Confusion Matrix Viz

Eva Kaderabek* Panittha Suwannajang†

Department of Computer Science
University of Vienna

ABSTRACT

Confusion matrices contain a large amount of useful information that can lead the improvement of machine learning systems by hinting to possible error sources and identifying areas that need further enhancement. Therefore a tool that can be used without investing too much time reading manuals and that is able to provide all required data in a clear way can disburden the process of debugging immense. The application presented in this paper tries to fulfill these demands. The center of ConfusionMatrixViz is the common heat map, derived from the given confusion matrix, reorderable and easy to be explored. Additionally bar charts present important values that can be read out of the matrix. The opportunity to analyse two matrices at the same time and derive a third by combining the two makes ConfusionMatrixViz particularly helpful for reviews concerning the improvement of a system. The user study confirms user friendliness.

1 INTRODUCTION

The skill to learn and improve on certain tasks is becoming increasingly important for computational tools. Therefore a growing number of different sorts of machine learning algorithms are evolving, each with the target to augment its efficiency in its particular domain independently by deriving axioms from the input-dataset, with which it can differentiate between entities with increasing accuracy. There are different ways to analyse the efficiency of such algorithm, one of them is to look at the output. Typically for this analysis is to organise the output in a confusion matrix that shows the number of entities given to and seen by the algorithm in a two-dimensional field.

Thereby standard classification contains two possible classes and is binary, examining e.g. cancer or not cancer [6]. Its result designed as a confusion matrix shows in well-arranged way elementary categories for the evaluation, namely true positive, true negative, false positive and false negative. These values are the basis for calculated rates like accuracy and precision.

Multiclass classifications differ between any number of classes over two, what exponentiates the complexity and the possible sources of error [6]. With an increasing number of classes the resulting confusion matrix becomes harder to interpret. Thus particular tools have been developed to extract as much useful information concerning the performance of an algorithm as possible. This is the area in which given tool, that will be presented here, shall be deployed. ConfusionMatrixViz is an online tool that provides helpful graphics and values for the analysis of confusion matrices with more than two, ideally more than five classes. It aims primarily at experts in the field and developers whom it should enable to identify the source of classification problems fast by filtering out poorly performing classes and the classes they were mistaken as. These users are provided with an upload function that gives them the chance to examine their own matrices. A second and as secondary seen user group consists of interested people, especially students, who got confronted with the topic and want to try out, which conclusions can be drawn from a confusion matrix. For them the tool provides example matrices.

The tool works with CSV-files that are arranged in the following way: the first row contains the class names, in the rows beneath are the values of the actual matrix. The main goal of the tool is to provide an opportunity to find performance flaws per overview, looking at the whole matrix as a heat map, and per detail search, fetching the derived values for each class.

In this paper three main topics will be discussed. First it will introduce ConfusionMatrixViz by reviewing the process that led to the current design and functions as well as showing its functionality. Second possible ways of interaction are presented to illustrate its usage. Third the setup and results of a small user study will demonstrate the degree of usability and usefulness of the system.

2 RELATED WORK

The most common way to depict a confusion matrix is by keeping its shape and using the values of each field to construe a heat map from it. This particular approach is to be found in many developed tools, if not as main graph then at least as accessory. Another way is to change the shape of the matrix trying to emphasize the relation between certain fields or to make completely different graphs based on the matrix. Works of both scopes have been taken into consideration during planning phase.

As the main task of the tool has changed late in the development period, many innovative ideas that could have been found in various works, some of them described subsequently, were dismissed from the beginning on. As the main target was to generate a tool, that specialises in visualising matrices as big as 1000×1000 fields every idea concerning single classes was given up as the data of one class out of 1000 seemed to be pointless to further investigate. The new target is to specialise in the visualisation of up to 50×50 matrices. Therefore single class views became meaningful. In the recent tool this data is visualized by bar charts as being the clearest way to show quantitative data in relation with an ordinal dimension.

2.1 Reordering the Matrix

Talbot [6] have introduced EnsembleMatrix in 2009, a system that enables its user to visualise a rash of confusion matrices from different classifiers on one dashboard. All matrices are depicted as heat maps. A linear combination widget gives the user the possibility to change the weight of classifiers and thereby the heat maps. As ConfusionMatrixViz solely works with the confusion matrix and without including meta data such function was not intended.

To reorder the matrix EnsembleMatrix uses the barycenter heuristic, that derives an adjacency matrix [4] and then reorders it by grouping clusters. With ConfusionMatrixViz it is possible to reorder the confusion matrix, although here the performance rate of each class is the linking factor instead of the relationship to each other. Thereby ConfusionMatrixViz construes a ranking order with the best performing classes concentrated in the left upper corner.

EnsembleMatrix further on offers the possibility to divide the matrix into partitions either due to an integrated function or arbitrary. Both enables the user to specify the cluster search.

