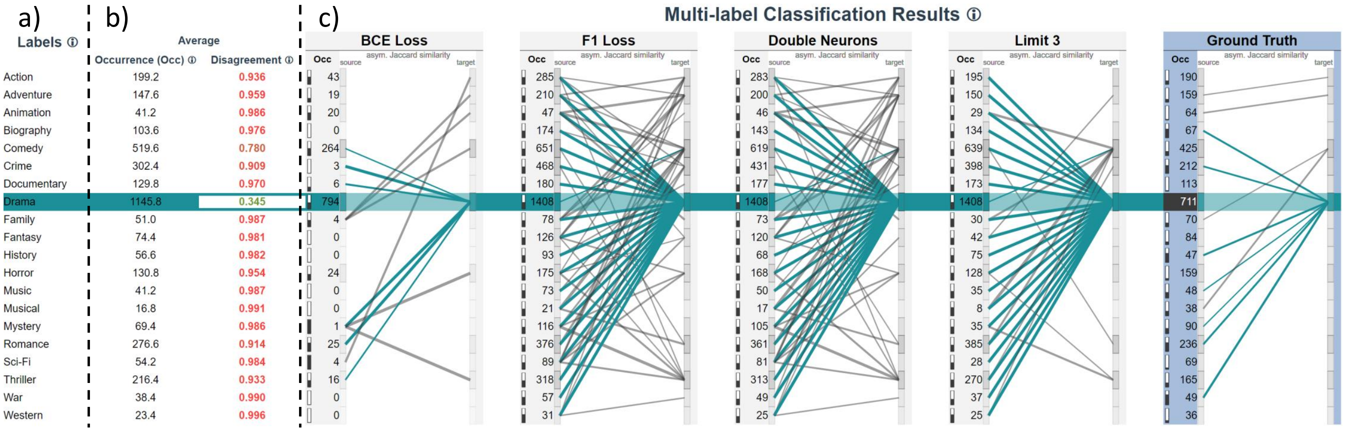
Vision, Modeling, and Visualization (2021)
B. Andres, M. Campen, and M. Sedlmair (Eds.)

Visual Comparison of Multi-label Classification Results

Cedric Krause [†1], Shivam Agarwal [‡1], Mohammad Ghoniem [§2], and Fabian Beck [¶1]

[1]University of Duisburg-Essen, Germany
[2]Luxembourg Institute of Science and Technology, Luxembourg



Labels	Average Occurrence (Occ)	Average Disagreement	BCE Loss Occ	F1 Loss Occ	Double Neurons Occ	Limit 3 Occ	Ground Truth Occ
Action	199.2	0.936	43	285	283	195	190
Adventure	147.6	0.959	19	210	200	150	159
Animation	41.2	0.986	20	47	46	29	64
Biography	103.6	0.976	0	174	143	134	67
Comedy	519.6	0.780	264	651	619	639	425
Crime	302.4	0.909	3	468	431	398	212
Documentary	129.8	0.970	6	180	177	173	113
Drama	1145.8	0.345	794	1408	1408	1408	711
Family	51.0	0.987	4	78	73	30	70
Fantasy	74.4	0.981	0	126	120	42	84
History	56.6	0.982	0	93	68	75	47
Horror	130.8	0.954	24	175	168	128	159
Music	41.2	0.987	0	73	50	35	48
Musical	16.8	0.991	0	21	17	8	38
Mystery	69.4	0.986	1	116	105	35	90
Romance	276.6	0.914	25	376	361	385	236
Sci-Fi	54.2	0.984	4	89	81	28	69
Thriller	216.4	0.933	16	318	313	270	165
War	38.4	0.990	0	57	49	37	49
Western	23.4	0.996	0	31	25	25	36

Figure 1: *The main view of our visualization:* a) *the labels in the dataset are represented as rows with their names displayed,* b) *aggregated statistics about each label,* c) *each column represents one multi-label classification result (in this example we have five multi-label classification results); the numbers on the left of each column (Occ) show the amount of items that were assigned the label of that row within the multi-label classification result; the graphs on the right of each column show overlap relationships between labels.*

Abstract

In multi-label classification, we do not only want to analyze individual data items but also the relationships between the assigned labels. Employing different sources and algorithms, the label assignments differ. We need to understand these differences to identify shared and conflicting assignments. We propose a visualization technique that addresses these challenges. In graphs, we present the labels for any classification result as nodes and the pairwise overlaps of labels as links between them. These graphs are juxtaposed for the different results and can be diffed *graphically. Clustering techniques are used to further analyze similarities between labels or classification results, respectively. We demonstrate our prototype in two application examples from the machine learning domain.*

1. Introduction

To structure large datasets, the data items can be grouped depending on their properties into multiple categories. In many scenarios, a given item can be part of several categories at the same time. Relevant examples are assigning multiple genres to movies, labeling various objects contained in images, assigning skills to employees, or keywords to documents. Hence, each item can have multiple labels, representing the categories it belongs to, and each label can be given to multiple items. Such multi-label data can originate from algorithms or human classification. It is likely that the multi-label classification results (henceforth: multi-labeling results) of the same items will vary for different sources as the classification procedure might not work flawlessly or the labeling task might be ambiguous. Visually comparing these different results can reveal shared and conflicting assignments, and it can provide relevant in-

[†] e-mail: cedric.krause@paluno.uni-due.de
[‡] e-mail: shivam.agarwal@paluno.uni-due.de
[§] e-mail: mohammad.ghoniem@list.lu
[¶] e-mail: fabian.beck@paluno.uni-due.de

sights about the multi-labeling results beyond traditional quantitative metrics like precision and recall.

The visual analysis of individual multi-labeling results can be translated to the problem of visualizing overlapping sets, which has been well researched [AMA*16]. However, comparing multi-labeling results from multiple sources adds new complexity to the visualization problem and is not yet sufficiently discussed in set visualization literature.

We propose a novel approach for the comparison of multi-labeling results that is based on a tabular representation enriched with inline graphics as the main view (Figure 1). Labels are represented as rows and multi-labeling results as columns. One label and one multi-labeling result can be selected and get highlighted. Each selection specifies a certain set of labeled items. The overlap of this selected set with other sets of items identified by labels and multi-labeling results, respectively, are then shown as small bars within the table cells. Labels are connected by graph edges within each column (multi-labeling result) to show the overlap of items between the labels. The approach supports visually comparing the graph of a selected multi-labeling result with all other ones to identify missing and excess links (Figure 2b). Lastly, we allow for clustering of both, the labels and the results, to enhance the display order and reduce clutter. Details of selected items are provided in a list-based view (Figure 2c/d).

Our contributions include the characterization of the research problem along abstract analysis questions (Section 3). Building on these, we design the approach for the visual comparison of multi-labeling results (Section 4). We illustrate the capabilities of our approach along two specific use cases from the machine learning domain, covering an exemplary engineer's perspective on improving his own models as well as comparing top contenders of an image recognition challenge (Section 5). The tool and a video are available as supplemental material along with the submission[†].

