ONLINE QoE PREDICTION

Vlado Menkovski, Georgios Exarchakos, Antonio Liotta

Electrical Engineering Department
Eindhoven University of Technology
Eindhoven, the Netherlands

ABSTRACT

The Quality of Experience (QoE) is an irreplaceable metric for evaluating the perceived quality of consumers of multimedia content. Due to the subjectiveness of QoE the most suitable way to measure it is by executing subjective studies. However, executing subjective studies is a complex and expensive process. Careful recreation of the viewing conditions is necessary, and a strict selection of the test subjects is required based on many criteria. This is why solutions are often found in various objective methodologies for measuring the QoE of multimedia. These solutions even though more practical are less accurate and cannot reflect the user expectations. In this work we present a method for building QoE prediction models using machine learning techniques from continuous real-time customer feedback, i.e., during the service execution. This online learning approach builds and adapts prediction models that estimate the QoE based on given Quality of Service metrics from real-time user feedback and does not require apriori execution of subjective studies.

Index Terms— *QoE, Machine Learning, Online Learning, Quality of Experience*

1. INTRODUCTION

Evaluating multimedia quality and user experience is of particular interest to any content and service provider. Understanding the user experience helps in measuring the value that the service brings to customer. Perceived quality can be used as a metric to optimize the encoding parameters and dimension the transport resources. Quality of Experience (QoE) is a subjective metric that measures the perceived quality of the viewers taking into account all the factors that affect their perception, as well as the viewer's expectations. Typically QoE is measured by executing subjective studies where responses of the testers are combined into Mean Opinion Score (MOS) values.

QoE is difficult to predict due to its subjective nature. People present differences in their preferences and variations in the expectations. There are many factors that affect QoE, some are from technical character, but there are also environmental conditions that influence the perception. QoE is commonly referred to as a scalar value, mainly for the simplicity reasons. However, some argue that it can be understood as a multidimensional value consisted of different aspects of quality [1]. There are many efforts that try to determine the aspects that contribute to the perceived quality and try to develop objective measurements for those aspects. So far, a complete and accurate objective methodology has not been developed and the de facto methods for measurement of QoE are the subjective studies.

Subjective studies albeit efficient in QoE estimation, represent a significant undertaking. They require reference material for comparison, tightly controlled viewing conditions and comprehensive selection of participants. These procedures represent cumbersome process, which sometimes is a technical challenge to execute, particularly in cases such as real-time streaming content.

Mainly because of these reasons most of the work in the area of QoE and multimedia quality has been focused on developing different objective methodologies for estimation of the quality values. The objective approaches investigate different aspects that affect the QoE, from the content itself to the transport conditions. Some of them take into the account the characteristics of the human visual system (HVS), some only focus on the fidelity of the signal. However, rarely do they take into account all of the contributing factors, such as type of terminal, type of content and finally the viewer's expectations.

In this work we present an approach that uses Machine Learning (ML) techniques to develop QoE prediction models which do not rely on training data from subjective studies, but are built in an online manner from real-time user feedback. This approach effectively circumvents the need for full-scale subjective studies. There is no need for specific participant selection, because the users of the service themselves are considered. The conditions of the service do not need to be replicated because the feedback is taken from the actual running service. As service conditions vary from one stream to the other our QoE prediction model learns more and becomes more complete and it's the accuracy improves. In addition this methodology provides for models that adapt to changes in the user preferences as well as to the introduction of new conditions in the environment such as new content and new terminal devices.