*e-mail: a0425336@unet.univie.ac.at
†e-mail: a1263951@unet.univie.ac.at

Behrisch [2] explain in their paper about matrix reordering methods different approaches to alter matrices of tabular data and adjacency matrices of networks. Especially the reordering of adjacency matrices seemed to be relevant to this project. Although the analysed matrices of networks are symmetrical, the investigation of meaningful patterns retains its significance. In confusion matrices symmetrical shapes point out correlations that are specifically important to detect possible disturbing classifiers. By finding significant shapes other than the diagonal of true positive values it can be possible to find wide interdependencies that affect the accuracy of the algorithm. Block Patterns and Off-diagonal Block Patterns are the most common to be found after clustering in confusion matrices and are hinting to a strong interrelation between a bigger group of classes. Line or Star Patterns would be a sign for a class that either overshadows the rest, cannot be identified at all or a combination of both.

In our project the development time was not long enough to further elaborate on this topic, but this sure is worth a second look. The interim solution offers a good starting point for different implementations.

2.2 Reshaping the Matrix

A solution found and tested by Ren [5] does not keep the matrix form at all. Instead it divides the cubes (each single field) from each other and tries to find a different arrangement that allows useful insights. Their system Squares is the result of a small study among experts, in which they tried to find the most important requirements for an analysis tool. Per questionnaire they found three main tasks a good tool must have: It should be able to show performance at different levels of detail, with overall and class level as the most important, followed by instance level. It should be adjustable to different performance metrics making it useful for systems that are barely allowed to fail as a false classification can cause irreversible damage as well as for systems whose missing rate is irrelevant. And at last it should include the input data to support the understandability of mistakes.

Their solution is to design every class like a bar chart with vertical bars that can contain of boxes/cubes, strips and stacks to show single or grouped instances of the dataset. Colour makes the boxes, strips or stacks allocatable, the filling signals their nature, whether they have been recognized correctly or not. As the forms on the left side of the middle line represent the fails and the forms on the right the hits this particular visualisation makes it very easy to compare false positives and false negatives which was one of their intentions.

A user test, in which they let test persons compare their representation to a classical heat map, confirms the higher usability of their squares. Nevertheless its unfamiliarity and therefore higher learning curve disqualified the idea for our system as ConfusionMatrixViz shall too be suitable for students and people who are hardly familiar with confusion matrices. A confusion matrix is easy to be read and contains all necessary informations. To fulfill our main goal to design a tool that is very simple and does not need any further explanation ConfusionMatrixViz offers all necessary measures in bar charts beneath the heat map(-s) that is (are) the main view. As Squares, these bar charts offer data on at least two levels, the overall rate and the values and rates for each class in particular. Further on it offers highscores with the five classes that performed best, had the highest true positive or negative values, or were the worst, having the highest false positive or negative values.

Alsallakh [1] went comparably innovative ways by searching for new possibilities to depict confusion matrices. Their solution is the confusion wheel, with the classes on the side and the relations as strokes with different width in the middle. Here the four different derivated values true positive, true negative, false positive and false negative, are encoded with different colours. Its biggest advantage of course is the middle in which the relations between classes are easy to be seen, although the assignment of wide and more narrow end can be misinterpreted easily. Maybe directed strokes with constant width would have been much easier to be understood. Additionally the nature of the connection is not that simple to be determined.

Another proposal for the assignment of colours is to mix their domains, colouring according to class, but reserving two colours to stress false positives and false negatives in a class. This is very confusing as it forces the brain to jump between domains. Here the solution of Squares would have been better, colouring all blocks according to their class, even if they are allocated wrongly.

This possibility was dismissed due to the same reason as any adaption of the Squares system. It would have needed further explanation.

3 Approach

Our visualization is focused on two main components, namely visualization of a confusion matrix itself and visualization of calculated values concerning a specific confusion matrix. Furthermore, there are two possibilities of visualization, a visualization of a single matrix and a visualization of comparing two matrices. To make visualization as effective as possible, our tool also offers explorative functionalities, like sort functions and tooltips. Last but not least, users can not only upload their own dataset, but users can also download the visualized metrics as well.

3.1 Visualization of a confusion matrix

At the design stage, when we were trying to decide, which way of encoding data is the most effective, we came up with a few ideas, such as a heat map diagram value encoded by saturation, a squared table with encoded value by size in each matrix cells , gantt chart style, where a fully filled matrix means the value is equal 100%, and another fancy idea Confusion Wheel a totally different approach of displaying confusion matrix, invented by a research team from TU-Wien [1]. See Fig. 1.