2. Related Work

For our approach, previous work on visualizing multi-labeling results is most relevant. We also discuss works on the related set visualizations and graph comparison techniques, which formed a basis of our visual analysis approach considering multi-label assignments as sets and abstracting label co-occurrence to graphs.

2.1. Visualizing Multi-label Classification Results

Usually, quantitative metrics are used for statistical comparison of multi-label classifiers (e.g., [MKGD12]). Recent works have shown the usefulness of including these metrics in visual analytics tools. For instance, *ComDia+* [PLHL19] helps to diagnose the reasons behind misclassifications, and improve the overall performance of the classifiers. The approach shows model performance in individual rows, while columns indicate the actual class of the data item. In each cell, precision and recall measures for each model are encoded via two semi-circles. Similarly, Theissler et al. [TVB*20] proposed a model-agnostic approach to compare multi-class classifiers and used different charts in the interface to compare classifier performance. However, these techniques rely on visually comparing the performance metrics of classification results. As a result, they are unable to support visual exploration and comparison of qualitative aspects like relationships among assigned labels. Demonstrating the value of qualitative aspects, Alsallakh et al. [AJY*18] extended a confusion matrix visualization to show that the confusion pattern over several classes followed a hierarchical structure using a single-label (only one label prediction per data item) multi-class classifier for image data. Hence, in our approach, we focus on representing relationships among labels from one multi-labeling result, and also compare the relationships across different results.

Several visualizations have been proposed to analyze single-label multi-class classification results (e.g., [AHH*14; RAL*17]). However, these techniques do not generalize well to visualize the multi-label data, where each data item can be assigned more than one label. Only a few works have been proposed that support visual analysis of multi-labeling results. For instance, *UnTangle Map* [CLG16] uses triangle vertices to show labels and visualizes the data items in a web of such connected triangles. This technique was demonstrated to be useful in understanding the relationship between labels. However, unlike our approach, the visualization can only show one classification result at a time and is unable to support comparison between different classification results.

2.2. Set Visualizations

We can model each label as a set and data items that were assigned the label as elements of the set. In doing so, set visualizations can be used to represent a multi-labeling result. Euler- and Venn-based set visualizations (e.g., [Wil12]) are intuitive and can represent all overlap relations when drawn for a small number of sets (<6). However, in multi-labeling scenarios, typically more labels are available. Hence, we do not include Euler- and Venn-based set visualizations in our approach. Most of the recent research focus on scalable set visualizations uses aggregated representation of sets and elements (e.g., *UpSet* [LGS*14] and *AggreSet* [YEB16]). Recently, hypergraph visualizations of set data have also demonstrated the ability to scale well (e.g., [VBP*19]). These techniques represent the data based on only one criterion of deciding the set membership of elements. In contrast, in our approach, we aim to compare the multi-labeling results on the same data items from different sources. Hence, we cannot directly apply these techniques to represent and compare several multi-labeling results.

2.3. Visual Comparison of Graphs

Generally, multilayer graph visualization can show different edge sets of a graph modeled in the different layers [MGM*19]. Our focus is on graphs that have the same nodes in all layers (namely, the available labels). Juxtaposition is one way to support comparison tasks [GAW*11; LJS20]. It has been shown that juxtaposed designs for the comparison of individual charts are familiar to people and easy to understand [LJS20]. Juxtaposed comparative designs are often combined with careful selection of a graph layout. For instance, *TimeArcTrees* [GBD09] uses a linear vertical layout, computes node positions to reduce the edge crossings, and jux-

[†] Hosted at: `https://vis-tools.paluno.uni-due.de/mlc/`

taposes graph variants on a horizontal axis. Parallel Edge Splatting [BVB*11] uses a similar linear vertical layout to represent nodes, but places them on two parallel vertical lines per graph instance. Directed edges are then drawn between nodes from two adjacent parallel lines. As a result, visualizing a graph using parallel edge splatting becomes more scalable because the crossings of the straight-line edges are easier to resolve than crossing arcs in *TimeArcTrees*. While this approach was developed to show dynamic graphs, Beck et al. [BPD11] suggest a similar layout for graph comparison. In addition, specific interactive highlighting features and graph merge operations support the comparison more explicitly. A user study has shown that various relevant visual patterns such as clusters, fan-in/-out structures, and high-level relationships can be identified with the approach [ABZD13]. We take inspiration from the technique and adapt it to compare pairwise overlap among the labels for different multi-labeling results.

3. Analysis Questions

Based on an analysis of the problem domain, we formulate analysis questions (AQ) that have guided the design decisions of our visualization approach. We group them by the targeted scope of the data and arrange them along their growing complexity. The analysis questions act as a reference throughout the paper and allow for a better connection between visualization design, application examples, and discussion. The analysis questions were discussed among the authors and iteratively refined, striving for completeness, consistency, and distinctness. The discussion was based on our previous experience working with multi-label classification problems (mainly, regarding document and image collections) and shortcomings of existing approaches (see Section 2).

A first step towards comparing labels over two or more multi-labeling results is contrasting label occurrences, which refers to the numbers of items that were assigned a given label. Generally, similar occurrence values across multiple results suggest similarity in label assignment. However, similar values do not necessarily imply that the label is undisputed, as the label could have been assigned to different groups of items with similar sizes. Therefore, the occurrence value alone is not sufficient. It is also required to analyze over several multi-labeling results whether the items with a particular label are the same. The identification of results in which certain items are classified differently than in the majority of other results can lead to a deeper understanding of the cause and reveal misconceptions in the multi-labeling results beyond general inaccuracy.

AQ 1 – Items and their labels over different results

AQ 1.1 *Which labels are most frequent in the dataset across different results?*

AQ 1.2 *To what degree are items assigned with the same label over different results?*

AQ 1.3 *How do the assigned labels for a specific item differ for different results?*

Additionally, similarity relationships between the labels are relevant to understand a multi-labeling result. It is possible, yet not scalable, to manually find co-occurrences of labels by scanning individual label sets of items. A more advanced analysis requires aggregating this information at the level of labels and investigating the relationships of labels depending on the overlap of the items they were assigned to. Special types of similarities, like subset relationships (e.g., all items with one label are assigned another label as well), might form hierarchy-like structures. Lastly, the pairwise similarity of multiple labels can reveal clusters of labels and, in turn, implicit higher-level topics among the labels.

AQ 2 – Relationships of labels within one result

AQ 2.1 *How strongly and in what way do specific labels overlap in a given result?*

AQ 2.2 *What function does a label serve in relation to other labels within a result?*

AQ 2.3 *Do the labels form higher-level clusters of similar labels?*

Comparing sources of multi-labeling results, we need to contrast their above-mentioned similarities. We want to study whether the same overlap relationships and clusters of labels can be observed throughout all results.

