Proceedings of the Twenty-Ninth AAAI Conference on Innovative Applications (IAAI-17))

# A Machine Learning Approach for Semantic Structuring of Scientific Charts in Scholarly Documents

**Rabah A. Al-Zaidy**
alzaidy@psu.edu
The Pennsylvania State University
University Park, PA

**C. Lee Giles**
giles@ist.psu.edu
The Pennsylvania State University
University Park, PA

## Abstract

Large scholarly repositories are designed to provide scientists and researchers with a wealth of information that is retrieved from data present in a variety of formats. A typical scholarly document contains information in a combined layout of texts and graphic images. Common types of graphics found in these documents are scientific charts that are used to represent data values in a visual format. Experimental results are rarely described without the aid of one form of a chart or another, whether it is 2D plot, bar chart, pie chart, etc. Metadata of these graphics is usually the only content that is made available for search by user queries. By processing the image content and extracting the data represented in the graphics, search engines will be able to handle more specific queries related to the data itself. In this paper we describe a machine learning based system that extracts and recognizes the various data fields present in a bar chart for semantic labeling. Our approach comprises of a graphics and text separation and extraction phase, followed by a component role classification for both text and graphic components that are in turn used for semantic analysis and representation of the chart. The proposed system is tested on a set of over 200 bar charts extracted from over 1,000 scientific articles in PDF format.

## Introduction

Search engines rely mainly on metadata of images to include them in search results. In large scholarly paper repositories, many publications contain experiment sections. Illustrative charts and tables are among the most common methods to present data of evaluation results. Additionally, charts exist in the web and are used widely in a variety of domains such as finance and news articles. In many instances where these charts are present, the numeric data they contain in graphic form is rarely mentioned in plain English text elsewhere. Thus, we propose an approach to extract the data from chart images, specifically bar charts, by using features extracted from both graphic components and text components. Charts contain both text and graphic components that are correlated to represent the data. For instance, the y-axis scale in a chart is represented by a line that contains ticks (or marks) to specify the unit step of the chart and the text containing a number describes the numerical amount for that unit step. Thus, in order for a machine to read the y-axis scale it must find the start and end point of a unit step and then read the text alongside it to recover the axis scale.

Identifying and separating text and graphic component regions in an image is one task, another is to determine the roles for extracted components. In a bar chart a colored box can either be a bar component that represents a data value, which will give it the role: "bar", or it might be a box belonging to the legend which, in this case, will have role: "legend". Similarly, a text region can either be a label for a data unit or the title of the entire chart, which are two different roles a text component may take. Image processing and analysis rely on feature extraction for the pixels and apply classification methods to determine the class of the component. Feature based classification in images has shown to be highly effective due to the nature of image features. Text-centric features have also shown effective in the classification of text component roles. In this work we aim to take advantage of both approaches to produce better results in automatic text and graphics role labeling. We describe the results of our approach on over 200 bar charts extracted from over 1,000 scientific publications.

## Related Work

Many applications have motivated the study of data extraction from scientific charts in various contexts. One application of chart data extraction is in assisting the visually impaired to read charts as in (Demir et al. 2010). They develop an interactive chart summarizing tool called *SIGHT* that reads graphical charts using extraction techniques further described in (Chester and Elzer 2005). Chart data extraction is also the basis for systems designed for extracting data from diagrams published by statistical agencies. The iGraph-Lite system proposed in (Ferres et al. 2007) is an example for this type of applications. A large amount of studies on chart data extraction are found in image and document search applications. Text strings and numerical values contained in charts and tables in scholarly documents are used along with their metadata, captions, document text mentions to enhance query and ranking results. Examples of works on this front include (Liu et al. 2007), (Kataria et al. 2008), (Lu et al. 2009), (Tuarob et al. 2013), (Fang et al. 2012), and (Al-Zaidy, Choudhury, and Giles 2016).