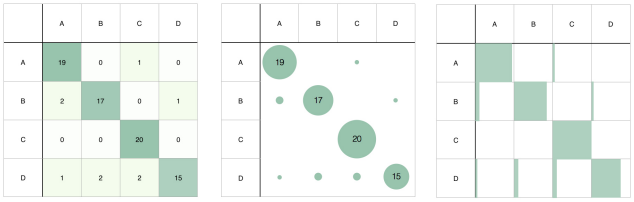

Figure 1: Single view; Left: Heatmap encoded with saturation. Center: Matrix encoded with size and Right: Gantt chart style

After debating, the Confusion Wheel is the first to fail of our list, as fancy as it looks, the readability of this confusion wheel in our opinion is fairly low, it is quite confusing actually. Matrix encoded by size, when it stands alone, it is quite an acceptable solution, but when compared to a heat map diagram, even though matrix encoded by size looks more interesting and exciting, but still the readability could not beat what a simple heat map diagram has to offer. Gantt chart style is quite an interesting idea, it is something that we are not often see it used in this purpose, we think gantt chart could be a good exploration, but that is also means higher risk of it being more difficult to understand. As our goal is to make it as easy as possible for the user to understand the visualization we decided to stick with a heat map diagram encoded with saturation.

3.2 Visualization of comparing two matrices

We came up with three different views and two different functionalities of comparing two matrices. These three views are Side-by-Side approach, Tab approach, and Movable ruler approach. It is easy

to identify which view is most effective, which is Side-by-Side approach. And the two functionalities are one matrix on top of the other one and Matrix_1 subtracted by Matrix_2.

Matrix_1 on top of Matrix_2, both matrices encoded with different colors and with 50% opacity, this way the hint of the new create color might be useful on telling which matrix does well than the other. The idea of this comparison is interesting, nevertheless when we tried to use this comparison method with real data, it turns out to be impractical. It was bought to our attention that human brain can discriminate only about 6 to 12 hue. The way we compared the matrices, it could produce uncountable hue range. With that reason this function of comparison is dropped.

Matrix_1 subtracted by Matrix_2, the result matrix presents the differentiate, in which classes Matrix_1 did better than Matrix_2 and also the other way around. We have not yet found disadvantage of this approach, rather quite an interesting result matrix, just the right solution for comparing two matrices.

3.3 Visualization of calculated values

The same thinking applies here, we decided to use bar chart, rather than line chart, scatter plot or pie chart. For so many reasons pie chart was actually never made to our list of choices. That reasons are for example radius of a circle is simple more difficult for human brain to evaluate the value and the fact that pie chart is best suited to be used for presentation of data that sum up to 100 percent. Line chart is also not suited for our case, because the calculated values are classified into classes and the values are not continuous. Deciding between scatter plot and bar chart is more challenging. With scatter plot we could present several dimensions of data in one graph, that would save us a lot of space, but in our humble opinion bar chart is the easiest chart to read, one focused value at a time.

3.4 Dashboard design

As for our dashboard design, we first identify primary functionalities and secondary functionalities. As result, primary functionalities are to display a single confusion matrix, display comparison view between two matrices and display further information about a selected confusion matrix in detail. Secondary functionalities are possibility of selecting sample datasets, upload datasets, sorting, reset and download. Final layout of our index page can be seen in Fig. 2.

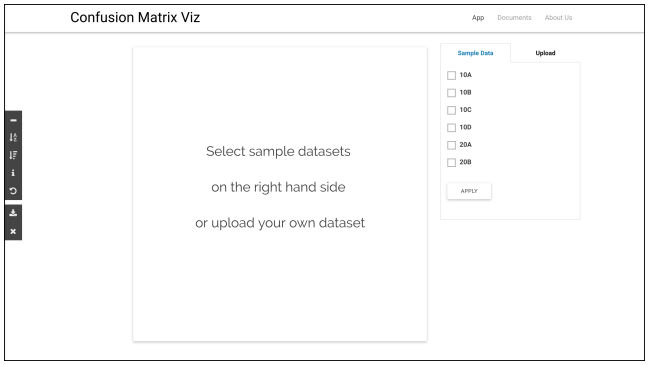

Figure 2: Index page

We use comparison view as a ground rule to design the dashboard. Our first design attempt was to keep everything within one screen, no need for scrolling down. This idea is smashed when we realized that there are a lot of calculated values to be presented. Primary functionalities taken most space of the screen, yet secondary functionalities should also to be visible and easy to access at all time (see Fig. 3). That is why general functionalities lays as floating icons on the left hand side, but functions like selecting sample dataset and uploading data required more space than just a button, these functions are displayed on the right hand side as a box with Sample Data and Upload tab. Calculated values are presents beneath the matrix diagram, with radio button, where user can select with type value to be presented.

