Universität Passau
Fakultät für Informatik und Mathematik

CLASSIFICATION OF VISUALIZATIONS IN SCIENTIFIC LITERATURE

Masterarbeit zur Erlangung des akademischen Grades
Master of Science (M.Sc.)

Lehrstuhl für Intelligent Systems und Lehrstuhl für Data Science
der Fakultät für Informatik und Mathematik
der Universität Passau

Name:	Arnold Azeem
Matrikelnummer:	79176
Fachbereich:	Informatik
Studiengang:	Master Informatik
Erstprüfer:	Prof. Dr. Christin Siefert
Zweitprüfer:	Prof. Dr. Michael Granitzer
Date:	26th April 2018

Contents

List of Figures

List of Tables

ABSTRACT

Distinct visualization techniques are used in scientific research publications to summarize large amount of data and also represent a variety of data. These visualizations help to communicate complex information and support the arguments presented in the paper in a easy to understand and follow way. These figures tend to reveal trends,patterns or relations that might otherwise be difficult to grasp using only text. In this context, classifying these visualizations is really relevant since there is a variety of visualizations and each one will have a different approach to processing it, example is extracting the raw data from it.

INTRODUCTION

A picture tells a thousand words even though a cliché stands to be very true especially when it comes to presenting complex findings in scientific research publications. The importance of these figures in papers cannot be undermined since they provide a way to easily interpret,find patterns and relations in the data which would have otherwise been more complex relying on only textual data. All though extracting data from a plot manually is relatively easier, doing the same task automatically requires each type of plot to be processed specifically. In this work we present a way to classify each plot effectively, since that the first step before further processing of a plot is possible.

MOTIVATION

Complex data is better explained in scientific papers with the aid of visualizations. These plots present complex data in an easy to understand way compared to textual representation. The data which these visualizations contain when extracted play an important role in events where another researcher wants to verify the work of the publisher, this data can also be used to develop other visualizations in situations where the paper needs to be presented to a different audience with a different background as opposed to the audience which the visualizations were created for, Also when comparing two plots the raw data helps make a better decision than just the figures. Since each plot will be processed differently to extract the raw data, it very relevant that we can distinguish one plot from another and this is the main aim of this thesis.

OBJECTIVE

The purpose of this thesis is to answer the question:

> HOW WELL CAN WE CLASSIFY THE DIFFERENT TYPES OF PLOTS IN SCIENTIFIC LITERATURE.

In this work we focus on only four plots. These plots are scatter plots,bar charts,line charts and Box plots. The diagram below shows the vision of this work, The first part of the diagram involves extracting or obtaining the four different types of plots mentioned earlier,after which we then label our plots and train a neural network model to be able to classify with high accuracy any of the four plots if shown to our model, then finally the raw data can be extracted from the detected plot. But this work mainly focuses on the red dotted lines shown below in the diagram which which is getting the plots, labeling them, training the model and classifying the plots.

