

## A Predictive Model for Video Lectures Classification

Priscylla Silva<sup>1,2</sup>, Evandro Costa<sup>1</sup>, Roberth Pinheiro<sup>1</sup>

<sup>1</sup>Instituto de Computação – Universidade Federal de Alagoas (UFAL)  
Campus A. C. Simões – Av. Lorival Melo Mota, s/n, Maceió – AL – Brazil

<sup>2</sup>Instituto Federal de Alagoas (IFAL) – Murici, AL – Brazil

{pms, evandro, rrraraujop}@ic.ufal.br

**Abstract.** *In the educational context, it is important to provide students with learning resources, such as tutorials, video lectures, and educational games to help their learning process, especially when they are at home and have difficulties or doubts. In these cases, recommendation systems have been used to suggest learning resources for students, avoiding the difficult task of making the manual process of searching and selecting resources. Most generic recommendation systems for video lectures use viewing history of the user to make recommendations of videos, which that are consistent with the interests of users. In the educational context, other factors must be considered, the video should not only be of interest to the student, as it will be used as a learning resource for the main purpose of helping the student to learn a particular subject or clarify doubts. Hence, in this paper we evaluated three classifiers and propose a predictive model to classify video lectures according to their quality. We applied machine learning algorithms on a set of video lectures by students classified according to some quality requirements. We conducted an experiment and preliminary results indicate good quality of the selected prediction model.*

### 1. Introduction

In the educational context, it is important that students are aware of available learning resources (such as tutorials, video lectures, and educational games) that will help their learning process, especially when they are studying at home by themselves and some difficulty or doubts may arise. In this case, when there is no teacher or specialist to assist students, they often search for learning resources on the Web. The search for relevant information on the Web is a well known problem and addressed by several studies in the literature [Baeza-Yates and Ribeiro-Neto 2011]. This problem consists in immense amount of information contained on the Internet and how hard it is to find some really quality content and information that meet the user needs. To solve this problem, recommendation systems were created. In the educational context, recommendation systems have been used to recommend learning resources for students, helping the students to save time in the process of searching and selecting resources, compared to manual process. One of the most used resources by students are the video lectures. Websites such as Youtube<sup>1</sup> and Vimeo<sup>2</sup> have many videos within various themes, including video lectures of various topics. These websites are used by students as their main resource for video lectures.

---

<sup>1</sup><http://www.youtube.com/>

<sup>2</sup><http://vimeo.com/>

The amount of videos on the Internet is growing at an explosive rate [Boll 2007], making it harder the search for good video lectures. When students have to learn a subject or they are in doubt, they generally perform the following steps: 1. Search on a website of videos using keywords from the subject; 2. Choose one of the first videos ranked to watch; 3. If the video is not good enough for them, then they stop watching it and try another video. Thus, one possible and non interesting scenario is one where a given student selects several bad videos followed, until he find a video that he classifies as good and watch the full video to learn what he needs. This happens because students, not always, have some relevant criteria to predict the quality of video lecture selected. A common attribute used to select the video by students is the amount of views (view count), but only this attribute is not enough to classify video lectures, because this attribute can be biased by the age of video and tags and description well elaborated.

Many generic recommendation systems for videos use viewing history from previous users to suggest videos that will interest users. In the educational context, other factors must be considered, eg whether the video lecture shows properly the subject that the student needs to learn. Recommendation systems can be used in educational systems to help students in the learning process [Ferro et al. 2011] and in, particularly, problem solving situations [Silva et al. 2013].

In this paper we evaluated three classifiers to classify video lectures according to their quality. The classifiers Naive Bayes [Bielza and Larrañaga 2014], SVM [Chang and Lin 2011] and C4.5 [Quinlan 1993] were used. The classifier with better performance compared to others was selected. We conducted an experiment and preliminary results indicate good quality of the selected prediction model. We make an analysis of classifier algorithm to determine the quality of video lectures extracted from Youtube.

This paper is organized as follows: Section 2 discusses the related work. Section 3 provides a brief discussion about the studied classifiers. Section 4 presents the evaluation method and experiments for the evaluation of classification algorithms. Section 5 presents the method, results and discussion of the experiment. Finally, the conclusion and future work are given in Section 6.

## **2. Related Work**

Researches on video recommendation focus primarily on three typical approaches: collaborative filtering (CF), content-based filtering (CBF), and hybrid filtering (HF), which combines the two previous approaches [Adomavicius and Tuzhilin 2005]. According to Zhao et al. in [Zhao et al. 2012], collaborative filtering is the most widely adopted approach for video recommendation. In this sense, most of the works in literature use viewing history user for the recommendation.