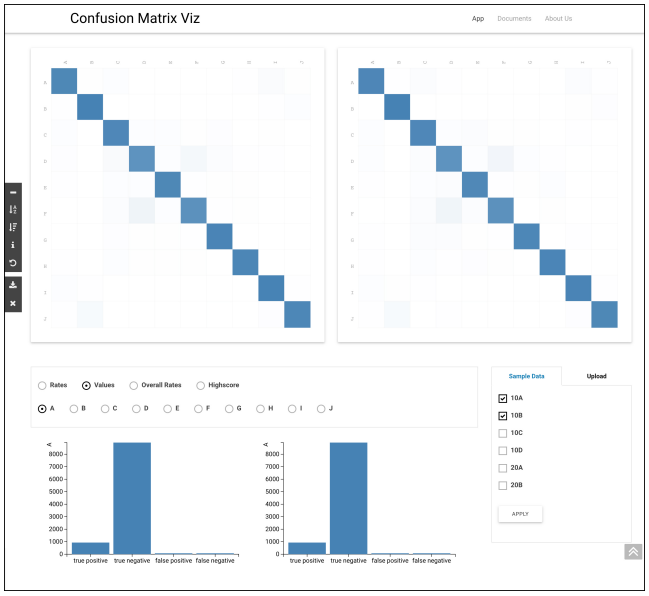

Figure 3: Comparison view with true positive, true negative, false positive and false negative values of each matrix

3.5 Sorting

Sorting offers a great way of exploring dataset. Our tool offers three different ways of sorting, sort by cluster, sort by class title in ascending alphabetical order and sort by selected row or column, by clicking on a row or column. Sorting by cluster is based on accuracy of each classes.

3.6 Zoom

As our old goal was to visualize confusion matrix with the size up to 1000×1000, a zoom function was definitely required, in order to explore such a large matrix. But after discussed the tasks in greater details with real users, our main focus is shifted to mainly visualize much smaller size matrices like 10×10. So, we eliminate the zoom function and added tooltips on mouse hover over each matrix instead. Aimed matrix size is reduced, it is compensated by showing existing information in greater details. It could be said that quantity is reduced, so we higher the quality.

4 Implementation

ConfusionMatrixViz is a single page web application, implemented with HTML, Javascript and CSS. One of the challenges that web developers have to face nowaday is responsive website, the web page has to look good on all devices no matter what is the size of the screen. We use a well known CSS framework; Bootstrap, to scales our application and keep it looks good across devices. jQuery, a feature-rich javascript library is used in conjunction with the good old javascript, to allow user interactions. With help of D3, another powerful javascript library, visualizing dynamic data became much easier, D3 brings data to life. All in all, Javascript is being used on both frontend and backend development.

4.1 Challenges

Various problems have had occurred during implementation phase. They are listed below:

	A	B
A	8	2
B	1	9

Table 1: Format of input at first attempt

A	B
8	2
1	9

Table 2: Input format, the tool required

- Dealing with D3 is a learning curve, the curve is quite steep at the beginning, but after a lot of trial and errors we get a grip of it.
- The task of generating a heat map was conjoined with an unexpected problem; finding the right format in which the input data should be shaped. At first we thought it makes most sense to have the input in matrix manner, first row and column are the header, squared inner matrix cell are the value (see Table 1). But this way, say in a 10 x 10 matrix, we would have 10 elements on the first row and 11 elements on the rest 10 rows. We then decided that we should leave the first column out of our input, because classes on actual and predicted axises are the same. We now left with 10 columns by 11 rows (see Table 2). After spending some time trying to generate the matrix with these 10 x 11 values, we failed, we then tried to find a another way. Our solution is to convert the input data from Table 2 into Table 3 with a simple block of code, see Listing 1.

.

Listing 1: Convert input data into desired format

```
function convertData(csv) {
    var arr = [];
    var h = d3.keys(csv[0]);

    for (var i = 0; i < h.length; i++) {
        for (var j = 0; j < h.length; j++) {
            arr.push({actual: h[j],
            predict: h[i],
            value: +csv[i][h[j]]});
        }
    }

    return arr;
}
```

- Generating 1000×1000 Confusion was one of the biggest problem we faced, but this problem is eliminated when we changed our main goal.
- As we were developing the tool, each of us work on separaate tasks, we then have faced problem in merging the source code into one. It is clear that every person has different style of writing code. We could have avoid this problem before it happens, by simply define general rules which both of us should follow.
- We stated above that responsive website is definitely a necessary feature, which requires a lot of our attention. Instead of simply define, $varwidth = 600px$ for example, we needed to always make sure to set the variable width to current width of the element depends on the screen by calling Listing 2.

.

Listing 2: Get element current width

```
var width = $(element).width();
```

actual	predict	value
A	A	8
B	A	1
A	B	2
B	B	9

Table 3: Format, in which the input data will be converted into

5 Results

To determine the usability of our tool two different simulated usage scenarios were played through as well as a qualitative Observation with voluntary participants was done. Each usage scenario shall represent a typical user out of the primary user group and a basic task they want to fulfill. The user test was performed in familiar environment for the participants with strong involvement of the tester. It can be seen as test runs with users that have never seen the system before and therefore will react solely based on their assumptions. Thereby we wanted to determine, how user friendly and simple the system is. Hence the usage scenarios will show the procedure how the developers have planned it whereas the user test will reveal how a person, who was never before confronted with the system, will try to extract information.

5.1 Usage Scenario

For the user scenario the planned way how a typical user can use ConfusionMatrixViz will be presented. The first scenario will show the handling of one matrix, the second will focus on the usage with two matrices.