AQ 3 – Relationships of labels over different results

AQ 3.1 *Do the same labels show similar relationships for all results?*

AQ 3.2 *Are the clusters of labels the same for all results?*

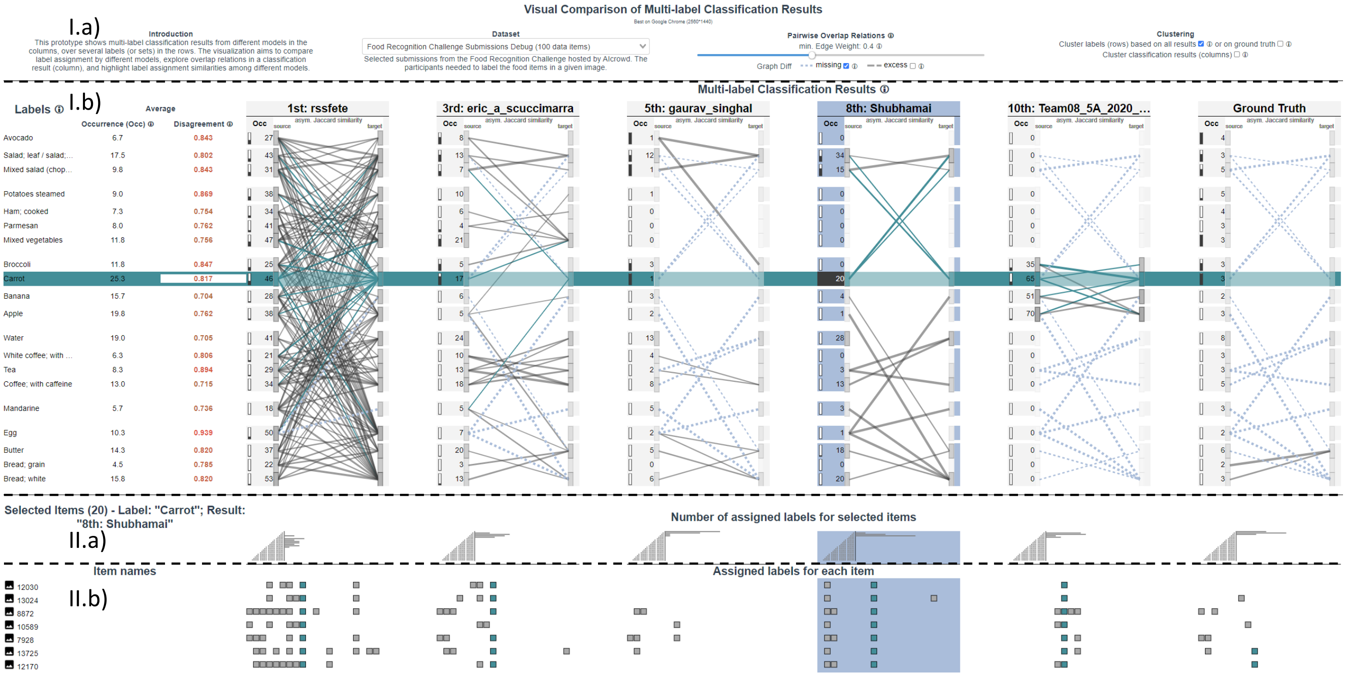
Out of the presented analysis questions, ▌**AQ 1** and ▌**AQ 3** concern the comparison of different multi-labeling results. The questions in ▌**AQ 2** complement those of ▌**AQ 1**, but are also needed in order to answer the questions posed in ▌**AQ 3**. The analysis questions ▌**AQ 2** and ▌**AQ 3** also clearly separate the multi-label classification tasks from single-label multi-class classification, which prevents said relationships between labels by definition.

4. Visual Comparison Approach

We propose a visualization approach to address the challenge of comparing various multi-labeling results. Figure 2 shows the system as a whole with its two main sections. The key component is the enriched tabular representation in the center, which is called *result comparison view*. In the bottom part of the screen, the *detail view* allows for further investigation of individual items and label patterns among them.

4.1. Result Comparison View

The main visualization (Figures 1 and 2b) has a tabular layout and maps the labels in the rows onto the label assignments and relationships in the respective multi-labeling results in the columns. Selection of labels and results steers the comparison of data and modifies which information is highlighted. Certain features can be turned on or off via controls atop (Figure 2I.a).



Visual Comparison of Multi-label Classification Results

Best on Google Chrome (2560*1440)

Introduction

This prototype shows multi-label classification results from different models in the columns, over several labels (or sets) in the rows. The visualization aims to compare label assignment by different models, explore overlap relations in a classification result (column), and highlight label assignment similarities among different models.

Dataset

Food Recognition Challenge Submissions Debug (100 data items)

Selected submissions from the Food Recognition Challenge hosted by AIcrowd. The participants needed to label the food items in a given image.

Pairwise Overlap Relations

min. Edge Weight: 0.4

Graph Diff: missing ☑ excess □

Clustering

Cluster labels (rows) based on all results ☑ or on ground truth □

Cluster classification results (columns) □

Multi-label Classification Results

Labels	Average Occurrence (Occ)	Average Disagreement	1st: rssfete Occ	3rd: eric_a_scuccimarra Occ	5th: gaurav_singhal Occ	8th: Shubhamai Occ	10th: Team08_5A_2020_… Occ	Ground Truth Occ
Avocado	6.7	0.843	27	8	1	0	0	4
Salad; leaf / salad;…	17.5	0.802	43	13	12	34	0	3
Mixed salad (chop…	9.8	0.843	31	7	1	15	0	5
Potatoes steamed	9.0	0.869	38	10	1	0	0	5
Ham; cooked	7.3	0.754	34	6	0	0	0	4
Parmesan	8.0	0.762	41	4	0	0	0	3
Mixed vegetables	11.8	0.756	47	21	0	0	0	3
Broccoli	11.8	0.847	25	5	3	0	35	3
Carrot	25.3	0.817	46	17	1	20	65	3
Banana	15.7	0.704	28	6	3	4	51	2
Apple	19.8	0.762	38	5	2	1	70	3
Water	19.0	0.705	41	24	13	28	0	8
White coffee; with …	6.3	0.806	21	10	4	0	0	3
Tea	8.3	0.894	29	13	2	3	0	3
Coffee; with caffeine	13.0	0.715	34	18	8	13	0	5
Mandarine	5.7	0.736	18	5	5	3	0	3
Egg	10.3	0.939	50	7	2	1	0	2
Butter	14.3	0.820	37	20	5	18	0	6
Bread; grain	4.5	0.785	22	3	0	0	0	2
Bread; white	15.8	0.820	53	13	6	20	0	3

Selected Items (20) - Label: "Carrot"; Result: "8th: Shubhamai"

Item names: 12030, 13024, 8872, 10589, 7928, 13725, 12170

I.a)
I.b)
II.a)
II.b)
asym. Jaccard similarity
source
target
Number of assigned labels for selected items
Assigned labels for each item

Figure 2: *The prototype contains four components:* I.a) *controls to select the dataset and settings that influence the main visualizations,* I.b) *the result comparison view in which the labels (rows) and the multi-labeling results (columns) are displayed,* II.a) *the selection detail view that shows the distribution of how many labels were assigned to how many items within the current selection (in this case 20 items),* II.b) *the item detail view which contains a pixel map of assigned labels to the individual items within the current selection.*

4.1.1. Aggregated Label Statistics

For each label, the corresponding name and aggregated statistics are shown on the left side of the result comparison view (Figure 1a/b). These statistics can highlight labels to the user that might be of special interest (e.g., very frequently or infrequently assigned labels, labels with high disagreement regarding the different multi-labeling results).