The work by Zhao et. al. [Zhao et al. 2012] suggests an approach by users viewing history for personalized video recommendation. Recommendations to a given user are based on the interest degree of this video by the user's friends, and the taste similarities between the user and his friends. How much interest a user has in a video is given by the degree of similarity between the video and the user profile (made up of set tags created through your viewing history). The set of tags is also used to calculate similarities between the user and their friends.

The work proposed by Yang et. al. in [Yang et al. 2009] presents a recommendation system of educational videos through the summarization of video content. This work aims to solve the problem that titles and descriptions of multimedia content are not always explicit and along with the video, several subjects are addressed, therefore the user needs to watch the full video to find out if that video contains the desired content. Therefore, this work is concerned with video recommendation with adequate content for students and teachers, for this, it performs a process of summarization of video subtitles, combined with the information contained in the title and description, and calculates the similarity between videos and user's profile, which consists of items and subjects of interest to users.

Zhao and Wand (2013) proposed a personalized recommendation algorithm for educational videos. This algorithm carries out the recommendations based on context and the trust relationships between users.

The aforementioned works are effective in their aims to recommend videos of interest to users, but in the educational context, other factors beyond the degree of user's interest must be taken into considerations, for example, pedagogical aspects of the video, such as its quality, not only their relevance.

One of the problems of using viewing history is that websites such as *YouTube*, mark the video as seen even if the user has not seen the full video, only watching a few seconds of it. Therefore, even videos that users did not like will be shown on their history. In the educational context, the main goal is to have an evaluation of the video, but unfortunately users don't always use options "like" offered by websites.

The differential of our work is to provide a classification of the quality of the videos. This classification can be used to improve existing recommendation systems.

### 3. Studied Classifiers

#### 3.1. Naive Bayes Classifier

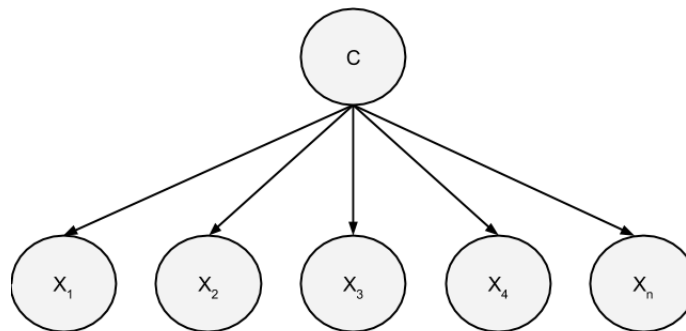
Naive Bayes classifier is one of the best known classifier that uses a Bayesian approach. It has shown to be very efficient on a variety of data classification problems [Taheri et al. 2011]. In the Naive Bayes, features are conditionally independent given the class, it means that all features have only the class as parent (Figure 1). Although the independence assumption is problematic in some cases, the accuracy of the Naive Bayes classifier is typically high, especially where the features are not strongly correlated [Domingos and Pazzani 1997].

The Naive Bayes classifies an instance  $X = (X_1, X_2, \dots, X_n)$  using Bayes rule, by selecting:

$$\operatorname{argmax}_{c \in C} P(c) \prod_{i=1}^n P(x_i|c) \quad (1)$$

#### 3.2. SVM Classifier

Support Vector Machine (SVM) is a popular supervised learning method for classification and regression [Wang 2008]. A SVM involves two stages: training and test. For basic classification into two types, the SVM finds a hyperplane which separates two-class data



**Figure 1. A Naive Bayes structure**

with the maximal margin. A SVM represents instances as points in space, where the instances of each category are separated by a clear space that is as wide as possible. New instance is mapped in that same space and predicted to belong to a category based on which side of the gap they fall on.

### **3.3. C4.5 Classifier**

C4.5 is a classification algorithm by Quinlan [Quinlan 1993]. It is a tree-induction algorithm and is one of the fastest algorithms for machine learning and data mining [Ruggieri 2002]. The C4.5 algorithm uses a training set and constructs a decision tree. The training set is a set of cases or instances. Each case specifies values for a collection of attributes and for a class. Each attribute may have either discrete or continuous values. A decision tree is a tree data structure consisting of decision nodes and leaves. A decision tree is used to classify a case.