5.1.1 Find Problematic Classes in One Matrix

Diana Hrubicek, 25 years old, is student of computer science and wants to develop an algorithm that recognizes particular shapes in pictures as the entities they are representing in reality. It is the project for her master degree. Dianas goal is to find a tool that analyses confusion matrices in a fast and simple way without requiring many adjustments or definitions for filters. Her main task is to find singular bad performing classes fast so that she can concentrate her efforts on them.

She loads up the most recent confusion matrix her system construed. As can be seen in Fig. 4 and Fig. 5.

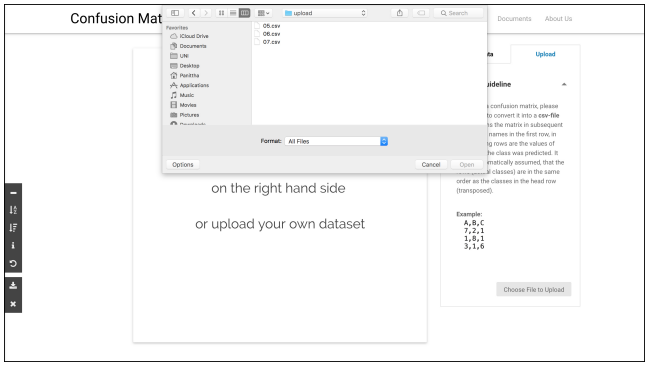

Figure 4: Upload dataset from user local storage

The heat map shows her a good overall accuracy of her system. As she scrolls down, radio buttons enable her to access the data

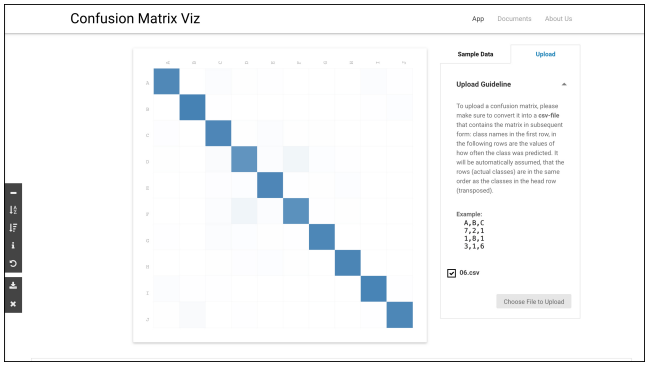

Figure 5: Visualization of uploaded dataset

derived from the matrix (Fig. 6). As she wants to focus on the worst performing classes she chooses Highscore (Fig. 7).

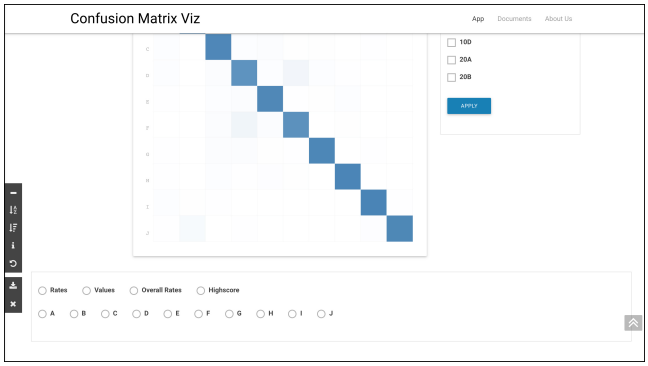

Figure 6: Visualization of uploaded dataset

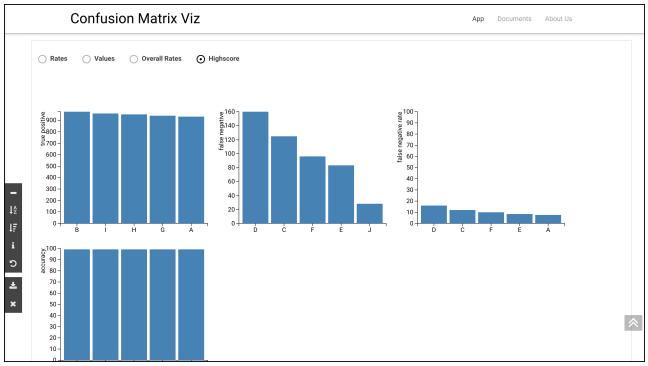

Figure 7: Visualization of calculated values; Highscore

The bar charts show her that class D has the highest false negative value and highest false negative rate (miss rate). To find out more about class D she looks at the rates in particular. The rates show that the false positive and the false negative rates are nearly equal (Fig. 8).

She concludes that D has problems with assigning as well as excluding entities as its instances. The negative predictive value is significantly higher than the positive predictive value, therefore the main problem seems to be the assignment of entities. The next step to improve her system will be to focus on the active classifiers of D.