The first value is the average occurrence over all multi-labeling results. It can indicate the quantitative importance of a label within the dataset (**AQ 1.1**). Errors or misunderstandings are likely to have a bigger impact on the desired outcome when labels with high occurrence values are affected by them.

The other value is the disagreement between all multi-labeling results for assigning a particular label. It is the degree to which the multi-labeling results deviated on the items assigned with the given label (**AQ 1.2**). More precisely, it is the average pairwise disagreement of multi-labeling results for the label, which is calculated as the share of items that were assigned the label only by one of the results. For the *i-th* multi-labeling result, we define $R_i : L \rightarrow \mathcal{P}(C)$ to map a label $l \in L$ to the set $S \in \mathcal{P}(C)$ of items that were assigned the label l, where C is the collection of all items in the dataset. The disagreement regarding the label l can, therefore, be described as

$$d(l) = \begin{cases} 0, \text{ if } R_i(l) = R_j(l) = \emptyset \\ \frac{1}{n}\sum\limits_{i=1}^{n} \frac{1}{n-1} \sum\limits_{\substack{j=1 \\ j \neq i}}^{n} \frac{|R_i(l) \cup R_j(l)| - |R_i(l) \cap R_j(l)|}{|R_i(l) \cup R_j(l)|}, & \text{else} \end{cases}$$

where n is the number of multi-labeling results. If a label was assigned to the same items for all multi-labeling results, the disagreement will be 0. Vice versa, the disagreement will be 1 if each item is assigned the label in only one single result. To visually highlight labels with high disagreement, the value is also redundantly encoded using colors from green (low disagreement, 0.345) to red (high disagreement, 0.987) (Figure 1b).

4.1.2. Label Occurrences and Relationships per Result

Each multi-labeling result is represented on one column, which is subdivided into two parts (Figure 1c). On the left side are the occurrence values for each label, placed in the respective row of the label. On the right side is a node–link diagram that connects the rows of two respective labels based on their pairwise overlap relations.

The occurrence values show how often the labels are assigned within the column's multi-labeling result. This number represents $|R_i(l)|$, the amount of items in the *i-th* multi-labeling result that was assigned the respective label l (**AQ 1.1**). These values, especially

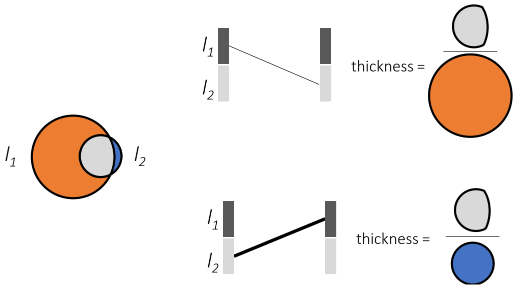


Figure 3: *The link thickness in the node–link diagram encodes the size of the overlap of items between the labels l_1 and l_2, normalized by the size of the item set of the source label of the link (l_1).*

when sorted (Section 4.1.4), can already be an indicator towards suggesting labels to investigate. Furthermore, the value has an influence on the node–link diagram in the respective column.

The graph shown in the node-link diagram is directed, and the direction of a link is always from left to right. To allow this concept, the graph is made artificially bipartite, meaning we discern between each label as a source (first set – on the left) and as a target (second set – on the right). The two sets of vertices encompass all labels of the dataset, but edges can only exist between vertices of different groups (i.e., from source to target). Beck et al. [BPD11] suggested this technique for graph comparison in order to improve the traceability of edges—and the overall visual scalability eventually—compared to representations that use arcs as links instead [GBD09].

Two boxes in every row serve as source and target nodes (▯). The opacity of the boxes encodes the occurrence value for the column's multi-labeling result relative to the maximum occurrence value over all labels and all results ($|R_i(l)|/\max_{i',l'}|R_{i'}(l')|$). This way, frequent labels in one result will appear more prominent. The thickness of each link in the diagram encodes the overlap between two labels, l_1 and l_2, which is calculated as the asymmetrical Jaccard similarity

$$J(l_1,l_2) = \frac{|R(l_1) \cap R(l_2)|}{|R(l_1)|}$$

between the connected labels (🟧 **AQ 2.1**). We chose an asymmetrical Jaccard similarity to express subset relationships between the labels. As illustrated in Figure 3, we can capture similarities where the set sizes are imbalanced. Judged for the larger set $R(l_1)$, the similarity might be weak, but for the smaller set $R(l_2)$, being almost a subset of $R(l_1)$, it is a much stronger connection—the symmetrical Jaccard similarity would lose this information. The asymmetrical similarities might also reveal subset hierarchies among the labels (e.g., *racing bicycle* is a subset of *bicycle* is a subset of *vehicle*), including non-perfect subset hierarchies (e.g., *bicycle* is mostly a *two-wheel vehicle*). To reduce clutter, we do not draw all pairwise links, but blend out weaker connections based on a threshold. The minimum similarity necessary can be interactively adjusted in the control panel above the table.

The visualization of multi-labeling results facilitates a comparison not only of the values in the table but also the relationships from the graphs, as the order of labels is the same across all columns. Due to this, the comparison of outgoing and incoming relationships with other labels is already partly supported across different multi-labeling results (🟥 **AQ 3.1** and 🟥 **AQ 3.2**), but is further eased by explicitly encoding graph *diffs* (Section 4.1.5).

4.1.3. Selection

We designed the interactive selection in the result comparison table to influence the contained visualizations and steer the comparison. It is interwoven with the colorization—as one of only two features in the tool. The selection color scheme is consistent throughout the entire system.

Before any selection is made, the occurrence values and graphs in the multi-labeling result columns apply a gray color scale. The user can select one label and one multi-labeling result at a time. The row of the selected label is then colored in turquoise (🟦). Figure 2 shows how the selection expands over the table cells onto all associated items like the nodes, incoming and outgoing links in the graphs for each multi-labeling result, as well as the labels of individual items (which are later discussed in Section 4.2.2). This groups all components of the visualization which represent the label, but also enables the user to better compare properties of the label in different multi-labeling results (🟥 **AQ 3.1**).

Selected results, on the other hand, are colored blue (▢) and that color is also propagated to the associated graph *diff* feature (more in Section 4.1.5). If both, a label and a multi-labeling result, are selected, there is one cell in the table that lays at their intersection. This can be considered the selected cell, and it is colored black (■) to clearly contrast it.

