VisMCA: A Visual Analytics System for Misclassification Correction and Analysis

VAST 2020 Mini-Challenge 2 Award: Honorable Mention for Detailed Analysis of Patterns of Misclassification

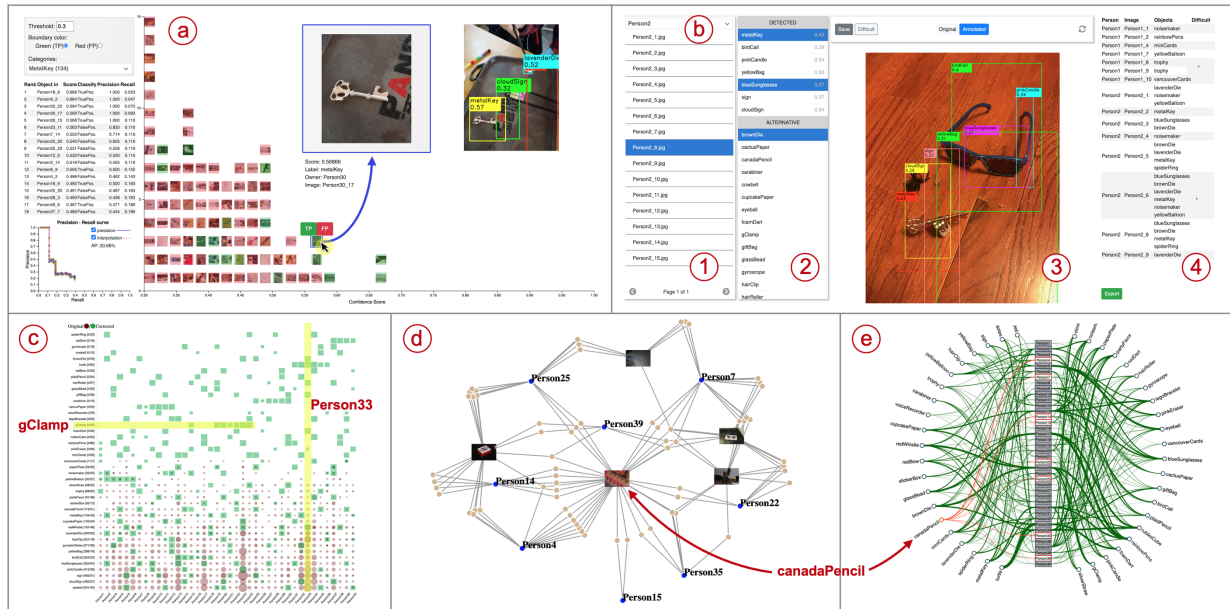Huyen N. Nguyen, Jake Gonzalez, Jian Guo, Ngan V.T. Nguyen, and Tommy Dang

Fig. 1. The VisMCA interface. The system presents the distribution of detections and allows real-time correction by the user (a). The image correction pipeline (b) supports label annotation and resulting data export. The matrix (c) visualizes the original and corrected labels with regards to respective owners. The filtered network of people sharing the same objects revolves around *canadaPencil* (d). The list-with-links repurposed network (e) provides an interactive visualization to identify the two-way owner-object relationships.

Abstract—This paper presents VisMCA, an interactive visual analytics system that supports deepening understanding in ML results, augmenting users' capabilities in correcting misclassification, and providing an analysis of underlying patterns, in response to the VAST Challenge 2020 Mini-Challenge 2. VisMCA facilitates tracking provenance and provides a comprehensive view of object detection results, easing re-labeling, and producing reliable, corrected data for future training. Our solution implements multiple analytical views on visual analysis to offer a deep insight for underlying pattern discovery.

Index Terms—VAST challenge, Visual analytics, Misclassification correction and analysis, Network visualization, Matrix visualization.