## **4. Experiment: Comparative Analysis between the classifiers**

The purpose of the experiment was to evaluate the predictive ability from classifiers towards video evaluation. To perform the experiment the Weka software was used [Bouckaert et al. 2010].

### **4.1. Dataset**

To perform the experiment 120 video lectures from YouTube were collected in the math domain. The video lectures belong to the following topics: logarithms, Cartesian plane, set theory, polynomial functions, geometric progression and matrices. A total of 15 undergraduates volunteered were selected to evaluate the videos. Each video lecture was assessed by 5 volunteers. All the volunteers assessed the same number of videos. Volunteers have the following characteristics in common: they were between 18 and 25 and students of courses in the exact sciences.

Volunteers evaluated the video lectures by applying grades from 0 to 10, the median from these evaluations was used to generate the overall assessment of video lecture. Each volunteer was instructed to review the video lectures on the following criteria: clarity, teaching method, depth in the proposed issue, audio quality and image, teacher's didactic, among others.

**Table 1. Attributes of the Video Lectures after the preprocessing.**

| Attribute                         | Labels                                                                           |
|-----------------------------------|----------------------------------------------------------------------------------|
| Title Size                        | 1 - short, 2 - average, 3 - long                                                 |
| Description Size                  | 1- short, 2 - average, 3 - long                                                  |
| Time                              | 1 - new, 2- average, 3 - old                                                     |
| Duration                          | 1 -short, 2 - average, 3 - long                                                  |
| View                              | 1 - very little, 2 - little,<br>3 - average, 4 - big, 5 - very big               |
| Like                              | 1 - no likes, 2 - few likes,<br>3 - average likes, 4 - many likes                |
| Dislike                           | 1 - no dislikes, 2 - few dislikes,<br>3 - average dislikes,<br>4 - many dislikes |
| Comment                           | 1 - no comments, 2 - few comments<br>3 - average comments,<br>4 - many comments  |
| Evaluation<br><i>Manual Label</i> | 1 - inadequate, 2 - bad,<br>3 - average, 4 - good,<br>5 - excellent              |

A video lecture is composed of the following attributes: *id*, title, description, duration, date of publication, view count, like count, dislike count, favorite count and comment count.

#### 4.2. Preprocessing

The preprocessing aims at ensuring the quality of data being used in the experiment. The preprocessing was carried out through two steps: cleaning and data transformation. The cleaning step is the removal of missing data, in this stage the favorite count attribute has been removed because it was considered an uninformative attribute given that he contained *nil* values in most of the videos and in the other videos had values below 10, it is believed that this occurred because the fact that few users mark video lectures as favorites, this is a more common feature with entertainment videos. The *id* attribute was also removed because it is not relevant to the process.

The data transformation step consists in discretization and normalization. The attributes view count, like count, comment count and dislike count are continuous attributes extracted from *YouTube*. These attributes were categorized using histogram analysis and transformed into ordinal attributes, as shown in Table 1. In order to use the selected classifiers, such as SVM, that use only numeric data, for each category was assigned a value, preserving the order relations (see Table 1). Each video has a class called “evaluation” that can take the labels: inadequate, bad, average, good, and excellent. This label is the result of the manual assessments done by volunteers.

### 4.3. Validation

The experiment was performed using 10-fold stratified cross-validation. This procedure divides the sample into  $k$  mutually exclusive parts (folds), for each step  $k - 1$  folds are used for training and the induced hypothesis is tested on the remaining fold. In order to get statistically meaningful results, the number of iterations used was 10. In case of 10-fold cross-validation this means 100 calls of one classifier with training data and tested against test data. The current experiment performs 10 runs of 10-fold stratified cross-validation on the dataset using Naive Bayes, SVM and C4.5 scheme, this means 300 calls.

The goal of the experiment is to check which classifier has the best performance in the labeling of *evaluation* attribute.