The problem of data extraction from charts involves two

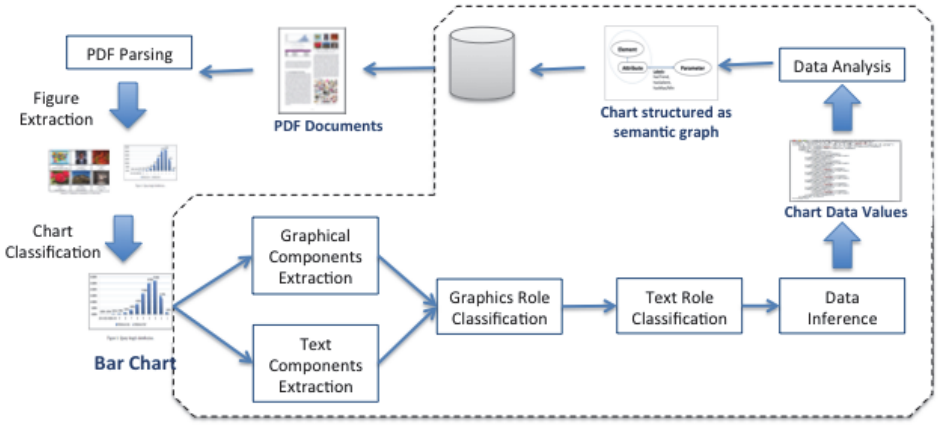


Figure 1: System Architecture

main steps. The first is graphic component and text region identification and extraction. The next step is identifying the roles that each of the extracted components play in the chart. The second step is where the actual machine understanding of basic chart elements occurs. In order to recover meaningful information from the chart each component must be assigned a label, e.g. x-axis, bar component, axis label value, etc. Various approaches have been proposed for each of these steps. In (Savva et al. 2011) the graphic components are extracted using connected component labeling, and the role recognition of the the components is done using heuristics chosen based on the properties of the specific chart. As for the text region identification and role identification, they rely on manual labeling through a custom image labeling interface. Fully automated systems have been developed as well, examples include (Lu et al. 2009), which is 2D-line plot data extraction system. In a more recent study (Al-Zaidy and Giles 2015), a method is proposed to automate the extraction of data from bar charts. Both graphic components and text regions are extracted using connected component analysis. As for components role identification, heuristics similar to those used in (Savva et al. 2011) are used. Some studies on the other hand focus on the text region extraction and role labeling. In (Huang and Tan 2007), the text regions are extracted using image processing techniques and then are passed to a classifier that will select one of 11 roles to label the texts. The method used in (Chen, Cafarella, and Adar 2011) applies a multiphase method to assign a role to the texts from 8 possible roles. The phases alternate between a features generation phase and a classification phase to assign the role labels. The first feature generation phase is based on text-centric features, the second is a location-based grouping process to generate location features. Other studies utilize the extracted data to generate summaries of the charts such as (Demir, Carberry, and McCoy 2008).

## Chart Component Extraction

In order to extract the data values represented by a bar chart, the chart text and graphics components must be identified and their locations retrieved. This section covers the techniques and methods used for component extraction. The extraction process follows the pipeline illustrated in figure 1. The first step is to extract two types of components, the graphic components and text components. Graphic components include: x-axis and y-axis, chart legend, and bars. By text component we refer to all text labels found in the chart area. The graphic components are extracted using the method described in (Al-Zaidy and Giles 2015).

### Graphic Component Extraction

The method applied to extract graphic components follows a 3 step process:

**Image Color Space Conversion** The image color is converted from RGB space to the L*ab space. This step is performed to provide higher accuracies in distinguishing a wider spectrum of colors than can be achieved using RGB color space.

**Grid Line Removal** Hough transforms are used to identify horizontal lines that are not long enough to be the x-axis. This step is useful in eliminating background boxes.

