Device-Based Content Clustering with DBSCAN And K-Prototype Algorithms in Mobile Applications

* Introduction

The time devices spend on apps and content are crucial factors in creating a more interactive experience and personalized content for the user. In other words, various tables, galleries, etc., in one application throughout the day. These activities are intended to be grouped according to parameters such as the time the user browses the web, likes photos, clicks on the gallery, and watches videos.

The results are proven by using hypothesis testing and summarized in the Tableau. It is revealed that it is possible to combine the wishes and tendencies of the user corresponding to the products in the market.

• First impression over the data

We have 2 tables available. In applications that users enter, an activity called session is started first, and every action made in this session is identified by unique user-log-id numbers. In other words, a session is opened and then when a new activity is clicked in this session, we save this content in user-log-id numbers. In other words, a session is opened and then a user-log-id is opened when some news is clicked in this session, another page is opened, the position is changed, the page is scrolled, even when no action is taken.

• Our first json file is named session.

SessionID: It is the session number that is opened. A device can open more than one session and take actions with various user-log-ids in the session.

SessionID is a unique number that a Web site server assigns a specific user during that user's visit (session). The session ID can be stored as a cookie, form field, or URL (Uniform Resource Locator). Some Web servers generate session IDs by simply incrementing static numbers. However, most servers use algorithms that involve more complex methods, such as taking into account the date and time of the visit and other variables defined by the server administrator.

AppId: GS1 Application Identifiers (AIs) are numeric prefixes used in barcodes and EPC/RFID-tags to define the meaning and format of encoded data elements. There is only one application in our table, so it is unnecessary to add this column to the model.

recordDate: The time created when SessionID is created will not be included in our model.

deviceType:The type of our device

deviceSystemVersion: The current version of the device. Since we have the model and specific id numbers of the devices, it will not have an effect on our model.

deviceUDID: Unique code for each device (mostly Apple). Unique Device Identifier

isSessionFinished: We do not have any information, but the numbers given to the sessions according to various situations. 90 percent of the data has the code number 3.

appVersion: The version of your application. We used it even though we did not expect it to have an effect on the model.

durationInSeconds: The time spent in the session.

• The second json file is the file named user\_log.

UserLogId: Unique code given to various actions taken in Session. For example, clicking the photo.

SessionId: It is the session number that is opened. A device can open more than one session and take actions with various user-log-ids in the session. Our tables will be merged over this column.

StartTime: We will not use the start of the log process in the model, since each log is unique.

StopTime: We will not use the end of the log process in the model because each log is unique.

Duration:Log duration is an important parameter.

ContentAction: Actions between the application and the operating system. For example Video View – the person watched the video, location has changed. One of our most important parameters.

ContentActionCode: The code given to the ContentAction concept mentioned above.

ContentDetail: The content of the action taken followed, for example a video of a famous model or moving at certain latitudes and longitudes.

ContentDetailCode: The code given to the ContentDetail concept mentioned above.

Action: It gives information about the layer of the work, 95% of the data is empty, but there are cases such as UI Feed, Photo View.

ActionDetail: The details about the Action.

ActionCode: Things done in content. For example, opening a photo is content, commenting is ActionDetail and the code for that action is ActionCode.

• The SessionID column is present in both tables, and by this means, inner join operation is used, and they made into a single table as below.

A screenshot of a computer

Description automatically generated with low confidence

• New attributes added to the data

Two numeric and continuous attributes have been added to the data. One of them is the ratio of the specific content action behavior of a device among all the content action behaviors of that device. we named it ratio1.

Table

Description automatically generated

Device numbered 51BBB58A-532B-4E0C-BEA4-7B832FC2D6A9 performed the ContentAction behavior named UIApplicationLocationDidChange in a log for 0.056658 seconds, a total of 13.5925 seconds. Ratio1 is equal to 0.056658/13.5925= 0.00416833.

The second derived parameter ratio2 is the time spent by the device in the log behavior divided by the total time spent. The device numbered 51BBB58A-532B-4E0C-BEA4-7B832FC2D6A9 has spent 0.056658 seconds in the a log, the device has spent a total of 348806 seconds in general. Ratio2 is equal to 0.056658/348806= 1.62434e-07.

• As a result of the EDA analysis, it was observed that there were simulator devices and the behaviors of these devices were deleted from the data.