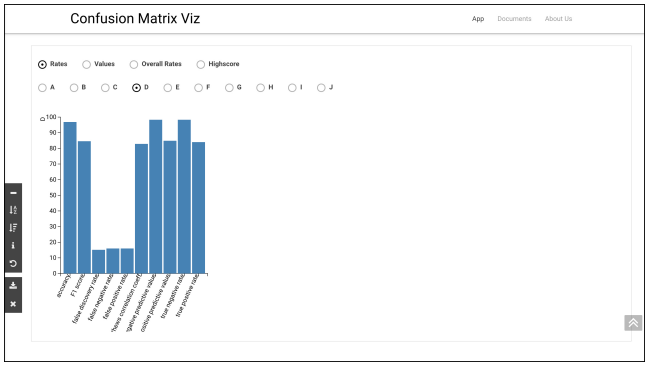

Figure 8: Visualization of rates of class D

5.1.2 Compare Two Matrices

Olga Svoboda, 51 years old, works for a software development company and is currently assigned to a project group working on an algorithm for facebook to classify comments on posts, especially harmful ones, and categorize them according to their offense. To retrace the development she wants to compare an old confusion matrix with one that was produced after significant changes in the system. Her goal is to produce a meaningful visualisation of a combination of both matrices that show the most significant differences.

To get the visualisation she uploads both matrices to ConfusionMatrixViz (Fig. 9).

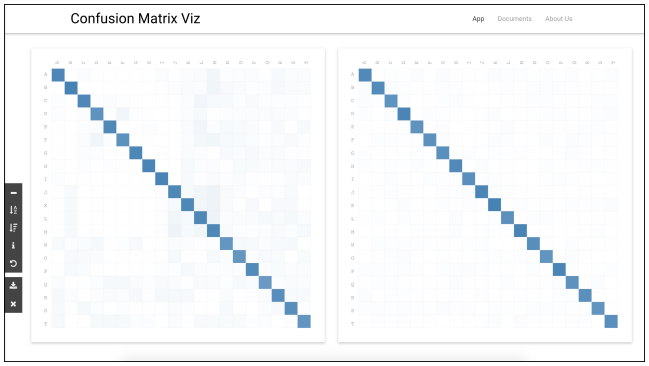

Figure 9: Visualization of two confusion matrices

To combine the matrices she chooses to use the A-minus-B-function offered on the sidebar (Fig. 10).

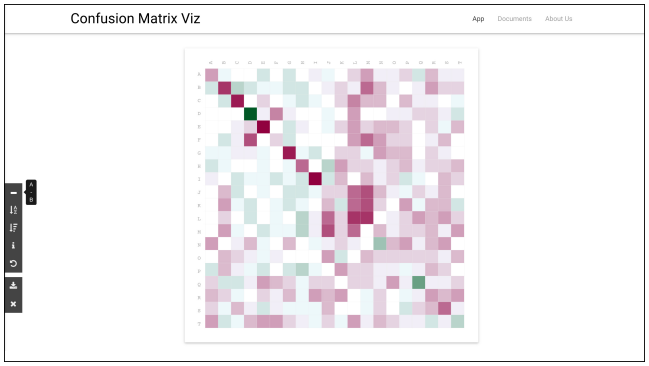

Figure 10: Visualization of A-minus-B-function

The two different colours show where the value of the first or

the second matrix was higher, the saturation clarifies how big the gap was. Seeing the combination especially on the diagonal of true positive values Olga can determine a major improvement of classes B and C, a setback for class D and so on. Latter classes do not show any striking changes. The overall rates show a significant improvement in terms of the false negative rate (Fig. 11).

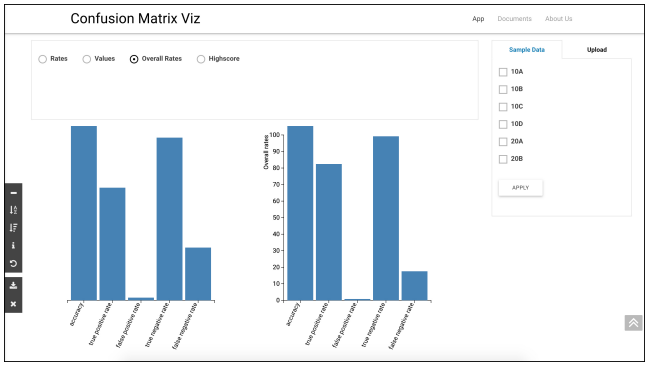

Figure 11: Overall rates of both matrices

The rates per class show the improvement per class (Fig. 12). Conspicuous classes can be selected to compare them (Fig. 13).