**Color Connected Component Labeling** This is performed over the image in L*ab color. The distance function used to measure the difference between pixel colors is the deltaE95 function. This step returns, for each of the connected components, the bounding box coordinates, area, and color.

### Text Component Extraction

The text region extraction steps are as follows:

**Binarization** The image is converted to binary format.

**B/W Connected Component Labeling** A pass of the black and white connected component labeling is run on the binary image. The returned components are then filtered to remove large components that are not candidates for being text characters. This step returns regions for single characters in the image.

**Isotropic Dilation** Each extracted character is dilated, using a small enough window, only to make each character

attach to the character next to it if they are in the same word. The window size is chosen to be small enough to not reach a standard space size.

**2nd Pass Connected Component Labeling** Since the previous dilation is performed, each word is now a single component rather than each character. Thus, the output of this pass is the coordinates of the bounding box of each text word in the chart image.

**Text Recognition-OCR** The patch of the image where the text region is located is passed to an OCR tool that returns the text content of that part of the image in string format.

The output of the text region extraction step is the location of the text region and the text string value. This is then passed to the text role labeling step.

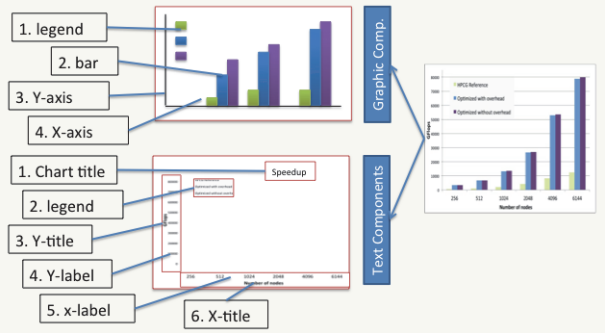


Figure 2: Role labels for Graphic and Text Components.

## Component Role Classification

From the previous section, the chart has been reduced to a set of image components. The information we have about these components at this point is the location of their bounding box in the image, whether they are text or graphics, color for graphic components and text string for text components. The next step to obtain the data is to correlate the text descriptors and numerical values with the graphic component whose value they are describing. We refer to this step as role labeling or role classification. Figure 2 shows the different role labels defined by our system for both graphic and text components. (Sample chart is from (Zhang et al. 2014)). In this section we describe the methods used for role classification.

### Graphic Component Classification

For bar charts, we define four role labels, bar, legend, x-axis, and y-axis. The components are subject to a noise cleaning step to remove false positives. We define 10 features for the graphic components. The graphic features that are selected for these components are:

**Shape** We are interested in rectangular/square shapes for bars and legends. The extent of a shape is a value between [0,1] that measures how much a shape fills its bounding box. The higher this value, the more likely it is a box shape. This feature is for both bars and legends, as both are box shaped.

**Color** The color of the centroid of the shape is used for this feature. The value is a binary true if the shape is either black or white and false otherwise. This feature determines background boxes.

**Distance to X-axis** The value of this feature is a the distance between bottom edge of the bounding box and the x-axis normalized over the image size. For bars, the distance will be very small or zero. For legends it can vary.

**Relative position to Y-axis** This takes a binary value of true if the shape is to the right of the y-axis. Bars are always to the right of the y-axis. Legends may or may not be.

**Relative shape width** The relative width of the shape to mean width of all graphic components in the image. Legends are usually smaller or wider than bars. Also, small sized bars may have their total area similar to that of a legend box, but the difference is that it's width is similar to those of other bars.

**Centricity** This is the normalized distance of the component from each bisector of the image. This is for legends as they are typically closer to the borders of the image.

**Height-width ratio** The ratio of height to width of the shape's bounding box. This feature is for legends since they are more commonly square-shaped.

**Type of closest component** This feature takes a binary value of true if the closest component to the graphics is a text component. This feature is also for legends since they are typically closest to texts.

We use the features to classify the role of the graphic component into either bar or legend. The classification results are tested using the c4.8 and random forest methods to compare accuracies.