◆

---

1 INTRODUCTION

The VAST Challenge 2020 Mini-Challenge 2 asked participants to uncover the relationships between people, objects, and image data retrieved from a social media platform, ultimately leading to identify the "totem"–a secret item shared within an unknown subgroup. There are 900 images, belonging to 40 people and describing 43 classes, running through a machine learning (ML) model to identify objects. Visualization techniques have previously been applied towards the explainability

of the ML model [3,4]. In this paper, we utilize visualization and visual analytics to correct and analyze misclassification.

We developed VisMCA, a comprehensive web-based visual analytics system that facilitates human-in-the-loop in misclassification correction and analysis, implemented with Python, HTML5, and D3js [1]. The systematic interface of VisMCA is shown in Figure 1. The general workflow includes 1) characterize the detection distribution across different levels of confidence score, then 2) correct misclassification and produce data for future ML uses, and 3) analyze underlying patterns via multiple analytical linked views. The linked views approach is applied extensively to construct a dashboard assisting monitoring tasks [2] and situational analysis [5]. The complete report and demonstration are available online at `https://huyen-nguyen.github.io/VAST2020mc2/`.

2 SYSTEM ARCHITECTURE

2.1 Detection Distribution

The overall view is presented in Figure 1-a. The detection distribution provides a quick overview of images with regards to different confident score threshold. Regarding interaction, users can quickly cut to images

- *The authors are with the Department of Computer Science, Texas Tech University, Lubbock, TX, USA. E-mail: {huyen.nguyen | jake.gonzalez | jian.guo | ngan.v.t.nguyen | tommy.dang}@ttu.edu.*
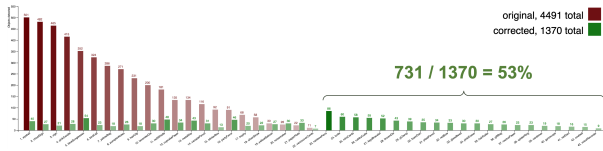- *Huyen N. Nguyen and Jake Gonzalez contributed equally to this work.*

Fig. 2. Distribution of classes before and after correction process.

with low initial confidence, mark true positives and false positives and update results on-the-fly by a simple right-click, and bulk select within a responsive layout. We use Average Precision - precision averaged across all recall values to evaluate each classifier's performance.

2.2 Misclassification Correction

Figure 1-b describes the pipeline for correcting classification errors. Panel 1 consists of image and person selection dropdown lists. Panel 2 contains a list for selecting and deselecting labels, divided into two sections: Detected–labels from the image's prediction, sorted by confidence score, and Alternative–all the other labels, sorted alphabetically. The sorted order aid users to quickly browse and select desire labels. Panel 3 displays the image frame, which includes two modes for displaying images: Original and Annotated with labels, bounding boxes, and confidence score, providing users with explicit illustration of the detections. Users can click the "Save" button to save the assigned labels, whose effect is updated on-the-fly on the data table in Panel 4.

With the "Export" feature, the re-labeled results can be exported in CSV format for further use. If the user is unsure about their manual annotation on some example, they can mark as *"Difficult."* This feature provides value to ML algorithms and future users of the results, as *difficult* examples can be treated differently in different training settings.

2.3 Object Distribution Matrix

The object distribution matrix shows how objects are distributed amongst the 40 suspected group members, as shown in Figure 1-c. The original data is depicted by brown circles whose size depends on the number of detections, and opacity is determined by the average of the confidence scores across all images containing that object. The corrected data is depicted by the green squares whose size represents an object's occurrences in a person's images. The people and objects are presented horizontally and vertically, along with statistics of the occurrences in images in the original and corrected dataset.

2.4 Network of Top Object

To gain a clear view of the interconnection between different entities, we built a complete network of people and their objects, which can be found in the demonstration link in Section 1. For simplicity, we present the filtered network revolving around *canadaPencil*, as shown in Figure 1-d. A similar filtering technique is found to be useful when the network architecture is complex [6]. The blue and yellow nodes represent people and objects, respectively, where nodes representing the same object are linked to a reference image.