2. RELATED WORK

Measuring QoE is getting more interest as more services are introduced where the accurate and on-time delivery of data affects the perceived quality. Most of the work in this

area explores different objective ways to map the QoE with characteristics of the encoding and/or transport. The ITU standardization process has defined five groups of models for estimation of QoE [2]. They are as follows: media layer, packet layer, parametric planning, bit stream and hybrid models. The media layer models focus on the media content, the losses during encoding and compression, and the fidelity of the multimedia signal. They represent objective metrics where the original signal is compared to the compressed one and the amount of differences or error is evaluated as loss of quality. There are versions that use full reference to compare with the original signal, but also there are models that have restricted or no reference. Common comparison methods include Mean Squared Error (MSE) and Peak Signal to Noise Ratio (PSNR), which lack the understanding of how the content is perceived by the viewer or the Human Visual System (HVS). These methods usually deliver lower accuracy results [3]. The packet layer models focus on the transport quality loss, and look at data from packet headers. On the other hand, the parametric planning models consider network allocation of resources. They both have smaller computational footprint and are less intrusive than the media layer models, but also only look at a single perspective of the conditions that affect QoE. The bitstream models look at the transport errors and other network impairments and try to map them the loss of quality. These are more intrusive than the previous, but share the same limitations. There are methods the combine the Network Quality of Service (NQoS) with the Application Quality of Service (AQoS) [4] towards a more holistic approach that improves the accuracy by looking at more than one condition of QoE. Nevertheless, these models still lack the understanding of HVS, and even more so, none of the models takes into account the expectations of the viewers.

Most discussions in the area of QoE conclude that Mean Opinion Score (MOS) derived from subjective studies is the most relevant metric, and as such it is usually used as a benchmark. There are methods motivated from this fact that try to directly map the MOS to the known AQoS and NQoS conditions such as [5] and [6]. Some have gone further and use different statistical methods to correlate these parameters with the QoE. However, these methods need subjective data from apriori subjective studies to build the models. Furthermore, the models cannot adapt to changes neither easily nor quickly. We previously showed that ML can greatly improve the accuracy of the QoS-to-QoE correlation in [7] and [8]. Hereby we extend that method with a novel online learning capability, a significantly more versatile, accurate and dynamic approach to predict QoE.

3. ML BACKGROUND AND USED ALGORITHMS

In this section we are presenting a short background of the ML techniques and the algorithms used in our online QoE prediction method. This method is based on building prediction models from real-time user feedback and available QoS data from streaming parameters, such as network probes. Those prediction models are further used to estimate the QoE based on available data during service execution. The training of the prediction models is implemented by using Online Learning techniques. These techniques, for induction of prediction models, use an inflow of data sequentially in the training algorithm.

Online Learning is a part of a more general group of techniques called supervised learning. A more classic supervised-learning training involves using all available data in a batch procedure to train the models. Formally this procedure can be described as follows: the training set is a set of example pairs in the form of $(\vec{X}, y)$, where $\vec{X}$ is an input vector of attributes and y is a class or a label. In our case for example $\vec{X}$ could be a set of the QoS attributes such as video bitrate, frame rate, and audio bitrate. The label can be the value of the QoE (good, fair, bad…). The goal of the procedure is to derive a function *f* such as $y = f(\vec{X})$. This function represents our prediction model and can be in many forms such as decision tree, artificial neural net, and a support vector machine.

In this work we have tested many different online learning algorithms, and for the needs of this application we found the most suitable to be Hoeffding Option Trees. In the following subsection we will walk you through the details for Hoeffding Option Trees and the algorithms for their induction.

3.1 Decision Trees

Decision Trees (DT) are models represented by a hierarchical tree structure where each node represents a test (or a question) and each leaf is associated with one possible decision (class or a label) (Fig.1). In ML there are algorithms that use DT as prediction models for classification. Those algorithms are also referred to as DT induction algorithms.