Text

Description automatically generated

• Rows thought to be caused by network-based errors have been deleted with the DBscan algorithm. Since the variables in object type have long string characters, the working of the algorithm is lightened with Label encoding.

To repeat here, the lines related to high-frequency internet interruptions and location changes caused by the network, rather than the gallery-table-video-photo operations made by the user, have been deleted.

Parameters inserted into the DBSCAN model.

Table

Description automatically generated with medium confidence

Important steps in the DBSCAN algorithm and ContentAction behaviors that aggregate singularly as anomalies.

Chart

Description automatically generated

Chart, line chart

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

Table

Description automatically generated

One of the clusters with possible anomaly behavior extracted from the data is shared below as an example.

During the 12012 line, only the ContentAction named FECLAuthorizationStatusChanged was put in this cluster. Behaviors like this are not the behaviors we want in clustering, but the behaviors that are requested to be deleted in order not to corrupt the data manually have been deleted completely adhering to the DBSCAN algorithm. In other words, the ContentAction named FECLAuthorizationStatusChanged has not been completely deleted. Only its parts within the clusters presented by the DBSCAN algorithm have been deleted! After this process, 224851 lines of data at the beginning decreased to 186096 lines.

A screenshot of a computer

Description automatically generated with low confidence

• After this step, using the K-Prototype algorithm, categorical and numerical variables were run together within the algorithm and clustering were observed.

Text

Description automatically generated with medium confidence

After iteration, which lasted for about 6 hours, the breaking point was observed based on the elbow method. Even though it is a superficial comment, the fact that the break is sharp in one place shows that the algorithm works efficiently.

Chart, line chart

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

By adding 5 different types of clustering given by the algorithm to each line, various ContentAction behaviors of the users interactively with the application were examined.

• Supporting the relationship between clusters and data with hypothesis tests using SPSS.

ANOVA

H0= All means of numerical values between clusters are equal

H1= All means of numeric values between clusters are not equal.

Table

Description automatically generated

There is a significant difference in the mean distributions of the numeric values between the clusters, and so their impact on clustering is significant! In other words, there is a statistically significant relationship between our numerical values and clusters! It has effects on the formation of clusters!

Chi2

H0= Clicked content is independent of clustering, has no effect on clustering, and there is no significant relationship between them.

H1=Cluster values are related to the content clicked, so there is a meaningful relationship between them!

In the test results, it should be examined whether there is a frequency less than 5% in more than 20% of the cells (0 cells (,0%) have expected count less than 5).

If this value is less than 20%, Pearson Chi 2 value can be read, but if this value exceeds 20%, then the continuation of the sentence (The minimum expected count) should be read and

\*If this value is "<5", Fisher's Exact test,

\* If this value is ”5 <=value>=25”, Continuity Correction,

\* If the value is >=25, the Pearson Chi Square value should be read.

Clusters - ContentAction

Table

Description automatically generated

More than 20% (32.9%) of the cells have frequencies less than 5. The expected value is less than 5. In this case, the Fisher test should be interpreted, but the matrix dimensions are not suitable for the calculations and therefore the Monte Carlo test is applied at the 99% confidence interval.

Table

Description automatically generated

H0 hypothesis is rejected.

Clusters – Action

Table

Description automatically generated

The H0 hypothesis can be rejected by looking at the Pearson Chi 2 value.

Clusters -ActionDetail

Table

Description automatically generated

More than 20% of cells have frequencies less than 5. The expected value is less than 5. In this case, the Fisher test should be interpreted, but the matrix dimensions are not suitable for the calculations and therefore the Monte Carlo test is applied at the 99% confidence interval.

Table

Description automatically generated

H0 hypothesis is rejected.

