Automatic Annotation of Visualizations

Chufan Lai[1]* Zhixian Lin[1]† Can Liu[1]‡ Yun Han[1]§ Ruike Jiang[1]¶ Xiaoru Yuan[1,2]‖

1) Key Laboratory of Machine Perception (Ministry of Education), and School of EECS, Peking University
2) National Engineering Laboratory for Big Data Analysis Technology and Application, Beijing, China

Abstract

In this paper, we propose a technique for automatically annotating visualizations based on the user's textual descriptions. In our approach, the annotating task is fulfilled by performing a series of automatic visual searches. First, the description of the visualization is parsed into search requests for certain visual entities. At the same time, all visual entities exhibited in the visualization, along with their visual properties are extracted using Object Detection techniques based on Mask-RCNN models. Knowing what are there and what to look for, we then fulfill the generated search requests, so as to anchor each descriptive sentence to the described focal areas. In the next step, the corresponding annotations can be crafted efficiently. We have built a prototype tool that allows the user to upload a visualization image with its descriptions, and generates customized annotations.

Index Terms: Storytelling, annotations

1 Introduction

Visualization plays an important role when people share their data findings. However, reading a visualization could be a non-trivial task. When a description is given about the visualization, the audience needs to comprehend the description, knowing which entities or properties have been mentioned, and then visually search for them in the image. Due to the limited short-term memory, the reader often has to switch frequently between the description and the image.

A simple way to improve this process is to complement the visualization with proper annotations. By providing annotations, the presenter can effectively guide the audience's attention to the image area he/she is describing. It spares the audience the time for visual search and thus increases the efficiency of communication. Despite the benefits, a well-annotated visualization can be difficult to craft. Annotating all entities on the same image requires a good placement [1] to avoid occlusion. An easier way is to annotate one area at each time in a series of animation frames. But a fluent animation still demands time and expertise to make.

In this paper, we propose an automatic visualization-annotating technique to help presenters efficiently craft annotations for the purpose of data story-telling. With the proposed technique, the user can upload a visualization image with the corresponding description, and get a well-annotated visualization in a blink of an eye. Step-by-step animations are provided for a more fluent presentation. To that end, we process the visualization image and description into structured data respectively using state-of-the-art Object Detection (OD) and Natural Language Processing (NLP) techniques. By automatically matching each descriptive sentence with the described visual entities, we can generate the corresponding annotations in an efficient and accurate way.

2 Design Details

The proposed automatic annotating technique consists of three major modules: OD, NLP and Annotation. Each module is designed to process a specific part of the visualization.

2.1 Object Detection

The OD module processes the visualization image, and identifies the visual entities exhibited in the visualization. Visual entities can be categorized into two types: the data entities and the auxiliary entities. ***Data entities*** are visual elements that are used to encode the actual data, like the bars in a bar chart, or the points in a scatterplot. ***Auxiliary entities***, on the other hand, often convey the visual mapping or the context of the data. Examples include axes, titles, color legends, data labels, etc. Both types of entities can be referred to by their category names (e.g. "the X axis"), their visible labels (e.g. "the temperature"), or their visual properties (e.g. "the large red point").

For either data [3] or auxiliary entities [4], there do exist reverse-engineering detection techniques in the visualization community. However, most of them are specialized for certain visualization types. With a new visualization, they need to be redesigned. Seeking to build up a more concise and universal framework, we adopt Mask-RCNN [2], a state-of-the-art object detection technique, for the purpose of visual entity detection. Fig. 1 demonstrates this process.

With the detected contour of each visual entity, we also extract the visual properties (e.g. color, size) and the visible texts (e.g. axis titles, data labels) using Optical Character Recognition (OCR). Such texts are passed to the NLP module to help interpret the description.

2.2 Natural Language Processing

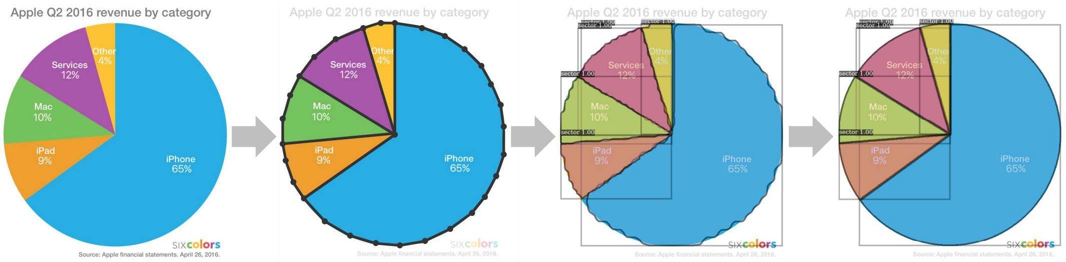
The NLP module handles the textual description, in order to understand what kind of visual entities/properties have been described, and generate the corresponding search requests. More specifically, we first segment the description into sentences. Part-of-speech (POS) tagging and dependency parsing are applied to each sentence respectively to analyze the types of words (e.g. noun, verb, etc.) and the sentence structure.

Vocabularies are defined to help identify keywords that describe visual properties (e.g. "red", "large") and visualization functionalities (e.g. "axis", "legend"). Such keywords are often well-known to ordinary people, but seldom visible in the image. Visible texts extracted by OCR are also recognized and bound to their categories (e.g. axis title). A sentence library is also defined for each kind of description (e.g. color/size/axis description) so that we can understand the dependencies between entities and their properties. For example, the phrase "the points in the middle" has the structure "[entity] [preposition] [location]", while both "point" and "middle" are identified as keywords. Structured data like $shape$: "$point$", $location$: "$middle$" are passed to the Annotation module for automatic visual searches.