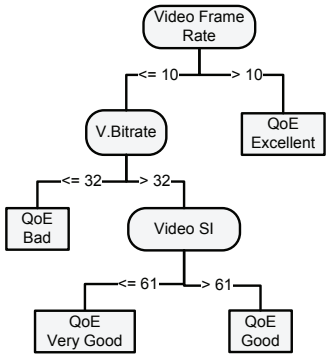

Figure 1. Decision Tree

One very significant algorithm and basis for many future developments is the ID3 [9] algorithm. ID3 is an iterative algorithm that assigns a test on a selected attribute ($x \in \vec{X}$) for each node in the DT. The selection of the attribute tested is based on the maximum information gain metric. For each possible value that the attribute has in the dataset the algorithm assigns a different branch. The procedure continues down the branches using a subset of the training

data passing the test in the parent node. The algorithm finishes a particular path when it is not reasonable anymore to split the training subsets. Then the leaf node is added and a class is associated to this node which is the majority class of the remaining data.

Using the same principles the C4.5 [10] algorithm is developed. This algorithm overcomes many weaknesses of ID3, such as handling continuous attributes, training data with missing values, and many different pruning enhancements that deal with overfitting.

3.2 Hoeffding Trees and incremental tree induction

Hoeffding Trees [11] are a DT model that is designed to handle extremely large training sets. The expected training set is therefore not intended to remain in memory, but to be processed from a stream in a single pass. The fact that the data is processed sequentially or one datapoint at the time characterizes this approach as Online Learning. The learner in this case has only a partial view of the data. This means that the selected attribute for the test in a node cannot be made with full confidence for any split criteria, but it has to be made with a more relaxed one. The Hoeffding Tree deals with the issue of the number of examples needed to make a split decision by relying on a statistical result known as Hoeffding bound. We make n observations of a random variable r with a range R and determine the computed mean of r to be $\bar{r}$. The Hoeffding bound states with probability $1-\delta$ that the true mean of the variable is $\bar{r}-\varepsilon$ whereby

$$\varepsilon = \sqrt{\frac{R^2 ln\left(\frac{1}{\delta}\right)}{2n}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

If we define the attribute selection criterion as $G(\vec{X})$, then $\Delta G = G(x_1) - G(x_2) > 0$ assuming that the x_1 attribute is more favorable (with larger information gain) than x_2. Now, given the desired δ, the Hoeffding guarantees that x_1 is the better selection with probability δ if n examples are seen where $\Delta G > \varepsilon^2$.

3.3 Hoeffding Option Trees

Option trees generalize the regular decision trees by adding a new type of node, an option node [12]. Option nodes allow several tests instead of a single test per node. This effectively means that multiple paths are followed below the option node and classification is commonly done by a majority-voting scheme of the different paths. Option Decision Trees can reduce the error rate of Decision Trees by combining multiple models and predictions while still maintaining a single compact classifier. In the implementation we used the combination of the predictions of different paths is done with weighted voting [13], where the individual probability predictions of each class are summed.

3.5 Oza Bagging

Ensemble methods use multiple classifiers to obtain better performance than the stand-alone classifiers themselves. We use the Oza Bagging algorithm which is an Online Learning ensemble method. Ensemble methods train the multiple classifiers with different strategies. Then their predictions are combined to form a more accurate group prediction. Their strength is in improving the generalization capabilities of the stand-alone classifier [14]. Bagging is a procedure of bootstrap aggregation where one base classifier of the ensemble is trained on the whole dataset D and the rest of the classifiers are trained on a sub sample of D, sampled uniformly with replacement. Online bagging [15] modifies this method for streaming data in the following manner: each example of data $(\vec{X}, y)$ is presented to a base classifier K times, where K is a random variable with Poisson(1) distribution. The authors of [15] claim that the online bagging classifier's accuracy converges to the batch bagging classifier's given certain conditions and a training set where the number examples tend to grow to infinity.

4. ONLINE QOE PREDICTION APPROACH

The online QoE prediction approach is developed to estimate the QoE MOS using prediction models trained from user feedback. The user feedback is collected continuously and the models are built in an online fashion using Online Learning ML techniques.