2.5 List-with-links Repurposed Network

The visualization is shown in Figure 1-e, in which people are shown in a vertical column, and objects are arrayed in a circle. The curved green links use width to indicate the number of detections: the number of images containing an object belongs to a person, highlighted in orange on hovering. This visualization is useful to quickly find groups of people that shared objects and the set of objects that one owns.

3 RESULTS

There are a total of 43 classes in the training set, distribution quite evenly among the 900 images. However, there are detections in only 22 classes and no detection for any of the remaining classes. This can be seen from the matrix in Figure 1C, where the upper half of the matrix shows no detection (no brown circles), while there actually are objects in those images (green squares). The ML model missed out at least
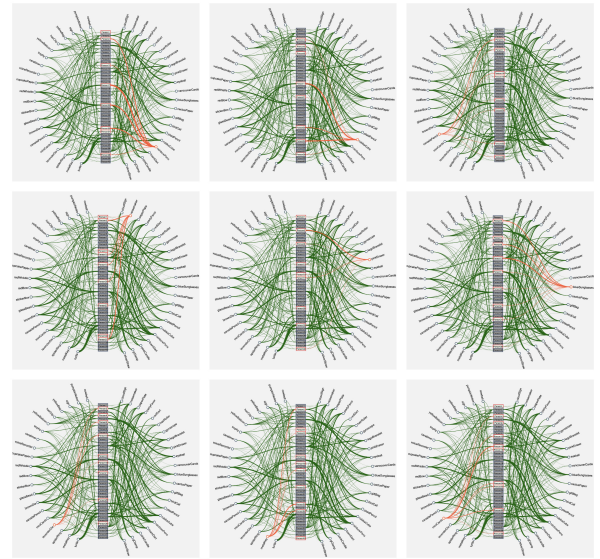

Fig. 3. The networks of nine objects, each shared by *at least* 8 people.

53% of the objects, as shown in Figure 2, hence not a good model. As highlighted in Figure 1C, from *Person19* to *Person24* having *gClamp*. For *Person33*, only one object was correctly detected (one overlap); all other detections in this person's images are false positive.

As stated in the challenge description, there are eight people connected to the totem. As shown in Figure 3, there are four objects owned by only eight people: *canadaPencil*, *rainbowPens*, *noisemaker*, and *rubiksCube*. We speculated that each person should have at least two images of the totem, representing sending and receiving signals. Among those four objects, only *canadaPencil* owners have two or more images per person, while the rest includes owners having one image, which concludes that *canadaPencil* is the totem.

4 CONCLUSION AND FUTURE WORK

This paper presents VisMCA, an interactive visual analytics system for a comprehensive view of ML results: augmenting users' capabilities in correcting misclassification and providing an analysis of underlying patterns. To make the correction process more efficient, we suggested considering overlapping bounding boxes and common object combinations. This approach might require validation testing and verification in future work, along with addressing uncertainty in classification results.

REFERENCES

[1] M. Bostock, V. Ogievetsky, and J. Heer. D³ data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011.

[2] T. Dang, V. Pham, H. N. Nguyen, and N. V. Nguyen. Agasedviz: visualizing groundwater availability of ogallala aquifer, usa. *Environmental Earth Sciences*, 79(5):1–12, 2020.

[3] T. Dang, H. Van, H. Nguyen, V. Pham, and R. Hewett. Deepvix: Explaining long short-term memory network with high dimensional time series data. In *Proceedings of the 11th International Conference on Advances in Information Technology*, IAIT2020. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3406601.3406643

[4] D. D. Le, V. Pham, H. N. Nguyen, and T. Dang. Visualization and explainable machine learning for efficient manufacturing and system operations. *Smart and Sustainable Manufacturing Systems*, 3(2):127–147, 2019.

[5] H. N. Nguyen and T. Dang. EQSA: Earthquake situational analytics from social media. In *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 142–143, 2019.

[6] H. Van, H. N. Nguyen, R. Hewett, and T. Dang. Hackernets: Visualizing media conversations on internet of things, big data, and cybersecurity. In *2019 IEEE International Conference on Big Data (Big Data)*, pp. 3293–3302. IEEE, 2019.