### Text Region Classification

For the text components we extract the features from the graphic properties of the text regions. Additionally, we also have one feature that is related to the actual text value of the text region. The classification is to determine what graphic component the text is describing. The approach we use is a combination of two approaches. We propose three types of features for the text role classification: location/position centric, text-centric and graphical features. The location-based features are an adaptation of the classification method proposed by (Huang and Tan 2007) where they specify 5 features that are extracted from the texts to classify the regions. The text centric features are adopted from the (Chen, Cafarella, and Adar 2011) method. Graphical features are similar to those used in the heuristics-based methods of (Al-Zaidy and Giles 2015) and (Savva et al. 2011). The texts are classified into one of 7 roles: y-axis label, y-axis name, x-axis label, x-axis name, legend name, chart title, and other. The location/position based features mentioned are the following:

**Distance to closest graphics** The distance between the text region and the closest graphic component to it, is deter-

Table 1: Graphics Components Role Classification Accuracy using C4.8 and RF Classifiers

| Role | Multiclass | | | | | | Binary | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Precision | | Recall | | F1 | | Precision | | Recall | | F1 | |
| | RF | C4.8 | RF | C4.8 | RF | C4.8 | RF | C4.8 | RF | C4.8 | RF | C4.8 |
| bar | 98.4 | 97.5 | 97.4 | 96.8 | 97.9 | 97.2 | 98.5 | 97.3 | 97.2 | 96.8 | 97.9 | 97 |
| legend | 92.5 | 89.9 | 93.5 | 88.7 | 93 | 89.3 | 93.6 | 90 | 90.4 | 86.3 | 92 | 88.1 |
| Other | 94.7 | 92.9 | 96.3 | 94.3 | 95.5 | 93.6 | - | - | - | - | - | - |

mined as following:

$$h(T_i, G_j) = \min_{t \in T_i} \min_{g \in G_j} d(t, g)$$

where, $T_i$ and $G_i$ are the text and graphic components respectively, and $t$ and $g$ are the sets of points on the perimeter of each component. This is a measure of the closest Euclidian distance, $d$, between the two closest points on the perimeter of each component.

**Relative position of the closest graphics** This is determined by the angle between a text block and it's nearest graphic component. The positions take values: top, bottom, right, left. top right, top left, bottom left, bottom right.

**Position to Y-axis** This is a binary value that is true if the text is to the left of the y-axis.

**Position to X-axis** Also, a binary value. The value is true if the text region is located below the x-axis.

**Centricity** Horizontal and vertical centricity are calculated as the normalized distances between the centroid of the text region and both the vertical and horizontal bisections of the image.

The text-centric features are the following:

**Capitalization** Percentage of characters in the word that are capitalized.

**String Length** Normalized number of words in the text region. Axes titles and chart title are more likely to contain more than one word.

**isNumeric** Whether the text has a nominal or numeric value. The y-axis labels are always numbers.

The following are the graphical based features:

**Orientation** Vertical or horizontal orientations of the text box as a binary value. This feature is for the y-axis title, which typically appears in charts in a vertical layout.

**Closest Graphic** This feature stores the class of the graphic component closest to the text box. This requires that the graphic component classification has already been completed.

These features are then used to train both a c4.8 and a random forest classifier to determine the role of the text label.

## Semantic Analysis

The purpose of chart component extraction is to convert the representation of information contained in a chart image from graphical to a text representation. Depending on the application requiring the chart information, the needed information representation can vary. In this section we describe how we produce machine-usable information from the data extracted from bar chart images.