### 4.4. Evaluation Metrics

Given an algorithm  $A$  and a set of instances denominated  $T$ , assume  $T$  is divided into  $k$  partitions. In the case of 10-fold cross-validation,  $k = 10$ . For each partition  $i$ , the hypothesis  $h_i$  is induced and the error denoted by  $err(h_i)$ , where  $i = \{1, 2, \dots, k\}$  is calculated. The mean, variance and standard deviation for all partitions are calculated using the following formulas:

$$mean(A) = mean(A, T) = \frac{1}{k} \sum_{i=1}^k err(h_i) \quad (2)$$

$$var(A) = var(A, T) = \frac{1}{k} \left[ \frac{1}{k-1} \sum_{i=1}^k (err(h_i) - mean(A, T))^2 \right] \quad (3)$$

$$sd(A) = sd(A, T) = \sqrt{var(A, T)} \quad (4)$$

When comparing two inductors in the same domain  $T$ , the standard deviation can be seen as a picture of the robustness of the algorithm: if the errors (calculated on different test sets) derived from induced hypotheses using different training sets are very different from one experiment to another, this indicates that the inductor is not robust to changes in the training set, coming from the same distribution. To compare two machine learning algorithms and decide which one is better (with confidence level of 80%), just take the general case to determine whether the difference between two algorithms ( $A_i$  and  $A_j$ ) is significant or not, assuming a normal distribution. For this, the mean and standard deviation combinations are calculated according to the following equations:

$$mean(A_i - A_j) = mean(A_i) - mean(A_j) \quad (5)$$

$$sd(A_i - A_j) = \sqrt{\frac{sd(A_i)^2 + sd(A_j)^2}{2}} \quad (6)$$

$$z(A_i - A_j) = \frac{mean(A_i - A_j)}{sd(A_i - A_j)} \quad (7)$$

**Table 2. Results of the comparative analysis of classifiers used.**

|       |             |          |             |
|-------|-------------|----------|-------------|
| $A_i$ | Naive Bayes | SVM      | C4.5        |
| $A_j$ | SVM         | C4.5     | Naive Bayes |
| mean  | 0,0785      | -20,0939 | 20,0939     |
| sd    | 0,0555      | 56,9593  | 56,9593     |
| z     | 1,4142      | -0,3528  | 0,3528      |

The equation 4 computes the difference between the accuracy of  $A_i$  and  $A_j$ . The equation 5 calculates the standard deviation of the sample estimate of the difference. The equation 6 computes the z-score for the difference.

If  $z(A_i - A_j) > 0$  then  $A_j$  overcomes  $A_i$  and if  $z(A_i - A_j) \geq 1.29$  then  $A_j$  overcomes  $A_i$  with 80% degree of confidence.

If  $z(A_i - A_j) \leq 0$  then  $A_i$  overcomes  $A_j$  and if  $z(A_i - A_j) \leq -1.29$  then  $A_i$  overcomes  $A_j$  with 80% degree of confidence.

## 5. Results and Discussion

In the results of the experiment, the classifier that showed the best performance was the SVM. The Table 2 shows the results of comparative analysis of classifiers used.

- In the comparison between Naive Bayes and SVM, where  $A_i =$  Naive Bayes and  $A_j =$  SVM, we have  $z(A_i - A_j) > 0$  and  $z(A_i - A_j) > 1.29$ , therefore the SVM outperforms Naive Bayes with confidence level of 80%.
- In the comparison between SVM and C4.5, where  $A_i =$  SVM and  $A_j =$  C4.5, we have  $z(A_i - A_j) < 0$ , therefore the SVM outperforms C4.5, but does not overcome the level of confidence of 80%, because  $z(A_i - A_j) > -1.29$ .
- In the comparison between C4.5 and Naive Bayes, where  $A_i =$  C4.5 and  $A_j =$  Naive Bayes, we have  $z(A_i - A_j) > 0$ , therefore the Naive Bayes outperforms C4.5, but does not overcome the level of confidence of 80%, because  $z(A_i - A_j) < 1.29$ .

During the experiment it was observed that the attributes most relevant were: view, like and dislike.

### 5.1. Treats to Validity

Although we have achieved good results with our experiments, we verified three treats to validity of our work: i) *The small number of volunteers* (15) for evaluate the video lectures, and the possibility that different users having rated the videos considering different weights to evaluation criteria, for example, a user may consider that the audio quality is more important than the teacher's didactic, but another user may consider that the didactic of the teacher is more important than quality of audio and video; ii) *The limited domain and limited dataset* - we performed the experiments with video lectures in the Math domain, however we intend to perform experiments in other domains and increase the number of instances in our dataset; iii) *The limited number of attributes* - in our work we used only nine attributes. We intend to perform new experiments increasing the number of attributes, such as: analysis of the subtitles, audio and image quality, type of lesson (theoretical or problem solving), resources used in the video lecture (blackboard, slides or pen and paper), among others.