This method is particularly useful for live streaming services for which apriori subjective tests are difficult to implement. For the implementation of this method we require the existence of a mechanism to gather the user feedback at service run time. Our method, as depicted in Figure 2, relies on whatever system data is available, usually QoS data from the streaming setup as well as QoS data from network probes. This data is mapped with user feedback to make training datapoints. Each time a user feedback is received, it is mapped with QoS data ($\vec{X}$ being QoS and y being QoE) and fed back to the learner engine. The learner is an Online Learning algorithm that builds a prediction model from this data. Meanwhile the built prediction model is used to predict the QoE from QoS available data.

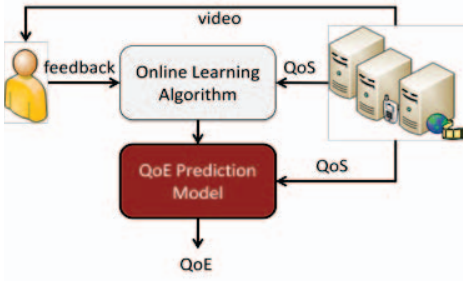

Figure 2. Online QoE Prediction approach

In other words we have a model that estimates the QoE based on available system data, and improves over time as more and more feedback is available. Furthermore, if changes in the environment happen such as introduction of different terminal types or different content, the model will adapt as soon as there is feedback for the new conditions.

This method provides for an accurate and flexible way to measure QoE on systems where viewer feedback is

available, without engaging in comprehensive subjective testing.

5. EXPERIMENTAL SETUP

In order to validate and demonstrate this method we implemented the following experimental setup. We used data from a subjective study for QoE of mobile multimedia streaming content [16]. The study is based on the Method of limits [17], where each viewer is reporting the threshold where the QoE changes from acceptable to unacceptable.

Table 1. QoS conditions for the sequences in the subjective studies

	Segment	*Time (seconds)*	*Video bitrate (kbit/s)*	*Audio bitrate (kbit/s)*	*Frame-rate*
Mobile	1	1-20	384	12.2	25
	2	21-40	303	12.2	25
	3	41-60	243	12.2	20
	4	61-80	194	12.2	15
	5	81-100	128	12.2	12.5
	6	101-120	96	12.2	10
	7	121-140	64	12.2	6
	8	141-160	32	12.2	6
PDA	1	1-20	448	32	25
	2	21-40	349	32	25
	3	41-60	285	32	20
	4	61-80	224	32	15
	5	81-100	128	32	10
	6	101-120	96	32	10
	7	121-140	64	32	6
	8	141-160	32	32	6
Laptop	1	1-20	448	32	25
	2	21-40	349	32	25
	3	41-60	285	32	20
	4	61-80	224	32	15
	5	81-100	128	32	10
	6	101-120	96	32	10
	7	121-140	64	32	6
	8	141-160	32	32	6

The subjective study is done on three devices: a mobile phone, pda and a laptop. The devices have different screen sizes and they are held in a different manner, both factors that affect how the quality is perceived. In addition to this users have different expectations of the performance on each of the devices. Table 1 presents the different QoS parameters of the streaming content. The content has been split in 20 seconds-long segments with decreasing quality. From the study results we have created a dataset which is presented in a subset in Table 2.

Table 2. Sample set of the subjective study results

Terminal	*Video SI*	*Video TI*	*Video Bitrate (kb/s)*	*Video Frame-rate*	*QoE Accept*
mobile	70	141	64	6	no
mobile	60	153	128	12.5	yes
pda	71	125	128	10	yes
laptop	67	70	32	10	no
pda	62	100	285	20	yes
laptop	67	70	32	10	no
laptop	23	130	363	25	yes
mobile	21	187	32	6	no

The streams in Table 2 are defined by the terminal type they were watched on, the video Spatial Information (SI), video Temporal Information (TI), video bitrate and video frame-rate. The video SI and video TI have similar values for videos of the same content type. For example head-and-shoulder video of news broadcast typically has low movement in space and in time, hence the two parameters have low values. The video TI and SI are calculated as explained in [18].

Datapoints are then generated for each measurement of these parameters and the answer for the QoE. The order of datapoints is then randomized and finally they are presented to the online learning system one by one.