### Data Inference

In order to recover the data in raw numerical form, we use the method in (Al-Zaidy, Choudhury, and Giles 2016). Once the graphical and textual components are determined, the values of the data are recovered from the bars by multiplying the height of the bars (in pixels) by the y-scale value-to-pixel ratio. This ratio is computed by dividing the difference between two y-scale labels over their vertical distance in pixels. This does not recover logarithmic scale y-axes values. The remaining values from the chart are extracted by the classification results. By the end of this step the data is as if it has been recovered to the original data table that was used to generate the chart. Each legend is a column head and the x-axis labels are the row names. The next steps process the data values to generate a semantically-enhanced representation of the data.

### Semantic Graph Representation

To enable the construction of the semantic graph representation of the chart, we use four main semantic values: trend, maximum and minimum, x-axis is a timeline, x-axis is ordinal. We extract this additional information by analyzing the extracted data values. The nodes in the graph represent the x and y axes titles. The edge between them represents the extracted semantics above. The chart is then stored as these data triples and can be easily employed in applications requiring this structure.

### Chart Synopsis

A plain-text description of the charts is obtained as an addition to the previous semantics to enrich the metadata of the charts. The description is generated using the algorithm proposed in (Al-Zaidy, Choudhury, and Giles 2016). The values obtained in the semantic graph are used to construct an English text summary of the chart's main message. The assumption is that each chart can be described by certain messages that the designer intended to illustrate graphically through the chart. The message can be to simply present the rank of each data value or to display a trend among the data values. For further details on the messages charts present, the reader is encouraged to read (Elzer et al. 2006). The synopsis comprises of a sentence or two based on the features

Table 2: Text Role Classification Accuracy using C4.8 and RF Classifiers

| Role | Multiclass | | | | | | Binary | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Precision | | Recall | | F1 | | Precision | | Recall | | F1 | |
| | RF | C4.8 | RF | C4.8 | RF | C4.8 | RF | C4.8 | RF | C4.8 | RF | C4.8 |
| Y-title | 96.8 | 93.4 | 95.8 | 91.6 | 96.3 | 92.5 | 96.9 | 94.3 | 93.8 | 90.8 | 95.3 | 92.6 |
| Y-label | 96.4 | 95.8 | 99.1 | 98 | 97.7 | 96.9 | 97.4 | 95.6 | 98.6 | 97.6 | 98 | 96.6 |
| Legend | 87.5 | 81.2 | 90.9 | 85.9 | 89.2 | 83.5 | 93 | 84.6 | 85.1 | 82.8 | 88.9 | 83.7 |
| X-labels | 95.1 | 93.8 | 95.9 | 95.1 | 95.5 | 94.4 | 96 | 94.8 | 95.2 | 95 | 95.6 | 94.9 |
| X-title | 85.7 | 84.7 | 82.4 | 77.8 | 84 | 81.1 | 89.3 | 80.1 | 76.8 | 76.4 | 82.6 | 78.2 |
| Chart-title | 88.2 | 84.7 | 86.5 | 83.1 | 87.4 | 83.9 | 92.2 | 85.4 | 81.9 | 80.8 | 86.8 | 83 |
| Other | 91.9 | 83.5 | 80.4 | 74.4 | 85.8 | 78.7 | - | - | - | - | - | - |

Table 3: Data Extraction accuracy for Rule-Based vs. Machine Learning Role Labeling -1

| | Precision | | Recall | |
|---|---|---|---|---|
| | RB | ML | RB | ML |
| Data Values | 89.65 | 98.14 | 93.48 | 78.06 |
| X-labels | 40.03 | 91.93 | 84.32 | 79.26 |
| Legend | - | 59.26 | - | 43.75 |

Table 4: Data Extraction accuracy for Rule-Based vs. Machine Learning Role Labeling -2

| | Accuracy % | |
|---|---|---|
| | RB | ML |
| X-title | 75.76 | 90.91 |
| Y-title | 80.95 | 95.24 |
| Y-scale | 63.27 | 78 |
| Chart-title | - | 99 |

found in the data using the analysis described in the previous section. If the data displays a trend, either increasing or decreasing, we check if the x-axis is ordinal or a time series and then construct a sentence stating the data trend along with the trend in the ordinal values, or the timeline. Then select that as the description. If no trend exists, we check for maximum and minimum values in the data and display those as the description sentence. If neither a trend or a max/min value exists, we simply describe the rank of each data value.