Whenever a selection is made, small, partially filled bars, which we call *overlap bars*, appear next to the occurrence values in the result comparison view (▯, ▯, and ▮). The size of a bar encodes how many of a cell's items are shared with those of a selected cell (🟨 **AQ 1.2**). There are three cases to discern, which reflect also in the color of the bar (Figure 4): (i) Whenever a label *and* multi-labeling result are selected, the bar is relative to the selected cell at their intersection (■). (ii) If only a label is selected, the bars in a given column are relative to the cell in the same multi-labeling result (🟦) but in the row of the selected label. (iii) Vice versa, if only a multi-labeling result is selected, the bars in a given row are relative to the cell in the same row (▢) but in the column of the selected multi-labeling result. The overlap bars can indicate special relationships between labels, like superset relationships. For example, the items of the selected cell are a superset of the items in those cells where the overlap bars are completely filled (▮). Hovering an overlap bar triggers a tooltip with a detailed description and the exact size of the overlap.

4.1.4. Ordering and Clustering

By default, the order of labels and multi-labeling results is taken from the raw data. Additionally, we implemented functionality to sort the labels in ascending or descending order of the label statistics (Section 4.1.1) or the occurrence values of an arbitrary multi-labeling result (🟨 **AQ 1.1**). The sorting can be selected on the header of the column by which the labels should be sorted.

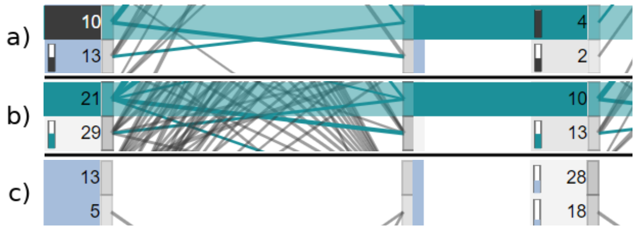


Figure 4: a) *A row and a column are selected, the black cell (10) is at their intersection and all black bars are relative to it. For example, all 4 items from the top right are among the selected 10 items.* b) *Only a row is selected, so, within each column, the bars are relative to the turquoise cell in the same column (🟦, 21 and 10).* c) *Only a column is selected, and, within each row, the bars are relative to the blue cell in the same row (🟦, 13 and 5).*

Alternatively, there are two modes of label clustering, one of which only considers the multi-labeling result marked as the ground truth, if available, while the other aggregates all multi-labeling results. The distances between the labels are calculated based on vectors that contain either 0 or 1 for each item, depending on whether the label was assigned to the particular item or not. In the aggregated clustering mode, the vector for each label is concatenated over all multi-labeling results. The metric to determine the distances between the vectors is the asymmetrical Jaccard similarity, again. We apply the *AGNES* algorithm [KR09] for clustering the labels using the above similarity definition, which creates a tree structure with the labels as leaves. The result contains an inherent order of leaves, which is then applied to rearrange the rows of the labels. The new ordering places connected labels closer together and, as a consequence, makes clusters of similar labels clearer to see.

In order to further enhance the visibility of clusters, we introduce gaps between the rows for visual separation (Figure 2I.b). There are two thresholds that determine the size of the gap between two labels. Links between clusters become easier to separate from those within clusters, as the former span over the gaps (▌**AQ 2.3**). If the links of a multi-labeling result frequently exceed the boundaries of clusters it can indicate deviance from the basis of the clustering (i.e. the ground truth or all results combined) (▌**AQ 3.2**).

Clustering of multi-labeling results is also possible and reorders the columns based on their similarity. It works analogously as described above for the label clustering on aggregated results, only that the label vectors are concatenated by column instead of by row. Also, differently sized gaps between the columns support the visual perception of clusters.

4.1.5. Graph Diff

When comparing different graphs, it can be difficult to remember and trace the equivalent relationships between the labels across the different columns. We addressed this issue by adding functionality to *diff* graphs. If the graph diff is enabled and a multi-labeling result is selected, the graph in the corresponding column is viewed as the focus and all other graphs will be adjusted relative to the focused graph (Figure 2I.b). While the focused graph remains unchanged, all non-focused graphs are then checked for differences in the graph links towards the focused one (considering the minimum weight for an edge to be drawn). Any link in the focused graph will be added to all other graphs if it does not exist already. To mark the added link, we use the blue color of the selected column and a dashed line with short dashes. The opposite case is also implemented. Any link that exists in a non-focused graph, but no link between the same labels exists in the focused graph, will also change its appearance. It keeps its original color but a dash pattern with longer dashes is applied. This largely keeps the original representation of the graph intact but adds a visual feature that helps the user understand differences in the graphs without looking back to the focused column (▌**AQ 3.1** and ▌**AQ 3.2**).

4.2. Detail View

Below the result comparison view is the detail view (Figure 2II.a/b). Whereas the result comparison view abstracts from the items and aggregates label occurrences, the purpose of the detail view is to allow the user to investigate specifics of the item selection and individual items.

4.2.1. Selection Detail View

When a multi-labeling result and a label are selected in the result comparison view, the items that were assigned that label within that multi-labeling result are selected. We have added a histogram in the column of each multi-labeling result that shows the amount of items that were assigned a certain number of labels (Figure 2II.a). This allows us to see if the label was typically assigned with many other ones, abstracting from the information of which specific labels it frequently occurs with. For instance, if the label was typically assigned alone, this can explain why there are only few relationships between it and other labels (▌**AQ 2.2**).

Left of the axis is the number of labels assigned, encoded into gray pixel marks. The pixel marks are grouped in blocks of five to make it easier to perceive their number. The bars on the right encode the number of selected items that were assigned the respective number of labels. Above the first pixel marks, if applicable, a bar represents the number of selected items that were not assigned any label within that multi-labeling result.

4.2.2. Item Detail View

Below, we list the individual items in the current selection. On the left, we provide the item name. On the right, aligned in the multi-labeling results' columns (Figure 2II.b), are pixel maps (□□ □□□ ■ □□ □ □ □) showing the assigned labels for the item in the respective result. The pixel map has as many slots as there are labels in the dataset. Each slot stands for one label and the order is the same as in the result comparison view (only now they are arranged from left to right). If the label was given to the item in that multi-labeling result, a square will be shown (□). In case the label that is selected in the result comparison view is assigned to the item, the square will instead be turquoise (■) (▌**AQ 1.3**).

Upon subselecting an item in the list, the squares for this item will be replaced by the actual labels given in the multi-labeling result, where the selected label in the result comparison view will

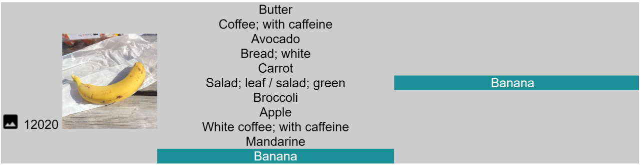

12020
Butter
Coffee; with caffeine
Avocado
Bread; white
Carrot
Salad; leaf / salad; green
Broccoli
Apple
White coffee; with caffeine
Mandarine
Banana
Banana

Figure 5: *One item expanded in the item detail view that shows the raw image and the assigned labels for two results.*

still be highlighted. Furthermore, the raw data item can be shown if such a view is available. In the case of image classification tasks, this view contains the original image (Figure 5).

5. Application Examples

To demonstrate our approach, we have applied it to two different image classification scenarios. One is from the perspective of an engineer who compares self-built machine learning models, while the other is from an image recognition challenge, which was held by AIcrowd [AIc].