For an Online Learning System we used the Massive Online Analysis (MOA) [19] ML platform for data stream mining, which has implementations of Hoeffding Option Tees and Oza Bagging algorithm. MOA is based on the WEKA [20] ML data mining platform and it is optimized for fast stream mining which implies a lot of online data passing through the classifier. In our case the viewer feedback is considered scarce and expensive, so we can only expect small amount of it. In light of this difference we have modified some of the parameters of the algorithm to serve our purpose, mainly the n_{min} grace period from 200 (default value) to 1. It is not meaningful to wait for 200 datapoints until we start building the decision tree when we only have 3500 datapoints available.

More work was needed for the validation procedure of the models. In MOA there is the assumption of abundance of data, and the estimation of the accuracy of the prediction models and their accuracy is done by interleaving testing and training. In this way there is part of the data that is dedicated for testing, and this data is not used for training of the models. Consequently, the accuracy of these models could be lower in cases of small amount of data. Standard approach for validation in situations with scarce data is cross validation [21].

We implemented a ten-fold cross validation scheme to calculate the accuracy of the classifier. This validation scheme splits the data into 90% for training and 10% for testing, and then it repeats this process ten times. Each time different combination of datapoints is used for training and testing. We also implemented a validation scheme for testing concept drift where the classifier is trained on one dataset and then at a certain point a new dataset is introduced. The second dataset can contain different distribution of the data. At the moment when the new data is introduced the model is not aware of the change and predicts based on knowledge from the previous data. Then new feedback is introduced from the second dataset. The algorithm updates the model according to the introduced data. In our algorithm stack, different algorithms behave differently. The Hoeffding Option Tree discards some nodes and induces new ones. The weights on the different paths of the option tree might be modified, and the online ensemble classifier can decide to update the weights to the individual classifiers if their accuracy decreases. With the proposed experimental setup we can monitor the introduction of new

concepts in the dataset and the speed of adaptation to the model.

6. EXPERIMENTAL RESULTS AND ANALYSIS

The results of the execution of the Online Learning using the Hoeffding Option Tree algorithm are shown in Figure 3.

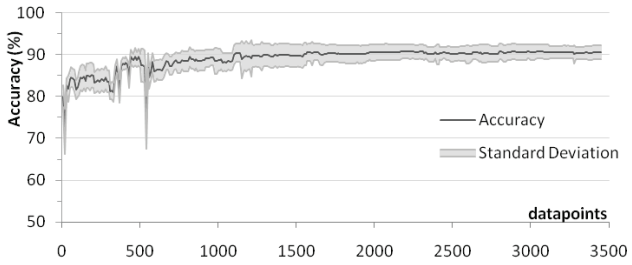

Figure 3. Hoeffding Option Tree results

The classification accuracy rises fast to over 80% accuracy with fewer than 100 datapoints, i.e. user-generated feedback instances. After around 1000 datapoints the classifiers converges to its accuracy of approximately 90%. In the same manner the standard deviation of the accuracy falls quickly to bellow 3 (with just a few exceptions) and then falls to around 2 after introducing 1600 points.

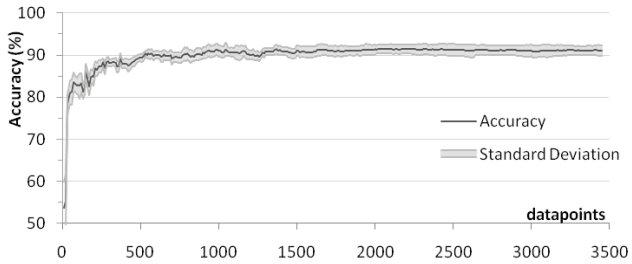

Figure 4. OzaBagging Hoeffding Option Tree results

We obtained qualitatively similar findings from the execution of the ensemble Online Learning algorithm Oza bagging Hoeffding Option Tree (Figure 4). We can see that both algorithms reach very high prediction accuracy (90%) very rapidly (order of a thousand of datapoints). As expected the ensemble approach gains accuracy faster. Furthermore the classifier's standard deviation of accuracy over the different folds of the cross-validation is much lower than in the stand-alone classifier.