## Evaluation

In this section we describe the evaluation of our method on a set of 213 bar charts extracted from over 1000 PDF files. The tool used to extract the charts is PDFFigures (Clark and Divvala 2015), (Clark and Divvala 2016). Further details on the chart extraction from PDFs are found in (Al-Zaidy, Choudhury, and Giles 2016). The charts were selected randomly from the PDF documents, however, to obtain a more diverse set of charts, we reject a chart if the set already contained a chart extracted from that PDF. Most charts from the same document have the same layout. The PDF documents are articles published in top Computer Science conferences.

### Graphics Role Labeling

The first part is to evaluate the classification of the graphic components. Table 1 shows the results for multi-class and binary classification using the c4.8 classifier and the random forest classifier with 10-fold cross validation. As shown the random forest produced better results and higher accuracies. It is noted that the bar accuracies are higher and that is due to the fact that bars have more consistent layout in the chart as opposed to legend boxes. Also, the removal of small size components during the connected components labeling step, can cause some legend boxes to be filtered out of the image all together.

### Text Role Labeling

The extracted text regions are classified into one of 7 roles and accuracies are reported for both the C4.8 classifier and random forest. Table 2 shows the precision and recall for each of the roles (the first column) obtained by each of the classifiers applied as multi-class and binary. As noted, the random forest classification scheme provides higher accuracies for the text role labeling. The results are highest for the y-scale values, which is a very important field, since the extraction of the y-scale, and consequently the data values, rely on the correct extraction of these values. The title of the x-axis has lowest accuracy, and that is due to the fact that many charts in our data set did not contain an x-axis title.

### Data Extraction Accuracy

The quality of the role labeling phase naturally affects the accuracy of the final data values we recover from the chart. To evaluate the effectiveness of our machine-learning approach where we train the classifiers to label the roles of the components, we compare the data extraction accuracies using our methods with those obtained by a rule-based approach proposed in (Al-Zaidy and Giles 2015). The data set used for this evaluation is a set of 50 charts different than those used in training our classifier. Tables 3 and 4 show the precision and recall for the data extraction. As expected the precision achieved using the ML approach is higher for most of the values. Legends had low values for both precision and recall because the evaluation did not tolerate any type of error in the recovery. If the entire legend map was not recovered fully it was considered a miss. Also, in some cases when the legend texts were correctly labeled as being legend text, they were associated with an incorrect legend

box, which is considered erroneous in our evaluation. Chart titles have notably high accuracies, this is due to firstly, most charts had the title at the top. This is a general observation about the data set. Additionally, for the data set used in this experiment only 11 out of the 50 charts contained a title. Which is also noted as common in Computer Science papers as not many of the charts contain a title.

## Conclusion

In this paper we present a machine learning approach to determine the roles of both graphical and textual components in bar chart images. The role classification is an essential step in the chart data extraction process. Thus, a machine learning approach for component role-labeling is proposed to improve over existing rule-based methods. Rule based approaches, although produced high recall, were rather low in precision. Additionally, the texts in charts have specific layout features that make them good candidates for a machine learning approach. To evaluate the results of the classification we compare accuracies obtained by two decision tree based classifiers. We also compare the results of the classification using multi-class and binary classification. Moreover, we compare the final data accuracies using an existing rule-based method against our proposed machine learning approach. The evaluation shows that for precision we get the best results using the random forest binary classification. The recall is highest we when use the results of the multi-class random forest.

Current and future extensions to this work includes the deployment of the extraction approach into a search tool that will be able to index the extracted semantics and use them for more complex user queries. Additionally, further meta data such as chart captions and mentions in the text can be used to enhance the synopsis as well as the semantics. Also, since the system is also easily integrated in web applications, web services for automatic reading of charts in scholarly articles is one future extension of this work.