2.3 Annotation

The Annotation module matches the detected entities with the search requests, so as to generate the animated annotations for the purpose

*e-mail: chufan.lai@pku.edu.cn
†e-mail: zhixian.lin@pku.edu.cn
‡e-mail: can.liu@pku.edu.cn
§e-mail: yunhan@pku.edu.cn
¶e-mail: jiangrk@pku.edu.cn
‖e-mail:xiaoru.yuan@pku.edu.cn


Apple Q2 2016 revenue by category
Other 4%
Services 12%
Mac 10%
iPad 9%
iPhone 65%
sixcolors
Source: Apple financial statements. April 26, 2016.
sector 1.00

(a) Original (b) Labeled (c) Predicted (d) Final

Figure 1: The workflow of visual entity detection. We collect various types of a) visualization images and b) label the contours of visual entities for model training. The object detection model is able to return c) both the bounding box and the rough contour for each entity. d) We refine the contours so that they can be used for accurate annotating.

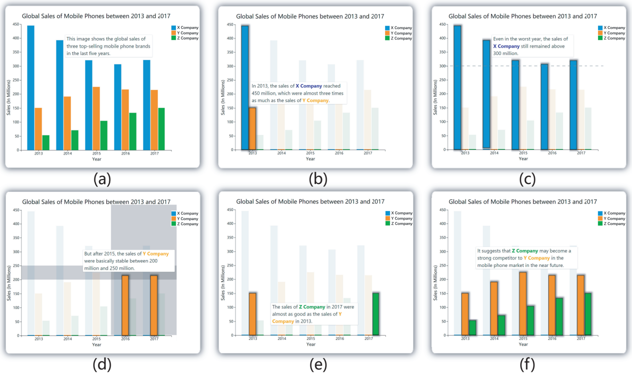
of data story-telling. Different types (color/size/location) of matching are performed individually, with their results combined to handle composite descriptions (e.g. "the large red point in the middle"). Once found, the described entities are highlighted in the image, with the descriptive sentence shown aside as the annotation. A force-directed mechanism is used to place the annotations so that they are close enough to the described entities and won't overlap with any existing entity. Different sentences are displayed in different frames of the animation to promote a step-by-step presentation.

3 Case Study

For the case study, we present a complicated data story on a grouped bar chart. It depicts the global sales of mobile phones in 2017 for three anonymous companies: X, Y, and Z (Fig. 2). The x-axis is titled "Year", while the y-axis is titled "Sales" with a unit "Million". Three color legends are present for the three companies.

The description contains 9 sentences: *(1) This image shows the global sales of three top-selling mobile phone brands in the last five years. (2) We see that X Company has always been the best-selling brand. (3) In 2013, the sales of X Company reached 450 million, which were almost three times as much as the sales of Y Company. (4) Even in the worst year, the sales of X Company still remained above 300 million. (5) As the second-ranked brand, Y Company's sales rose steadily from 2013 to 2015. (6) But after 2015, the sales of Y Company were basically stable between 200 million and 250 million. (7) Z Company, on the other hand, shows a rapidly growing trend in the last five years. (8) The sales of Z Company in 2017 were almost as good as the sales of Y Company in 2013. (9) It suggests that Z Company may become a strong competitor to Y Company in the mobile phone market in the near future.*

In sentence (1), the presenter explains the contents of the chart without any particular focus. Therefore, it's a context sentence and is displayed in the empty space (Fig. 2 (a)). In sentence (3), two legends (X Company and Y Company) are mentioned together in the same year "2013". Bars with the two legend colors are successfully identified and highlighted in this year (Fig. 2 (b)), with the description displayed aside. In sentence (4), an auxiliary line is appended automatically for the numerical threshold "300 million", which is described using preposition "above" (Fig. 2 (c)). Similarly, two shaded areas are used in sentence (5) for the two range related descriptions: "between 200 million and 250 million" and "after 2015" (Fig. 2 (d)). Sentence (8) is somehow challenging since the two companies are mentioned in different years. We see that the annotation box is placed in the empty space between the two entities (Fig. 2 (e)). It demonstrates the effectiveness of our annotation positioning algorithm. In the last sentence, two companies are mentioned without specifying any axis range. Therefore, all satisfying entities are highlighted (Fig. 2 (f)).



(a) (b) (c) (d) (e) (f)

Figure 2: Automatic annotations for a bar chart with titled axes and color legends.

4 Conclusion

In this paper, we propose an automatic annotating technique to promote efficient data storytelling with visualizations. The presenter can upload a visualization with the corresponding textual description, and get a series of vivid animations with well-annotated visualizations for the purpose of data storytelling. The process is highly efficient and can be finished in seconds. It saves the presenters lots of time and efforts for annotation crafting, allowing them to focus more on the contents of the data stories.

References

[1] R. Azuma and C. Furmanski. Evaluating label placement for augmented reality view management. In *Proceedings of the 2nd IEEE/ACM international Symposium on Mixed and Augmented Reality*, page 66. IEEE Computer Society, 2003.

[2] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2980–2988, 2017.

[3] D. Jung, W. Kim, H. Song, J. Hwang, B. Lee, B. H. Kim, and J. Seo. Chartsense: Interactive data extraction from chart images. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Denver, CO, USA, May 06-11, 2017.*, pages 6706–6717, 2017.

[4] J. Poco, A. Mayhua, and J. Heer. Extracting and retargeting color mappings from bitmap images of visualizations. *IEEE Trans. Vis. Comput. Graph.*, 24(1):637–646, 2018.