of the figure to be returned (for example, "Table 1" or "Figure 1") so that mentions of that figure can be mined from the rest of the text. Previous work on figure extraction has focused on documents from the biomedical (Lopez and others 2011), chemistry (Choudhury and others 2013) or high energy physics (Praczyk and Nogueras-Iso 2013) literature. In this work we study the problem of figure extraction for the domain of computer science papers. Towards this end we release a new dataset, annotated with ground truth bounding boxes, for this domain.

2 Background

At a high level, PDF files can be viewed as a series of operators which draw individual elements of the document. These operators can include text operators, which draw characters at particular coordinates with particular fonts and styles, vector graphic operators that draw lines or shapes, or image operators that draw embedded image files stored internally within the PDF. Within a document, a figure can either be encoded as an embedded image, or a number of embedded images arranged side-by-side, or as a combination of vector graphics and text, or a mix of all three. In more recent computer science documents most plots and flow charts are composed of vector graphics. While many "off-the-shelf" tools exist that can extract embedded images from PDFs, such as PDFBox[3] or Poppler (Poppler 2014), these tools do not extract vector graphic based images. Additionally, such tools leave the problem of caption association unsolved and are liable to extract unwanted images such as logos or mathematical symbols. Image segmentation tools such as the ones found in Tesseract (Smith 2007) or Leptonica[4] are also inadequate. Such tools frequently misclassify regions of the document that contain very text heavy figures as being text, or misclassify bits of text as being images if that text is near graphical elements or contains mathematical symbols.

Figure extraction has been a topic of recent research interest. In (Choudhury and others 2013) captions were extracted using regular expressions followed by classification using font and textual features. We build upon this idea by additionally leveraging a consistency assumption, that all figures and tables within a paper will be labelled with the same style, to prune out false positives. Their system also extracts figures by detecting images embedded in the PDF, however it does not handle vector graphics. Work done by (Lopez and others 2011) identified captions through regular expression followed by clustering the resulting figure mentions. Graphical elements were then clustered by parsing the PDF files and analyzing the individual operators to find figures. While their approach reached 96% accuracy on articles from the biomedical domain their method does not handle figures composed of combinations of image operators and text operators, which is prevalent in the domain of computer science papers. Work by (Praczyk and Nogueras-Iso 2013) similarly detects figures and captions by merging nearby graphical and textual elements while heuristically filtering out irrelevant elements found in the document. Their approach is similar to ours in that an attempt is made to classify text as being body text or as part of a figure to allow the extraction of text heavy figures. However, their algorithm requires making the

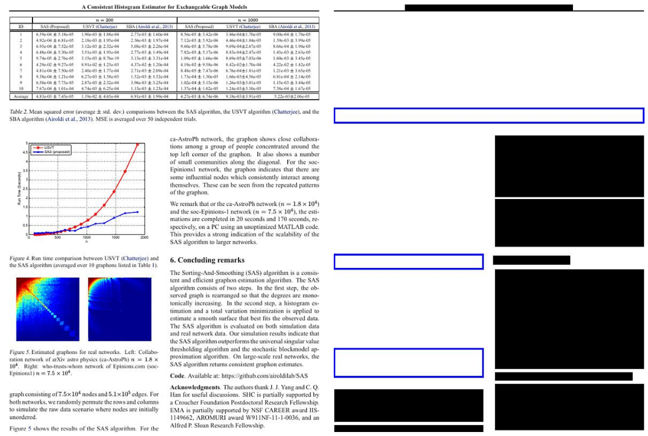

Figure 1: A scholarly document (left, page from (Chan and Airoldi 2014)), and the same document with body text masked with filled boxes, captions masked with empty boxes, and tables and figures removed (right). By examining the right image it is easy to guess which regions of the page each caption refers to, even without knowing the text, what the captions say, and anything about the graphical elements on the page.

assumption that figures, while possibly containing text, contain at least some non-trivial graphical elements to 'seed' clusters of elements that compose figures. Our work avoids making this assumption and can thus handle a greater variety of figures, as well as generalize easily to tables. Our work also addresses the problems the can arise when captions are adjacent to multiple figures or multiple figures are adjacent to each other.

Extracting tables has also been addressed as a separate task of great interest in the information extraction community. Our work is not directly comparable since the system presented here locates tables but does not attempt to organize their text into cells in order to fully reconstruct them. Nevertheless locating tables within documents is a non-trivial part of this research problem. Most approaches use carefully built heuristics based on detecting columns of text, vertical and horizontal lines, or white space. A survey can be found at (Zanibbi, Blostein, and Cordy 2004) or in the results of a recent competition (Khusro, Latif, and Ullah 2014). Our work shows that table detection can be completed without relying on hand crafted templates or detailed table detection heuristics by exploiting more general assumptions about the format of the documents being parsed.

A number of search engines for scholarly articles have made attempts to integrate table and figure information. CiteSeerX (Wu and others 2014) extracts and indexes tables in order to allow users to search them, but does not handle figures. The Yale Image Finder (Xu, McCusker, and Krauthammer 2008) allows users to search a large database of figures, but does not automatically extract these figures from documents.

3 Approach

Our algorithm leverages a simple but important observation that if a region of the page does not contain body text and is adjacent to a caption, it is very likely to contain the figure

[3] http://pdfbox.apache.org/

[4] http://www.leptonica.com/