Two senior co-authors having longtime experience in visual analytics worked with the first scenario, while two external machine learning experts from AIcrowd tested the approach for the second scenario. With each user, we performed two sessions. The first sessions were focused on the discussion of the analysis questions, as well as the explanation of the prototype and possible improvements. After these sessions, the users' feedback was incorporated into the implementation. About two weeks later, the second sessions followed, which were more focused on drawing insights through the use of the prototype. The users had time to freely explore the dataset and were later on asked to answer some of the analysis questions posed in Section 3. The interviews lasted ∼60 minutes per user and session. The sessions were recorded and qualitatively analyzed to collect the gathered insights and the provided feedback.

The following discussion of the examples are based on findings from the user sessions, however, merged, condensed, and slightly adapted for clarity and brevity. The general user feedback, implemented improvements, and suggestions for future work are discussed separately after the examples.

5.1. Detecting Genres from Movie Posters

For the first scenario we selected an old Kaggle challenge [Kag]. The task is to create a model that will assign genres (as labels) to movies (as items), based on the movies' posters alone. The actual movie posters are available, and the ground truth is also included based on the genres for each movie on the IMDb platform [IMD]. Each of the four studied multi-labeling results represents the classification of one particular neural network composed of a pre-trained model combined with a custom classification head. Two of the models are derived from a TensorFlow tutorial [Mai19], which differ only in the used loss function of the model (BCE Loss and F1 Loss). The two other models also use the F1 Loss, but have twice the number of neurons in the custom head. Lastly, the Limit 3 model is further restricted to assign three genres per movie, as that is also the case in the ground truth.

Due to the unequal distribution of genres in the training data, we only considered the top 20 labels, judged by number of movies, as a small number of examples might be insufficient for a model to learn the features of a specific genre. We then removed the movies that were not assigned any of the remaining genres in the ground truth. The models were trained on the resulting training set of around 25,000 images. The visualized data in the tool is the validation set and consists of 1,408 previously unseen images.

Looking at the result comparison view (Figure 1) for this scenario, both users quickly identified that the label *Drama* was assigned most often over all multi-labeling results (▌**AQ 1.1**). Three of the models even assigned the label to every single item in the dataset, which was also correctly interpreted as explanation for the relatively low disagreement for the label (0.345) by one of the users. The other user further observed the label occurs only about half as often within the BCE Loss model and the ground truth. When selecting the label *Drama* in the ground truth (711), it becomes evident that those 711 items differ substantially from the 794 items that were assigned the label in the BCE Loss model (794). The overlap bar shows that less than half of the items are in both of these sets, indicating that the results are not as similar as the occurrence values make it seem (▌**AQ 1.2**).

One observation both users made in the graph visualizations is a fan-in pattern on the label *Drama*, meaning that many links exist from other labels towards *Drama*. Figure 1 shows this pattern for all multi-labeling results and the thickness of the incoming lines indicate that there are strong relations, up to a complete subset for some labels (▌**AQ 2.1**). This is not surprising for those results where *Drama* was assigned to all items and, therefore, all other labels were assigned together with *Drama*. But looking at the other multi-labeling results, we see the same pattern, albeit somewhat weaker for BCE Loss and the ground truth (▌**AQ 3.1**). We can conclude that *Drama* is a central label in all results (▌**AQ 2.2**) and the items it is assigned to are a superset of those for several other labels.

The genre *Western* in the ground truth does not have any links in the graph (with the default minimum edge weight of 0.4). Three of the other multi-labeling results, however, have links going out from the label. Interested in more details about this, one user selected the label *Western* in the ground truth. The 36 items are then selected and can be investigated further in the detail view (Figure 6). Whereas, in the ground truth, the majority of items were only assigned the *Western* label, the histograms for the other multi-labeling results show very different distributions (▌**AQ 3.1**) for those 36 images. For the F1 Loss and Double Neurons results, we can observe normal distributions around an average of three assigned labels. The Limit 3 result, as intended by the model, has assigned three labels to each item. This highlights different labeling strategies of the created models, none of which led to a high resemblance of the ground truth.

5.2. AI Food Recognition Challenge

Our second use case is the *Food Recognition Challenge* by AIcrowd [AIc]. For this challenge, the contestants were asked to

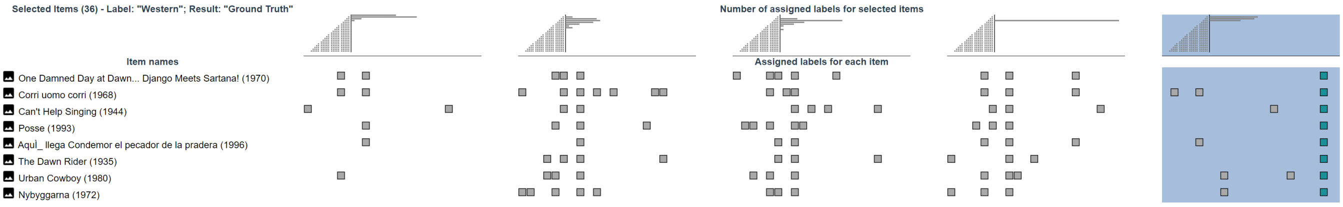

Selected Items (36) - Label: "Western"; Result: "Ground Truth"
Number of assigned labels for selected items
Item names
Assigned labels for each item
One Damned Day at Dawn... Django Meets Sartana! (1970)
Corri uomo corri (1968)
Can't Help Singing (1944)
Posse (1993)
AquÌ_ llega Condemor el pecador de la pradera (1996)
The Dawn Rider (1935)
Urban Cowboy (1980)
Nybyggarna (1972)

Figure 6: *A cutout of the detail view when selecting the 36 images with the label* Western *in the ground truth of our movie genre dataset. Histograms plotting the number of items onto the amount of labels assigned to them. Below are the first items and their label assignments in the multi-labeling results shown in the pixel maps.*

identify different foods on images. As the set of images used for the evaluation is kept secret, we did not have access to a ground truth for the evaluation dataset. However, we also incorporated a debug dataset that the participants could use to test their models. The debug dataset has been annotated by the organizers of the challenge and functions here as a ground truth. The raw images from the debug dataset are also available for us to show in our prototype. We manually selected five of the top 10 submissions that showed different labeling behavior and removed the remaining ones. For the debug dataset, we then also added the ground truth. The number of labels was reduced from several hundred to the 20 most frequent, as they likely have the biggest impact on the results. Among the two external users who worked with this dataset in our prototype, one was part of the evaluation team for the challenge, while the other ranked in the top 10 of the challenge submissions. Both users spent the majority of the time within the debug dataset and only switched to the evaluation dataset to confirm that their observations hold for that dataset as well.

One of the users right away mentioned the colors for the disagreement values to be of high interest for the analysis (▮ **AQ 1.2**). This also led him to start investigating the label *Water*. From looking at the occurrence value in the ground truth, he quickly inferred that *Water* is the most frequent label in the dataset, which aligned with his experience (▮ **AQ 1.1**).