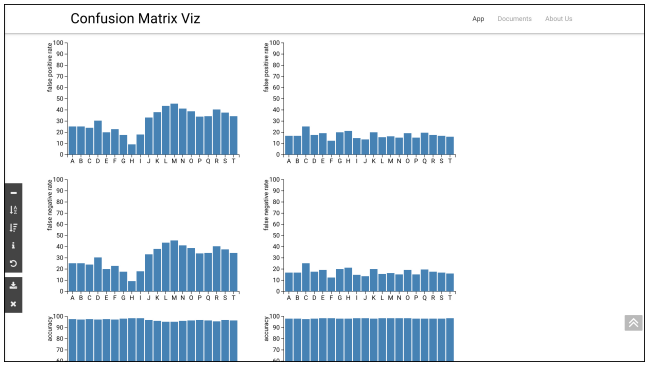

Figure 12: Further rates of both matrices; False positive and false negative rate

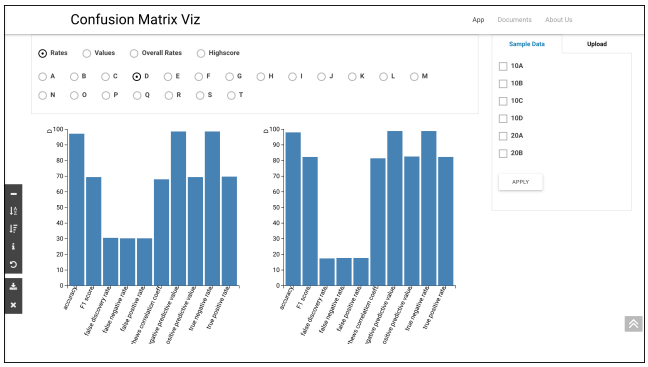

Figure 13: All rates of class D in particular

Olga sees that class D has improved its performance although it has 76 less hits than it had before. By studying the matrix she realizes that instances of D were very often mistaken as F. Recognising a symmetry she realizes that F was too very often mistaken as D.

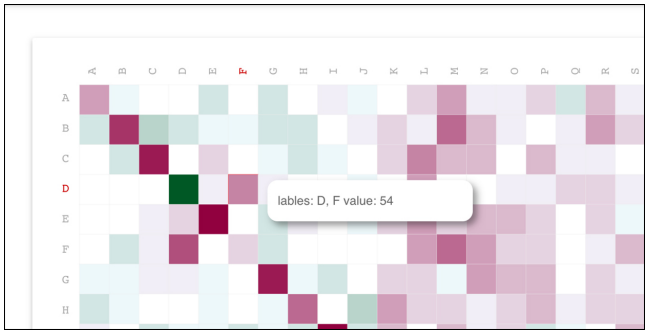

Figure 14: Hovering through A-minus-B resulted matrix

She decides to concentrate on this particular issue in continuation. See Fig. 14 and Fig. 15

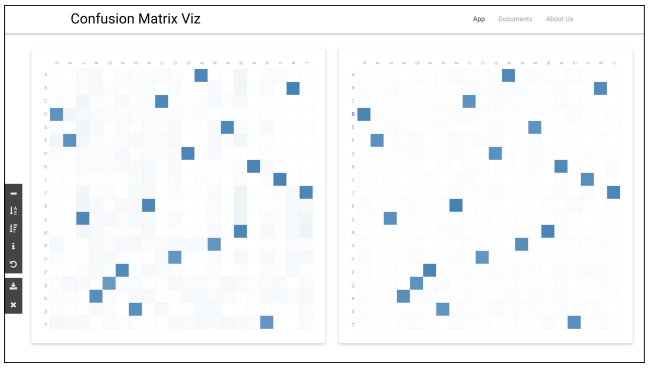

Figure 15: Row D sorted in descending order

5.2 Evaluation

To test out the tool it would have been ideal to define another analysis tool which task is comparable to the current, develop tasks that has to be completed on both systems and find people actively working in the area of expertise to fulfill the tasks. The time each person needs to conclude an exercise could have been measured and complemented by statements of the person concerning their usage experience. Ren [5] used this approach to confirm the advantage of their tool Squares. It is common practice to compare the new system with existing ones to legitimate its existence.

In present case comparable tools and experts are not at hand, hence the study to test out the system has been defined as a qualitative one using a think-aloud protocol [3]. Three participants were given a set of four different tasks, that were explained to them separately, after each task the observation was interrupted to explain the next exercise. As the participants tried to fulfill the task they were encouraged to talk about their thinking process and their concerns. Tips were given during the observation and reasons were discussed how the participant could have reached the same conclusion.

The tasks were:

- T1: Having one matrix, find out the worst performing class.
- T2: Having one matrix, find out the class that the solution class from T1 was most mistaken as.
- T3: From two matrices, find out which one shows a better overall performance.

- T4: From two matrices, given that they are drawn from different times in progress, see which classes have improved.

The participants each had no special expertise in respect to machine learning or computers in general. Therefore they represent our secondary user group, interested people that want to explore.

5.2.1 Execution

The study was performed with three different participants in their homes, a familiar environment for them. To prepare them for the test the meaning and usage of confusion matrices were explained to them till they showed in the conversation they have understood. Next the web page in question was shown without any explanations to not influence their learning process. The first task was presented to them and they began to speculate about possible approaches.

Each step was discussed so that the tester got an insight in the perception of a person that is using the tool the first time. If the participant got stuck and did not know how to proceed a possible next step was suggested to them and the reason why they would have never got it on their own was discussed.