Thanks to the ANOVA test,

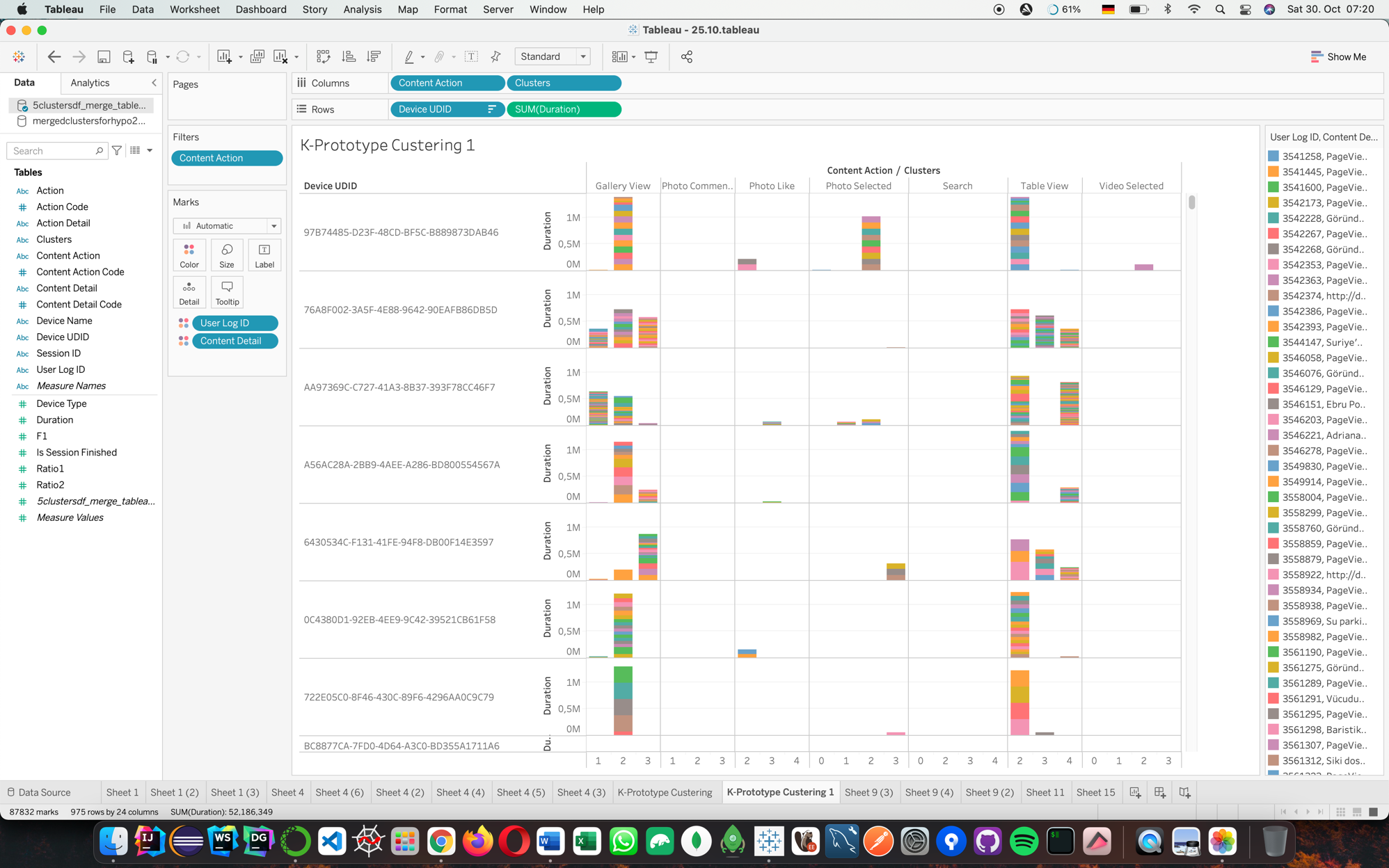
There is a significant difference in the mean distributions of the numeric values between the clusters, and so their impact on clustering is significant! In other words, there is a statistically significant relationship between our numerical values and clusters! It has effects on the formation of clusters!