Figure 2I.b shows that one of the strongest relationships in one of the submission was between the labels *Salad* and *Carrot* (▮ **AQ 2.1**). Selecting the 20 items for *Carrot* (20), the user noticed further incoming and outgoing links from the label *Mixed Salad*. He then continued his analysis of these relationships in the detail view. The user identified a pattern among the pixels maps: *Carrot* is not only strongly connected to *Salad* and *Mixed Salad* individually, but there is also a high number of items with all three labels assigned.

Both users were interested in the comparison between the different multi-labeling results on a holistic level and used the graph *diff* feature a lot for this. From the evaluator's perspective, our user selected the ground truth and compared it to the submissions, while constantly changing the minimum edge weight to check whether these differences appear for strong and weak overlap relations alike. The user who participated in the challenge selected one submission as the basis for the graph *diff*. He was curious to see how the selected model differs from others, especially, what the other models were assigning differently (▮ **AQ 3.1**).

The clustering result based on the aggregation of all multi-labeling results yields clusters that appear logical (▮ **AQ 2.3**) (Figure 2I.b). An interesting observation one of our users made is that the 3rd place submission has a lot of strong links within the clusters, while it has few that surpass the boundaries of clusters. This is a pattern that is not as apparent in any of the other results (▮ **AQ 3.2**).

In terms of different label assignment strategies, both users clearly pointed out two submissions that were different from the others. The submission in 10th place only assigned four of the 20 labels in the dataset at all (Figure 2I.b). The users assumed that the classifier does not know of the other labels and is just a very simple model. On the other extreme is the submission that ended up performing best in the challenge. The model has assigned the most labels by far, which can be seen in the occurrence values and the density of the graph (Figure 2I.b). Looking at some specific images in the detail view, one of our users confirmed that the assignments are mostly due to underfitting. An example of these faulty label assignments is shown in Figure 5 (▮ **AQ 1.3**). Our user from the evaluation team recognized this and mentioned that they adjusted the set of rules for the future challenges because of this behavior. He concluded that using our prototype would have helped him to identify this problem more quickly.

5.3. General Feedback

Based on the feedback from the first sessions, we made the interface more self-explanatory and added on-demand explanations as tooltips. The improvements were well received by the users in their second sessions. Moreover, the histograms were a suggestion from one user in the first session. He later used this feature as the main entry point towards analyzing a specific label. The clustering result based on the ground truth alone was another suggestion from one of the users. Also, the pixel map layout in the item detail view was a suggestion from the first interviews, which allowed for visual pattern search. That was not possible with our previous version.

In general, the users gave us positive feedback on our approach. They mentioned several ideas on how our system could help them, or other users, in working with machine learning approaches. The comparison of different models could help users decide which approach they should focus on in their development. Furthermore, if a ground truth or another baseline is available, our system would support the users to understand the multi-labeling results more profoundly, which would allow them to make more informed decisions to adjust a model. This could go as far as comparing the same model with different hyperparameter combinations. In the context of AIcrowd, the evaluation expert specifically mentioned using the system to check for problems within the dataset, and submissions, to improve future challenges.

The users also mentioned shortcomings of the prototype and made recommendations for future features. This includes improving the scalability of the approach with regard to the number of labels and multi-labeling results. Moreover, one user suggested sorting the items in the item detail view by disagreement between multi-labeling results, instead of displaying the items unordered.

6. Discussion and Limitations

The application examples and user feedback have demonstrated that our approach supports the initially formulated analysis questions (Section 3). We observed that specifically 🟥 **AQ 3.1** produces interesting insights in different ways. Moreover, 🟨 **AQ 1.1** and 🟨 **AQ 1.2** are very easy to answer with only a few quick glances at the result comparison view. In contrast, 🟧 **AQ 2.2** has only shown shallow insights.

As demonstrated, the approach is applicable to comparing multi-labeling results, however, with the constraint that the same set of items are labeled with the same set of labels in all results. This includes scenarios where different classification algorithms compete, where variants and configurations of one classification algorithm are tested, and all other scenarios where multiple sources suggest a classification using the same labels (e.g., human coders). In contrast, this excludes scenarios where each result is based on different items or the results use different labels, for instance, mining topics from text documents. While the relevant analysis questions might be the same or similar for these scenarios, it remains an open research question how similar analyses can be supported for scenarios without a predefined labeling scheme.

Visualization approaches often work best with a certain size of the datasets. We have designed our approach to show 10–30 labels and 3–10 different labeling results. For larger numbers of labels and results, vertical scrolling and long edges as well as visually squeezed columns or horizontal scrolling would significantly decrease the readability of the visualization. To address this issue, intelligent collapsing and expanding of rows and columns might increase the scalability by an order of magnitude. Ideas from *Table Lens* [RC94] can be applied to implement a multi-focus solution with different zoom levels for rows, columns, and cells. In addition, the computed clusters of labels or results could be used to summarize groups of rows or columns, respectively. Regarding the number of items, our approach is more flexible. In the examples, we have represented up to about 1,500 items; showing more items is only restricted at the moment by the prototype's performance when loading the detail view.

Representing overlap of label assignment as a graph structure is at the center of our approach. It provides the clear advantage of simplifying set overlap to a more abstract representation. This abstraction allows to reason about clusters of labels and to identify high-level similarities and differences between different multi-labeling results. However, this abstraction also comes along with two challenges. First, the relationships as a derived property as well as their bipartite drawing are not straightforward to interpret and require some explanation. The bipartite drawing is also limited to pairwise relationships between labels, while higher-order label combinations cannot be displayed. Second, the graph representation depends on an edge weight threshold and can become overly dense or sparse in some examples. We have already tried to address these challenges by providing textual and visual explanations of the encoding, integrating an interactive slider for the edge weight threshold, and applying a *diff* mode for identifying the exact differences between the graphs. Our application examples have shown that, generally, these measures work and relevant findings can be made using them.

7. Conclusion

With a focus on showing overlap relationships between labels, we have proposed a visualization approach to analyze and compare multi-labeling results of different algorithms and sources. The approach itself is model-agnostic and applicable to all multi-labeling results with the same labels and the same items. We have transformed this set visualization problem into a graph comparison problem and visualize the graphs in the columns of our tabular representation. Dedicated features relating to the clustering of labels (rows) and multi-labeling results (columns), as well as graph differences and interactive highlighting, support identifying and comparing structures in these label relationships. In the application examples, we have observed differences and similarities in label assignments and label relationships of multi-labeling results that would have remained hidden with traditional quantitative evaluation methods based on precision and recall. In the future, we would like to integrate the approach closer into a machine learning pipeline where the algorithms can be selected, adapted, and fine-tuned from within in the tool with visual feedback on the learning process and the general evolution of the model.

Acknowledgments

We are very grateful to AIcrowd for providing the dataset of the food recognition challenge and to the two external experts for their detailed feedback. This research was partly funded by MERCUR (project: "Vergleichende Analyse dynamischer Netzwerkstrukturen im Zusammenspiel statistischer und visueller Methoden").