## References


Al-Zaidy, R. A., and Giles, C. L. 2015. Automatic extraction of data from bar charts. In *Proceedings of the 8th International Conference on Knowledge Capture*, 30. ACM.

Al-Zaidy, R. A.; Choudhury, S.; and Giles, C. L. 2016. Automatic summary generation for scientific data charts. In *AAAI 2016 Workshop on Scholarly Big Data*.

Chen, S. Z.; Cafarella, M. J.; and Adar, E. 2011. Searching for statistical diagrams. *Frontiers of Engineering, National Academy of Engineering* 69–78.

Chester, D., and Elzer, S. 2005. Getting computers to see information graphics so users do not have to. In *Foundations of Intelligent Systems*. Springer. 660–668.

Clark, C., and Divvala, S. 2015. Looking beyond text: Extracting figures, tables, and captions from computer science paper. In *AAAI Workshop on Scholarly Big Data*.

Clark, C., and Divvala, S. 2016. Pdffigures 2.0: Mining figures from research papers. In *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries*, JCDL '16, 143–152. New York, NY, USA: ACM.

Demir, S.; Oliver, D.; Schwartz, E.; Elzer, S.; Carberry, S.; and McCoy, K. F. 2010. Interactive sight into information graphics. In *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*, 16. ACM.

Demir, S.; Carberry, S.; and McCoy, K. F. 2008. Generating textual summaries of bar charts. In *Proceedings of the Fifth International Natural Language Generation Conference*, 7–15. Association for Computational Linguistics.

Elzer, S.; Green, N.; Carberry, S.; and Hoffman, J. 2006. A model of perceptual task effort for bar charts and its role in recognizing intention. *User Modeling and User-Adapted Interaction* 16(1):1–30.

Fang, J.; Mitra, P.; Tang, Z.; and Giles, C. L. 2012. Table header detection and classification. In *AAAI*.

Ferres, L.; Verkhogliad, P.; Lindgaard, G.; Boucher, L.; Chretien, A.; and Lachance, M. 2007. Improving accessibility to statistical graphs: the igraph-lite system. In *Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility*, 67–74. ACM.

Huang, W., and Tan, C. L. 2007. A system for understanding imaged infographics and its applications. In *Proceedings of the 2007 ACM symposium on Document engineering*, 9–18. ACM.

Kataria, S.; Browuer, W.; Mitra, P.; and Giles, C. L. 2008. Automatic extraction of data points and text blocks from 2-dimensional plots in digital documents. In *AAAI*, volume 8, 1169–1174.

Liu, Y.; Bai, K.; Mitra, P.; and Giles, C. L. 2007. Tablerank: A ranking algorithm for table search and retrieval. In *Proceedings of the National Conference on Artificial Intelligence*, volume 22, 317. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

Lu, X.; Kataria, S.; Brouwer, W. J.; Wang, J. Z.; Mitra, P.; and Giles, C. L. 2009. Automated analysis of images in documents for intelligent document search. *International Journal on Document Analysis and Recognition (IJDAR)* 12(2):65–81.

Savva, M.; Kong, N.; Chhajta, A.; Fei-Fei, L.; Agrawala, M.; and Heer, J. 2011. Revision: Automated classification, analysis and redesign of chart images. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 393–402. ACM.

Tuarob, S.; Bhatia, S.; Mitra, P.; and Giles, C. L. 2013. Automatic detection of pseudocodes in scholarly documents using machine learning. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, 738–742. IEEE.

Zhang, X.; Yang, C.; Liu, F.; Liu, Y.; and Lu, Y. 2014. Optimizing and scaling hpcg on tianhe-2: early experience. In *International Conference on Algorithms and Architectures for Parallel Processing*, 28–41. Springer.