5.2.2 Result and Feedback

The general impression of the tester was that ConfusionMatrixViz is easy to be understood but certain tasks are hard to fulfill because the way to the solution needs background information that the participants did not have. Additionally the system still has some glitches and quirks that required a person to be vastly familiar with the program.

During the observation the following problems arose:

- The possibility to combine two matrices was not clear right away and was discovered accidentally by trying out buttons.
- The combined matrix needed some explanation to make clear its purpose and how to understand it.
- With two confusion matrices, to distinguish between the bar charts belonging to one or the other matrix invoke uncertainty

These matters were clear right away:

- The users could upload the CSV-file they wanted or choose a matrix from the sample data.
- Reordering the matrix was mostly (2 out of 3) discovered instantly.
- The tooltips that appeared by hover over one of the heat map fields turned out to be the most helpful tool to solve all four tasks.

In conclusion ConfusionMatrixViz proved to be a simple applicable tool to provide a deeper insight into the facts hidden in a confusion matrix.

6 Discussion

ConfusionMatrixViz is a tool that offers the possibility to analyse confusion matrices of multiclass classification systems. Its design is based on a classification of functionalities into primary and secondary. The main goal was to keep its appearance as simple as possible so that the user cannot be overwhelmed by options. The center of the system is the heat map in which the confusion matrix, either uploaded or chosen from the samples, is turned. Derived values have been banned to the bottom of the page. Being visualized as bar charts they can be easily understood and compared.

The short survey to the usability of ConfusionMatrixViz showed that the tool does not need a lot of explanation and is suitable to give overall insights into confusion matrices. It is not that convenient to tackle specific problems as its graphs need to be thoughtfully combined with others to give the answer to a certain question.

6.1 Strengths and Weaknesses

As the little tags that can be found hovering above the heat map provide a good way to examine the node of two classes the heat map is the most important feature of ConfusionMatrixViz. The combination by dividing the right from the left matrix gives a clear overview concerning the differences and bring a valuable view.

The decision to keep heat maps as main graphs instead of developing innovative new ways to alter a confusion matrix proved itself right as it holds low the learning curve and can be understood and used instantly even by users that are not particularly familiar with the topic.

As the evaluation, especially the user test showed, the page lacks of descriptions and many functionalities are to be discovered via trying out. Additionally it misses a way to combine bar charts and therefore make bars directly comparison. The bars too do not say which precise value is hidden in them, only the y-axis gives some perspective.

6.2 Conclusions

The project description demanded a system that makes big matrices 1000×1000 accessible. A conversation with an actual potential user and the presentation feedback of the lecturer changed the direction to a more realistic target. Thereby single class views became more important. It must be said that the current tool is not at all exploiting the possibilities it has, especially concerning the bar charts. New functionalities can make it possible to expand even deeper into the derived numbers. The focus should have shifted to this area much earlier.

The technology used primarily to construct the graphs, namely d3, needed a lot of research to be employed, getting it to work consumed most of the time. In retrospective it would have been a good idea to give up on d3 at an early stage of the project and find another way to construct charts.

Overall the project raised more interest into the analysis of confusion matrices as they seem to contain much more information than our tool is currently able to extract.

7 Task Separation

After evaluation of our first prototype, we found out that the prototype was far from being a good tool, so we decided to make a lot of changes, to us it feels almost like we have to totally redo the prototype. What we did for this milestone is equaled to the whole project.

Panittha is responsible for visualizing confusion matrices, including comparison view and function and sorting. Eva is responsible for all the calculation and its visualization. The web page layout and merging our work into one, is done by Panittha. Eva on the other hand is very good with writing, she wrote most part of this paper, which are Abstract, Introduction, Related work, Results and Discussion. Approach and Implementation section is written by Panittha.

Acknowledgments

The authors wish to thank Peter Ruch, for his insight. It helps us to develop hopefully a desired and useful tool.

References

[1] B. Alsallakh, A. Hanbury, H. Hauser, S. Miksch, and A. Rauber. Visual methods for analyzing probabilistic classification data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1703–1712, December 2014.

[2] M. Behrisch, B. Bach, N. H. Riche, T. Schreck, and J.-D. Fekete. Matrix reordering methods for table and network visualization. *Computer Graphics Forum*, 35(3):24 – 49, 2016.

[3] S. Carpendale. Evaluating information visualizations. *Information Visualization, Human-Centered Issues and Perspectives*, 4950:19–45, 2008.

[4] E. Mkinen and H. Siirtola. The barycenter heuristic and the reorderable matrix. *Informatica*, 29:357363, August 2005.

[5] D. Ren, S. Amershi, B. Lee, J. Suh, and J. D. Williams. Squares: Supporting interactive performance analysis for multiclass classiers. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):61 – 70, August 2016.

[6] J. Talbot, B. Lee, A. Kapoor, and D. S. Tan. Ensemblematrix: Interactive visualisation to support machine learning with multiple classifiers. *CHI*, 09:1283–1292, April 2009.