" a discourse. As in text, there need be no explicit signal of a rhetorical relation involving graphics. For example, a magazine article on environmental legislation could be headed by a color photo of a dead stand of trees. Its role as an *example* of the *result* of air pollution / acid rain would be clear without any specific mention in the text. Graphics is often exemplary and/or extensional, making its point with an example or with contrasting cases. Many abstractions easily expressed in language are essentially impossible to represent in a static graphic, e.g., a "tendency" to do something, or a "hope" that something will happen, or that a certain method is "reliable". But on the whole, graphics can participate in virtually all the text rhetorical relations discussed in the classic paper (Mann and Thompson 1987).

Grammar-based approaches to diagram parsing can generate useful metadata for diagrams (Futrelle and Nikolakis 1995; Golin 1991; Helm, Marriott, and Odersky 1991; Wittenburg, Weitzman, and Talley 1991). The results of the syntactic analysis can be used in the generation of semantic representations (Futrelle and Nikolakis 1995) that can in turn be used in summarization.

Statistical analysis of data can result in an extreme form of distillation to produce what is essentially a summary, e.g., the computation of the mean and standard deviation for a dataset, or the fitting of a straight line or simple curve to a dataset. When a point statistic such as a mean is computed, the result does not even require a data plot.

The nature of graphics allows summarization in ways that are quite impossible with text. One example is the use of "thumbnails" which are figures that are markedly reduced in size from the original. Doing the same for text would render it unreadable, but graphics thumbnails are often quite useful. They are used frequently on the World-Wide Web to reduce download times and space requirements.

The interactive nature of graphics can be used in summarization. For example, thumbnails used on the Web can normally be expanded to larger and more detailed form by selecting them. Another tool is color or other forms of highlighting. They allow a system to display rather complex structures, but then to selectively colorize or bold certain elements of the viewer's choice, or put other information in a muted background form. Because of the gestalt nature of much graphics, the essentially instantaneous recognition that can occur, these techniques can be quite useful. When this is done with text it places a greater burden on the viewer, who would have to devote time to reading each new piece of text, e.g., when following hyperlinks in a browser. These interactive techniques allow a higher information density than would normally be feasible for hardcopy, since the density is selectively lowered when certain subsets of information are highlighted or others deleted from view.

Even more sophisticated interactions can easily be imagined, e.g., selecting a portion of text that discusses a subset of the information in a figure could lead to the highlighting of only that information, or hiding the non-referenced information. Similarly, selecting a portion of a figure could lead to the display of only that portion of the text that discusses it. Such an interactive mapping between corresponding symbolic and graphical structures is used in our Diagram Understanding System Inspector (DUSI) (Futrelle and Nikolakis 1995). HTML image maps are similar (Schengili-Roberts 1996). The inverse, mapping from a click to a graphic region, is available in some Web pages in which, for example, a click on a city map will execute a CGI script (Gundavaram 1996) that will cause the map to be recentered or zoomed around the click point. A shift from a detailed street map to a broader area city or region map in which only major arteries are shown is a form of summarization by distillation. The opposite shift from a broad view to a local view is a form of summarization by selection, actually with expanded rather than reduced detail.

Other lines of research that can contribute to graphics summarization are the work on diagram generation (North 1997), and the co-generation of text/graphics explanations (André and Rist 1993; Feiner and McKeown 1990; Marks and Reiter 1990). There is a great deal of relevant research on text summarization, e.g., (ACL/EACL-Summarization 1997; Mani and Maybury 1998), which we have profited from, but the focus on graphics in this paper and limitations of space have prevented any detailed discussion of it here. This Symposium is replete with up-to-date papers on the topic, which in turn refer to the large and burgeoning literature on text summarization. For additional results by our group on graphics summarization, see (Futrelle and Chambers 1998).

Conclusion

Document summarization including graphics is a worthy goal and a necessary component of any system that claims to handle document content in full. Though there are overlaps between summarization concepts and techniques for graphics and text, graphics has a distinct analogical character. It requires its own concepts and methodology that go beyond text but which are integrated with text. We have illustrated how manual construction of summarization involving graphics can be done. The challenge is to automate them. Automation will depend in part on developing better content representation techniques for graphics in future electronic document systems. This metadata will be built by a combination of author-contributed information and direct analysis of figure content and of the content of graphics-related text.

Acknowledgments. Thanks to Dragomir Radev (Columbia), Linda Mizushima (USC/ISIS library), and Mark Maybury (Mitre).for furnishing me material related to summarization.

References

ACL/EACL-Summarization. 1997. *Proceedings of ACL/EACL'97 Intelligent Scalable Text Summarization Workshop*. Madrid: Assoc. Computational Linguistics.

André, Elisabeth, and Thomas Rist. 1993. The Design of Illustrated Documents as a Planning Task. In *Intelligent Multimedia Interfaces*, edited by M. T. Maybury. Menlo Park, CA: AAAI Press/ The MIT Press.