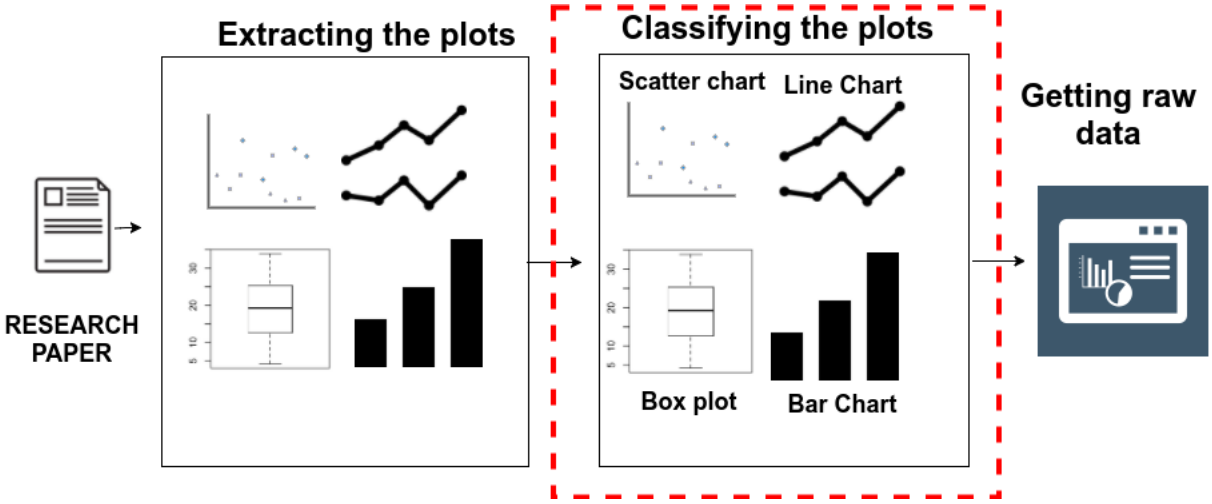


Figure 2.1: vison of the thesis

RELATED WORK

Our work is inspired by Architecture proposal for data extraction of chart images using Convolutional Neural Network(2017) [5], In this paper, they propose a way to gain the wealth of information contained in different visualization techniques. The paper talks about two main stages of accomplishing this task. Firstly, classification of the charts is done since it allows a different variety of chart to be detected automatically allowing the next step, which is the extraction of data from the classified plots. The paper, however, focuses on the first step, classification of charts. In this paper, a Convolutional Neural Network is used for the classification task. The Convolutional neural network encapsulates the characterization and classification processes during its learning process, unlike other techniques. The dataset used for this task were searched for and downloaded from Google image search. Table 3.1 shows the chart types which were collected and the number of train and test sets which the respective charts were divided into.

Chart Type	Test	Train
Area Chart	50	555
Bar Chart	50	657
Line Chart	50	489
Map	50	476
Pareto Chart	50	261
Pie Chart	50	361
Radar Chart	50	454
Scatter Chart	50	552
Table	44	236
Venn Diagram	48	304
Total	498	4345

Table 3.1: Number of Train and Test Dataset collected

For the classification, a variant of convolutional neural network called LeNet-based CNN model is used. The model was implemented using [1], LeNet-based CNN has an architecture which is comprised of 3 convolutional layers, followed by a fully connected layer. The model is trained in a way that the dataset is divided into mini-batches, samples of fixed sizes(100) are selected and fed into the CNN, as a result of this process the model becomes robust since it learns to generalize from the different min-batches which are fed into the model. Also, all the images are converted to JPG and resized to 224x224x3, that is, 224 pixels of height, 224 of width and 3 layers of output. The other parameters used were 1000 epochs and a learning rate of 0.003. The accuracy at the end of the training process was 70%.

Dataset

In this section, the various datasets used in plotting are described. For each language a different set of CSV files are used for generating the plots. This is done to generate more diverse plots.

Dataset for Matlab

The Data used for creating the plots in Matlab were randomly chosen from Project Dataset [1], a free CSV data repository, DatPlot [2] and Plotly CSV repository in github [3]. The datasets are multidimensional and compiled from normal day to day activities like dating, what makes people happy etc, and objects like cameras and cars. On the average the datasets used contain about 500 instance and 5 different columns. The biggest dataset is called Speed dating data. It is made up of over 8,000 observations of answers to survey questions about how people rate themselves and how they rate others on several dimensions. The smallest dataset used has 33 instances and 12 columns. It contains information about cars. The number of gears and speed, just to name a few attributes.

Dataset for R

For the plots in R, 13 random CSV files where downloaded from an archive of datasets distributed with R called Rdatasets [4]. Rdatasets is a collection of dataset distributed with R. On the average there are 80 instances and 5 columns in each dataset. The biggest CSV file is the Australian athletes dataset. Its made of 203 instances and 14 columns and contains attributes like sex,height,weight and sports. The smallest dataset is the Canadian Women's Labour-Force Participation. This dataset has 30 rows and 7 columns. It con-

[1] Tensorflow

tains information like average wages of women, percent of adult women in the workforce etc.

Dataset for PYTHON

The data used for creating the plots in Python were 15 randomly seleted csv files also from Rdatasets [4]. The biggest dataset among the 15 is the Monoclonal gammapothy data, it contains natural history patients with monoclonal gammapothy of undetermined significance. The dataset is made up of 1384 observations with 10 columns, it has attributes like age, sex, time of death and last contact in months. On the average each dataset contains about 200 instances and 7 columns of multi-dimensional data. The smallest dataset however contains only 33 instances with 11 columns and is called the Nuclear Power Station Construction Data.The data relate to the construction of 32 light water reactor (LWR) plants constructed in the U.S.A in the late 1960's and early 1970's.

Dataset for Java

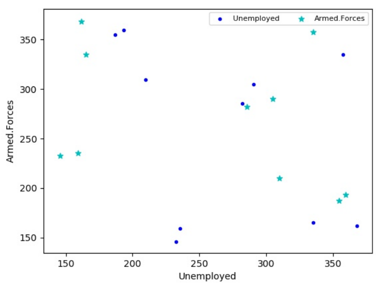
For the plots created in java, I used the dataset made available by Plotly [3], a github repository of CSV datasets used in the Plotly API examples. 14 random CSV files were downloaded, the biggest file has 1002 instances and 9 columns, and on the average each file contains about 100 instances and 9 columns. The smallest file however is made of 33 instances and 12 columns called the mtcars file. It contain information about a variety of different car models like the number of gears, speed etc. The table 3.2 contains the names of all CSV files that were used in the different languages with the different plotting programs.