Overall, we have presented results that show that we can already achieve an accuracy of over 80% by learning only 100 datapoints which are randomly selected feedback.

In Figure 5 we present the results from testing with concept drift. We split the dataset in two smaller ones. The first dataset contains only video with Temporal Information smaller than 110 which is around 60% (2010 out of 3370) of the data. The second set contains the remaining 40% of data.

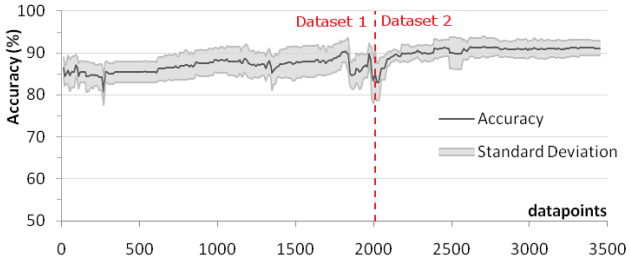

Figure 5. Hoeffding Option Tree with concept drift results

The result from using the Hoeffding Option Tree algorithm shows drop in the accuracy and increases in the standard deviation at the moment the new dataset is introduced. The accuracy is recovered very fast and converges above 90% in fewer than 200 datapoints. This is a very encouraging result that shows the capabilities of this algorithm to adapt to changes. In this experiment the model was trained on content with small TI (slow changing content) and then we introduced high changing content. Even with rather drastic change like this the accuracy recovered very fast.

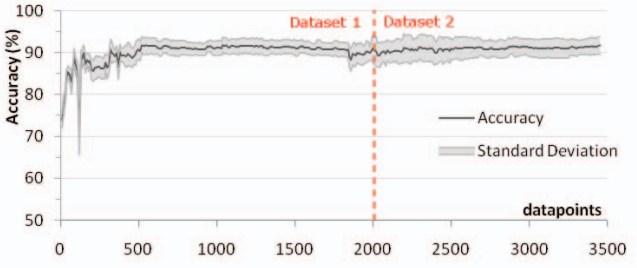

Figure 6. OzaBagging Hoeffding Option Tree with concept drift results

On the other hand, the results in Figure 6 from the Oza Bagging Hoeffding Option Tree ensemble show that this algorithm is much more robust to changes and deals with the concept drift with close-to-none loss in accuracy and limited rise in the standard deviation. This result is even more impressive and shows the robustness of the ensemble approach and justifies the added complexity in using an ensemble versus a standalone classifier.

To demonstrate the statistical significance and viability of our approach, Figure 7 illustrates how our two algorithms Hoeffding Option Tree (HOT) and Oza Bagging HOT (OzzaBagg HOT) compare with 3 standard ML approaches, namely Naïve Bayes, Support Vector Machine (SVM) and C4.5.

It is evident, that C4.5 performs best with 93% accuracy but the online learning algorithms we used follow closely behind with 90.5% and 91.1%.

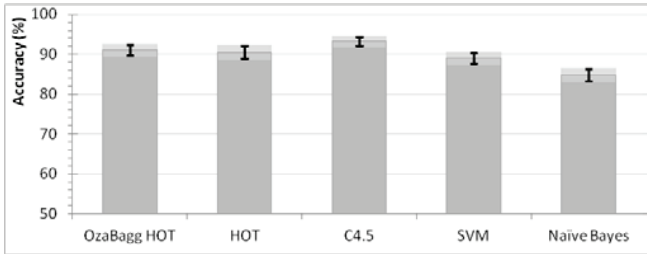

Figure 7. Comparison with standard ML algorithms

7. CONCLUSION

In this paper we present a methodology for estimation of QoE that relies on Online Learning, an ML technique for induction of prediction models sequentially one datapoint at a time. This methodology circumvents the need for complex and expensive apriori subjective studies building prediction models from user feedback. We demonstrate the usefulness of this approach by testing it on data that was previously derived via conventional subjective studies. Our QoE prediction models show high accuracy and high adaptability to concept drift in the dataset. The fact that the accuracy of the online learning algorithms are approaching the accuracy of the standard batch ML algorithms (of above 90%) demonstrates the applicability of the approach.