Boguraev, Branimir, and Christopher Kennedy. 1997. Salience-based Content Characterisation of Text Documents. Workshop on Intelligent Scalable Text Summarization, at Madrid, Spain.

Diels, Jean-Claude, Ralph Bernstein, Karl E. Stahlkopf, and Xin Miao Zhao. 1977. Lightning Control with Lasers. *Scientific American* 277 (August):50-55.

Dunning, Ted. 1993. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics* 19 (1):61-74.

Feiner, Steven K., and Kathleen R. McKeown. 1990. Coordinating Text and Graphics in Explanation Generation. AAAI 90.

Futrelle, R. P., and Tyler Chambers. 1998. Summarization of Diagrams in Documents (submitted). In *Advances in Scalable Text Summarization*, edited by I. Mani and M. Maybury: (to be published).

Futrelle, Robert P., and Natalya Fridman. 1995. Principles and Tools for Authoring Knowledge-Rich Documents. DEXA 95 (Database and Expert System Applications) Workshop on Digital Libraries, at London, UK.

Futrelle, R. P., and N. Nikolakis. 1995. Efficient Analysis of Complex Diagrams using Constraint-Based Parsing. ICDAR-95 (Intl. Conf. on Document Analysis & Recognition), at Montreal, Canada.

Golin, E.J. 1991. Parsing Visual Languages with Picture Layout Grammars. *Journal of Visual Languages and Computing* 2:371-393.

Gundavaram, Shishir. 1996. *CGI Programming on the World Wide Web (Nutshell Handbook)*: O'Reilly & Associates.

Hamilton, Marc A. 1996. Java and the Shift to Net-Centric Computing. *Computer* 29 (August):31-39.

Harel, David. 1995. On Visual Formalisms. In *Diagrammatic Reasoning. Cognitive and Computational Perspectives*, edited by B. Chandrasekaran, J. Glasgow and N. H. Narayanan. Menlo Park, CA, Cambridge, MA: AAAI Press, MIT Press.

Helm, R., K. Marriott, and M. Odersky. 1991. Building Visual Language Parsers. CHI '91 - Proc. Conf. on Human Factors in Computing Systems, at New Orleans, Louisiana.

Hovy, Eduard, and Chin Yew Lin. 1997. Automated Text Summarization in SUMMARIST. Workshop on Intelligent Scalable Text Summarization, at Madrid, Spain.

Kim, Youngjun, Lidia S. Watrud, and A. Matin. 1995. A Carbon Starvation Survival Gene of *Pseudomonas putida* is Regulated by σ54. *J. Bacteriology* 177 (7):1850-1859.

Latour, Bruno. 1990. Drawing things together. In *Representation in Scientific Practice*, edited by M. Lynch and S. Woolgar. Cambridge, MA: MIT Press.

Mani, Inderjeet, and Mark Maybury, eds. 1998. *Advances in Scalable Text Summarization*: (to appear).

Mann, William C., and Sandra A. Thompson. 1987. Rhetorical structure theory: A theory of text organization: USC/Information Sciences Institute.

Marks, Joseph, and Ehud Reiter. 1990. Avoiding Unwanted Conversational Implicatures in Text and Graphics. AAAI 90.

Moore, Johanna D., and Martha E. Pollack. 1992. A Problem for RST: The Need for Multi-Level Discourse Analysis. *Computational Linguistics* 18 (4):537-544.

Nicholas, Nick. 1996. *Parameters for Rhetorical Structure Theory Ontology*. Melbourne: http://www.arts.unimelb.edu.au/Dept/LALX/postgrad/rst_ontology.html.

North, Stephen, ed. 1997. *Graph Drawing (Symposium on Graph Drawing, GD '96)*. Vol. 1190, *Lect. Notes Computer Sci.* Berlin: Springer.

Paice, Chris D. 1990. Constructing literature abstracts by computer: Techniques and prospects. *Information Processing & Management* 26 (1):171-186.

Salton, G. 1989. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Reading, MA: Addison-Wesley Pub. Co.

Schengili-Roberts, Keith. 1996. *The Advanced Html Companion*: Academic Press.

Sloman, Aaron. 1995. Musings on the Roles of Logical and Non-Logical Representations of Intelligence. In *Diagrammatic Reasoning. Cognitive and Computational Perspectives*, edited by B. Chandrasekaran, J. Glasgow and N. H. Narayanan. Menlo Park, CA, Cambridge, MA: AAAI Press, MIT Press.

Stenning, Keith, and Robert Inder. 1995. Applying Semantic Concepts to Analyzing Media and Modalities. In *Diagrammatic Reasoning. Cognitive and Computational Perspectives*, edited by B. Chandrasekaran, J. Glasgow and N. H. Narayanan. Menlo Park, CA, Cambridge, MA: AAAI Press, MIT Press.

Wittenburg, K, L. Weitzman, and J. Talley. 1991. Unification-based grammars and tabular parsing for graphical languages. *Journal of Visual Languages and Computing* 2 (4):347-370.