DUMMY DATA			
PYTHON	MATLAB	R LANGUAGE	JAVA
3d_line_sample_data.csv LightFordwardFlapStall.csv line_3d_dataset.csv longley.csv loti.csv lung.csv nuclear.csv timeseries.csv USJudgeRatings WVSCulturalMap.csv wind_rose.csv volcano.csv uspop2.csvm	Camera.csv Cars.csv CausesOfDeath-France.csv Cereal.csv happiness.csv TestData1.csv TestData2.csv mpg.csv okcupid-religion.csv spectral.csv stockdata.csv subplots.csv	ais.csv Angell.csv Baumann.csv Bfox.csv cane.csv carprice.csv Chirot.csv Davis.csv Ericksen.csv Florida.csv Highway1.csv Pottery.csv Prestige.csv salinity.csv urine.csv	3d-line-plot.csv 3d-scatter.csv 2011_flight_paths.csv 2011_us_exports.csv auto-mpg.csv candlestick_dataset.csv finance-charts-apple.csv globe_contours.csv hobbs-pearson-trials.csv motor_trend_tests.csv nz_weather.csv volcano.csv iris.csv mtcars.csv

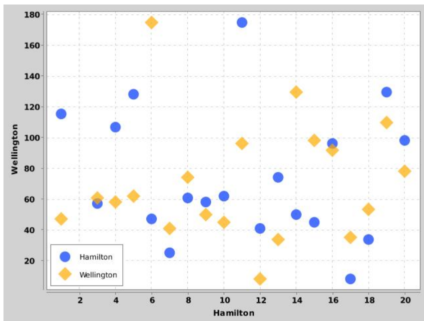
Table 3.2: Names of CSV files used in each language

CREATING PLOTS

The inspiration for creating a variety of plots to capture all type of plots used in scientific papers was gotten by inspecting the dataset of Architecture proposal for data extraction of chart images using Convolutional Neural Network paper [5] and Viziometrics: Analyzing visual information in the scientific literature [6] dataset. Scripts in various languages were written to handle the plotting and labeling process automatically. All datasets for a particular plot (example scatter plot for python) are put into one folder. The scripts reads each CSV file column by column while creating the plots. The tables below describe how the plots where created in each language, the plotting libraries used, the variants of a particular plot come under the type column, the parameter column describes parameters that were changed and finally the number of plots created were also added. The images below the tables are sample images that exist in our dataset of created plots for each language.