Our next priority is to work out effective means to automate the user's feedback collection process and incorporate it with our online learning framework. Further this work can benefit from an Active Learning approach that will selectively query for user feedback based on expected information gain from the acquired data.

8. REFERENCES

[1] S. Winkler, *Video Quality and Beyond*, Symmetricom, 2007.

[2] A. Takahashi, D. Hands, and V. Barriac, "Standardization activities in the ITU for a QoE assessment of IPTV," *Communications Magazine, IEEE*, vol. 46, 2008, pp. 78-84.

[3] S. Winkler, *Digital video quality : vision models and metrics*, Chichester West Sussex ;;Hoboken NJ: J. Wiley & Sons, 2005.

[4] M. Siller and J. Woods, "QoS arbitration for improving the QoE in multimedia transmission," *Visual Information Engineering, 2003. VIE 2003. International Conference on*, 2003, pp. 238-241.

[5] F. Agboma and A. Liotta, "QoE-aware QoS management," *Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia*, Linz, Austria: ACM, 2008, pp. 111-116.

[6] F. Agboma, "Quality of Experience Management in Mobile Content Delivery Systems," Department of Computing and Electronic Systems, University of Essex, 2009.

[7] V. Menkovski, A. Oredope, A. Liotta, and A. Cuadra Sánchez, "Optimized online learning for QoE prediction," 2009.

[8] V. Menkovski, A. Oredope, A. Liotta, and A. Cuadra Sánchez, "Predicting Quality of Experience in Multimedia Streaming," *Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia*, Kuala Lumpur, Malaysia: 2009, pp. 52-59.

[9] J. Quinlan, "Induction of Decision Trees," *Machine Learning*, vol. 1, Mar. 1986, pp. 81-106.

[10] J.R. Quinlan, *C4. 5: programs for machine learning*, Morgan Kaufmann, 2003.

[11] P. Domingos and G. Hulten, "Mining high-speed data streams," *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2000, pp. 71–80.

[12] R. Kohavi and C. Kunz, "Option Decision Trees with Majority Votes," *Proceedings of the Fourteenth International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., 1997, pp. 161-169.

[13] B. Pfahringer, G. Holmes, and R. Kirkby, "New Options for Hoeffding Trees," *AI 2007: Advances in Artificial Intelligence*, 2007, pp. 90-99.

[14] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, 1996, pp. 123–140.

[15] N.C. Oza and S. Russell, "Online bagging and boosting," *Artificial Intelligence and Statistics*, 2001, pp. 105–112.

[16] F. Agboma and A. Liotta, "Addressing user expectations in mobile content delivery," *Mobile Information Systems*, vol. 3, Jan. 2007, pp. 153-164.

[17] G.T. Fechner, E.G. Boring, H.E. Adler, and D.H. Howes, *Elements of psychophysics / Translated by Helmut E. Adler ; Edited by David H. Howes [and] Edwin G. Boring ; with an introd. by Edwin G. Boring*, New York :: Holt, Rinehart and Winston, 1966.

[18] R.I. ITU-T, "910," *Subjective video quality assessment methods for multimedia applications*, 1999.

[19] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavaldà, "New ensemble methods for evolving data streams," *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 139–148.

[20] I.H. Witten and E. Frank, *Data mining*, Morgan Kaufmann, 2005.

[21] R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," *IJCAI*, 1995, pp. 1137–1145.