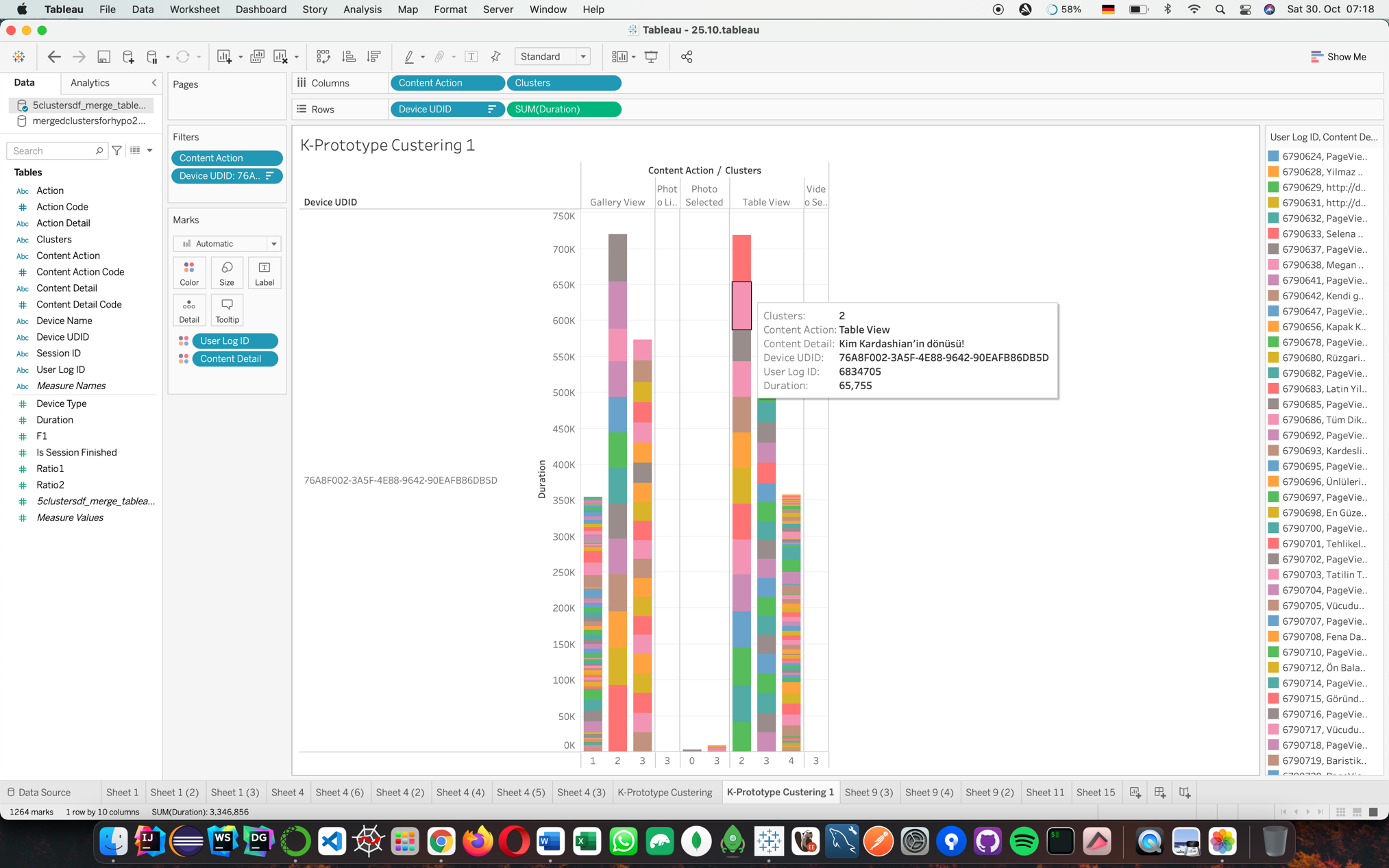
Thanks to the Chi 2 tests,

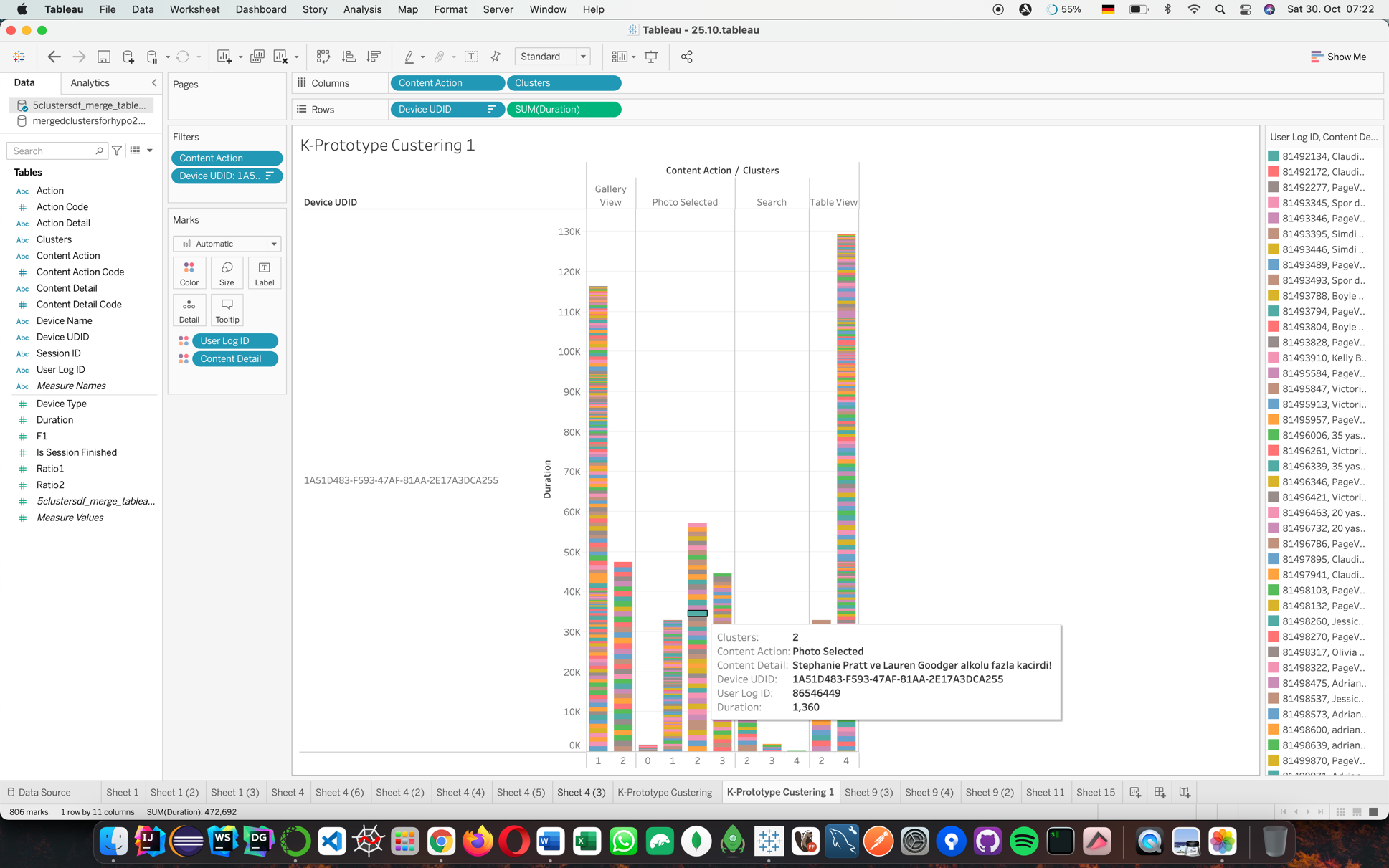
Clustering values depend on the clicked news content, so there is a meaningful relationship between them!

The above comments are feasible, all our variables eligible for testing have a significant effect-relation within the clustering.

• Clustering overview using Tableau



Let's pick random devices and see if their longest-running activities are collected in cluster 2.



At the end of the research, it is seen that ContentActions are clustered on the basis of time on all devices. The contents with the most time spent are added up in the 2nd clusters. If this content has lower durations, it is time-proportionally distributed among the other clusters.

The same ContentAction can be in different clusters on different devices, and that's exactly what we're aiming for. At this point, the contents in the 2nd cluster were examined and a device-specific segmentation was provided. We go much deeper and add the ContentDetail option, a sub-branch that contains the details of the ContentAction behavior, to the chart as color filter.

Today, it is seen in our work that there are thousands of contents that users interact with while determining device-specific advertisements/product promotions. As a result of data being made without time concept and unique segmentations within the device, only with individual parameters such as search or clicked photo, both the wrong product matches the wrong customer and the big one. and useless costs are known to arise. However, as a result of our study, an answer to the question of which content is more popular for each device and how we can reduce advertising costs by narrowing it down and bring the target customer and the target product together has been presented.

A nlp application was made for this cluster and it became a sellable data for the company. For example, valentina vignali, kate upton and kim kardashian are effective names for people using this application.

Text

Description automatically generated

• Summary and visualization of results

The program called Tableau was used to visualize the results. In order to stay true to the name of the research, the interactive “ContentActions” between the application and the user has been filtered. The reason for choosing this filtering is that there is textual information about the content or an intentionally instinct in the log data we have.