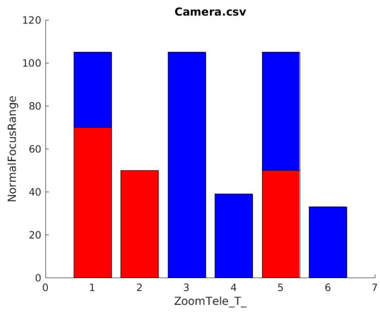


(a) created with Matplotlib

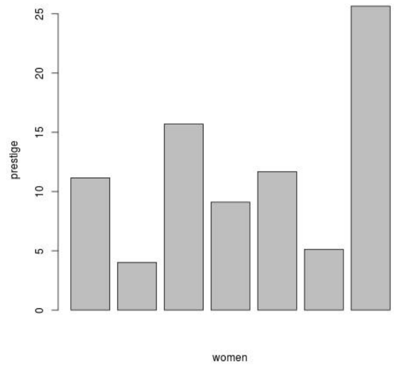


(b) created with Jfreechart library

Figure 4.1: Example Scatter plots

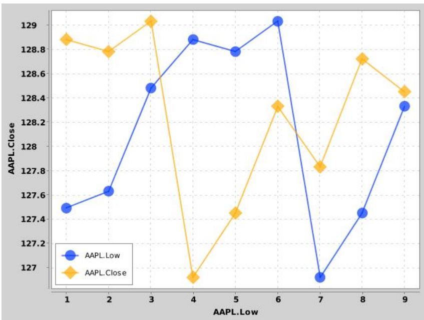


(a) created in Matlab

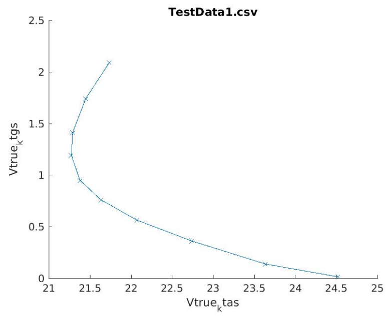


(b) created in R

Figure 4.2: Example Bar Charts

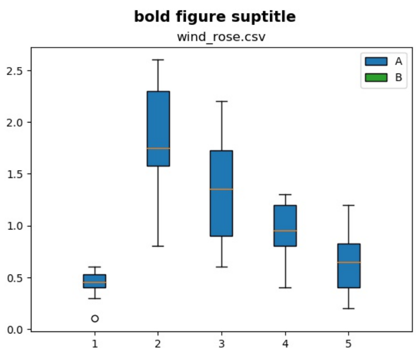


(a) created in Java

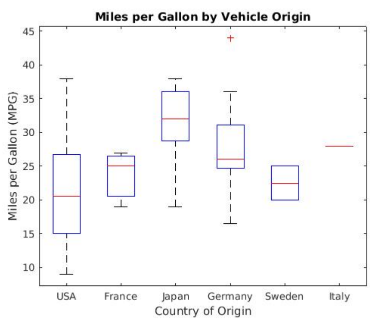


(b) created in Matlab

Figure 4.3: Example Line Charts



(a) created in python



(b) created in Matlab

Figure 4.4: Example Box Plots

SCATTER PLOTS				
Language	Library	Parameters	Number of plots	Types(scatter with)
Python	Matplotlib v2.1.2 Plotly v2.5.1 Seaborn v0.8.1	MarkerStyle [’o’, ’*’, ’.’, ’+’,’x’]	1020	Unique markers, With legends
MATlAB	Default	MarkerStyle [’o’, ’*’, ’.’, ’+’,’x’,’s’]	1044	
R	Default,Plotly R Library ggplot2	MarkerStyle [’o’, ’*’, ’.’, ’+’,’x’,’s’]	1644	
JAVA	XChart 3.5.1,jfreechart:1.0.192	setMarkerSize (16) LegendPosition	1644	

BAR CHARTS				
Language	Library	Parameters	Number of plots	Types(bar)
Python	Matplotlib v2.1.2 Plotly v2.5.1 Seaborn v0.8.1		1000	Horizontal and Vertical, Stacked, Grouped bar charts
MATlAB	Default	Width of bar	1000	
R	Default,Plotly R Library ggplot2		1144	
JAVA	XChart 3.5.1 jfreechart:1.0.192 javafx.scene	setMarkerSize (16) LegendPosition	1144	

LINE CHARTS				
Language	Library	Parameters	Number of plots	Types(Line with)
Python	Matplotlib v2.1.2 Plotly v2.5.1 Seaborn v0.8.1	Linestyle [’-’, ’–’, ’-.’, ’:’]	1000	Markers, Multiple Lines
MATlAB	Default	MarkerStyle [’o’, ’*’, ’.’, ’+’,’x’,’s’]	1000	
R	Default,Plotly R Library ggplot2		1644	
JAVA	XChart 3.5.1 javafx JFreeChart	setMarkerSize (16) LegendPosition	1644	

Box Plots				
Language	Library	Parameters	Number of plots	Types(Box with)
Python	Matplotlib v2.1.2		1000	

Bibliography

[1] James Eagan. Project datasets. https://perso.telecom-paristech.fr/eagan/class/igr204/datasets. [Online; accessed 20-April-2018].

[2] Michael Vogt. Datplot. https://vincentarelbundock.github.io/Rdatasets/datasets.html, 2011. [Online; accessed 20-April-2018].

[3] plotly/datasets. Latex — Wikipedia, the free encyclopedia. https://github.com/plotly/datasets, 2011. [Online; accessed 20-April-2018].

[4] vincentarel bundock. Rdatasets. https://vincentarelbundock.github.io/Rdatasets/datasets.html, 2011. [Online; accessed 20-April-2018].

[5] Paulo Roberto Silva Chagas Junior, Alexandre Abreu De Freitas, Rafael Daisuke Akiyama, Brunelli Pinto Miranda, Tiago Davi Oliveira De Araújo, Carlos Gustavo Resque Dos Santos, Bianchi Serique Meiguins, and Jefferson Magalhães De Morais. Architecture proposal for data extraction of chart images using convolutional neural network. In *Information Visualisation (IV), 2017 21st International Conference*, pages 318–323. IEEE, 2017.

[6] Po-shen Lee, Jevin D West, and Bill Howe. Viziometrics: Analyzing visual information in the scientific literature. *IEEE Transactions on Big Data*, 4(1):117–129, 2018.