References

[ABZD13] Abuthawabeh, Ala, Beck, Fabian, Zeckzer, Dirk, and Diehl, Stephan. "Finding Structures in Multi-Type Code Couplings with Node-Link and Matrix Visualizations". *Proceedings of the 1st IEEE Working Conference on Software Visualization*. 2013, 1–10. doi: 10.1109/VISSOFT.2013.6650530 3.

[AHH*14] Alsallakh, Bilal, Hanbury, Allan, Hauser, Helwig, et al. "Visual Methods for Analyzing Probabilistic Classification Data". *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), 1703–1712. issn: 1941-0506. doi: 10.1109/TVCG.2014.2346660 2.

[AIc] AIcrowd. *Food Recognition Challenge*. https://www.aicrowd.com/challenges/food-recognition-challenge Accessed: March, 2021 7.

[AJY*18] Alsallakh, Bilal, Jourabloo, Amin, Ye, Mao, et al. "Do Convolutional Neural Networks Learn Class Hierarchy?": *IEEE Transactions on Visualization and Computer Graphics* 24.1 (2018), 152–162. issn: 1941-0506. doi: 10.1109/TVCG.2017.2744683 2.

[AMA*16] Alsallakh, Bilal, Micallef, Luana, Aigner, Wolfgang, et al. "The State-of-the-Art of Set Visualization". *Computer Graphics Forum* 35.1 (2016), 234–260. doi: 10.1111/cgf.12722 2.

[BPD11] Beck, Fabian, Petkov, Radoslav, and Diehl, Stephan. "Visually exploring multi-dimensional code couplings". *2011 6th International Workshop on Visualizing Software for Understanding and Analysis (VISSOFT)*. IEEE. 2011, 1–8. doi: 10.1109/VISSOF.2011.6069455 3, 5.

[BVB*11] Burch, Michael, Vehlow, Corinna, Beck, Fabian, et al. "Parallel Edge Splatting for Scalable Dynamic Graph Visualization". *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), 2344–2353. issn: 1941-0506. doi: 10.1109/TVCG.2011.226 3.

[CLG16] Cao, Nan, Lin, Yu-Ru, and Gotz, David. "UnTangle Map: Visual Analysis of Probabilistic Multi-Label Data". *IEEE Transactions on Visualization and Computer Graphics* 22.2 (2016), 1149–1163. issn: 1941-0506. doi: 10.1109/TVCG.2015.2424878 2.

[GAW*11] Gleicher, Michael, Albers, Danielle, Walker, Rick, et al. "Visual Comparison for Information Visualization". *Information Visualization* 10.4 (2011), 289–309. issn: 1473-8716. doi: 10.1177/1473871611416549 2.

[GBD09] Greilich, Martin, Burch, Michael, and Diehl, Stephan. "Visualizing the Evolution of Compound Digraphs with TimeArcTrees". *Computer Graphics Forum* 28.3 (2009), 975–982. doi: https://doi.org/10.1111/j.1467-8659.2009.01451.x 2, 5.

[IMD] IMDb.com. *Internet Movie Database*. https://www.imdb.com/. Accessed: 2021-03-26 7.

[Kag] Kaggle. *Movie Genre from its Poster*. https://www.kaggle.com/neha1703/movie-genre-from-its-poster. Accessed: 2021-03-26 7.

[KR09] Kaufman, Leonard and Rousseeuw, Peter J. *Finding groups in data: an introduction to cluster analysis*. Vol. 344. John Wiley & Sons, 2009 6.

[LGS*14] Lex, Alexander, Gehlenborg, Nils, Strobelt, Hendrik, et al. "UpSet: Visualization of intersecting sets". *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), 1983–1992. doi: 10.1109/TVCG.2014.2346248 2.

[LJS20] L'Yi, Sehi, Jo, Jaemin, and Seo, Jinwook. "Comparative Layouts Revisited: Design Space, Guidelines, and Future Directions". *arXiv preprint arXiv:2009.00192* (2020). doi: 10.1109/TVCG.2020.3030419 2.

[Mai19] Maiza, Ashref. *Train a multi-label image classifier with macro soft-F1 loss in TensorFlow 2.0*. https://github.com/ashrefm/multi-label-soft-f1. 2019 7.

[MGM*19] McGee, Fintan, Ghoniem, Mohammad, Melançon, Guy, et al. "The State of the Art in Multilayer Network Visualization". *Computer Graphics Forum* 38.6 (2019), 125–149. issn: 1467-8659. doi: 10.1111/cgf.13610 2.

[MKGD12] Madjarov, Gjorgji, Kocev, Dragi, Gjorgjevikj, Dejan, and Džeroski, Sašo. "An extensive experimental comparison of methods for multi-label learning". *Pattern Recognition* 45.9 (2012). Best Papers of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA'2011), 3084–3104. issn: 0031-3203. doi: https://doi.org/10.1016/j.patcog.2012.03.004 2.

[PLHL19] Park, Chanhee, Lee, Jina, Han, Hyunwoo, and Lee, Kyungwon. "ComDia+: An Interactive Visual Analytics System for Comparing, Diagnosing, and Improving Multiclass Classifiers". *2019 IEEE Pacific Visualization Symposium (PacificVis)*. 2019, 313–317. doi: 10.1109/PacificVis.2019.00044 2.

[RAL*17] Ren, Donghao, Amershi, Saleema, Lee, Bongshin, et al. "Squares: Supporting Interactive Performance Analysis for Multiclass Classifiers". *IEEE Transactions on Visualization and Computer Graphics* 23.1 (2017), 61–70. issn: 1941-0506. doi: 10.1109/TVCG.2016.2598828 2.

[RC94] Rao, Ramana and Card, Stuart K. "The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information". *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1994, 318–322. isbn: 0-89791-650-6. doi: 10.1145/191666.191776 9.

[TVB*20] Theissler, Andreas, Vollert, Simon, Benz, Patrick, et al. "ML-ModelExplorer: An Explorative Model-Agnostic Approach to Evaluate and Compare Multi-class Classifiers". *Machine Learning and Knowledge Extraction*. 2020, 281–300. isbn: 978-3-030-57321-8 2.

[VBP*19] Valdivia, Paola, Buono, Paolo, Plaisant, Catherine, et al. "Analyzing Dynamic Hypergraphs with Parallel Aggregated Ordered Hypergraph Visualization". *IEEE Transactions on Visualization and Computer Graphics* (2019). doi: 10.1109/TVCG.2019.2933196 2.

[Wil12] Wilkinson, Leland. "Exact and Approximate Area-Proportional Circular Venn and Euler Diagrams". *IEEE Transactions on Visualization and Computer Graphics* 18.2 (2012), 321–331. issn: 1941-0506. doi: 10.1109/TVCG.2011.56 2.

[YEB16] Yalcin, M Adil, Elmqvist, Niklas, and Bederson, Benjamin B. "AggreSet: Rich and scalable set exploration using visualizations of element aggregations". *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), 688–697. doi: 10.1109/TVCG.2015.2467051 2.