**Table 3. Confusion Matrix from SVM classifier.**

| a | b | c | d | e |                |
|---|---|---|---|---|----------------|
| 5 | 0 | 0 | 0 | 0 | a = Excellent  |
| 1 | 1 | 0 | 0 | 0 | b = Good       |
| 0 | 0 | 1 | 0 | 0 | c = Average    |
| 1 | 1 | 0 | 1 | 0 | d = Bad        |
| 0 | 0 | 0 | 0 | 0 | e = Inadequate |

## 5.2. Experiment: SVM classifier as predictive model

After the comparative analysis experiments presented in Section 4, the SVM classifier was selected as the predictive model. To evaluate the model, we used 11 new video lectures that were manually evaluated by the same method presented in Section 4.1. The model correctly classified 8 instances (72.7273%) and 3 incorrectly (27.2727%). The confusion matrix can be seen in Table 3 and shows the predicted and actual classifications. In the Table 3, the rows correspond to desired results, namely the class to which belongs the pattern shown, and the columns represent the classification resulting from the use of the SVM. Through this matrix we can view errors precisely where they occurred, and any value outside the main diagonal symbolizes an error obtained.

## 6. Conclusion

In this work we present an analysis of classifiers to determine the quality of video lectures. We conducted experiments with the classifiers: Naive Bayes, SVM and C4.5. The classifier that showed the best performance was the SVM, that was selected as a predictive model. We conducted an experiment and preliminary results indicate good quality of the SVM as prediction model.

In our future work, we will conduct experiments with more users and videos. The analysis performed in this paper is part of an initial work to build a predictive model to determine the quality of video lectures. We plan to improve the prediction model with other factors such as context, viewing history, audio quality, and relationships between user can be used to provide better results. In the future, the authors plan to integrate this predictive model in a recommendation system of video lectures. This system will be integrated in an educational system.

The recommendation system integrated with a educational system can bring many benefits. The actual performance of the student in the activities of the educational system can be used in evaluating the quality of videos, making it possible to measure if the videos lectures helped concretely the student learn. Other information educational system can help identify students' preferences regarding the videos, for example, a student may like short videos and slides while another student likes long videos with the teacher using a blackboard.

## References

- Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17(6):734–749.



- Baeza-Yates, R. A. and Ribeiro-Neto, B. A. (2011). *Modern Information Retrieval - the concepts and technology behind search, Second edition*. Pearson Education Ltd., Harlow, England.
- Bielza, C. and Larrañaga, P. (2014). Discrete bayesian network classifiers: A survey. *ACM Comput. Surv.*, 47(1):5:1–5:43.
- Boll, S. (2007). Multitube—where web 2.0 and multimedia could meet. *IEEE MultiMedia*, 14(1):9–13.
- Bouckaert, R. R., Frank, E., Hall, M. A., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2010). WEKA—experiences with a java open-source project. *Journal of Machine Learning Research*, 11:2533–2541.
- Chang, C.-C. and Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27.
- Domingos, P. and Pazzani, M. (1997). On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3):103–130.
- Ferro, M., Nascimento, H., Paraguaçu, F., Costa, E., and Monteiro, L. (2011). Um modelo de sistema de recomendação de materiais didáticos para ambientes virtuais de aprendizagem. In *XXII Simpósio Brasileiro de Informática na Educação (SBIE 11)*, Aracaju, SE.
- Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Ruggieri, S. (2002). Efficient c4.5 [classification algorithm]. *Knowledge and Data Engineering, IEEE Transactions on*, 14(2):438–444.
- Silva, P., Costa, E., and Fonseca, B. (2013). Uma abordagem para provimento de recursos em um ambiente interativo de aprendizagem. In *XXIV Simpósio Brasileiro de Informática na Educação (SBIE 13)*, Campinas, SP.
- Taheri, S., Mammadov, M., and Bagirov, A. M. (2011). Improving naive bayes classifier using conditional probabilities. In *Proceedings of the Ninth Australasian Data Mining Conference - Volume 121*, AusDM '11, pages 63–68, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.
- Wang, G. (2008). A survey on training algorithms for support vector machine classifiers. In *Networked Computing and Advanced Information Management, 2008. NCM '08. Fourth International Conference on*, volume 1, pages 123–128.
- Yang, J.-C., Huang, Y.-T., Tsai, C.-C., Chung, C.-I., and Wu, Y.-C. (2009). An automatic multimedia content summarization system for video recommendation. *Educational Technology & Society*, 12(1):49–61.
- Zhao, X., Luan, H., Cai, J., Yuan, J., Chen, X., and Li, Z. (2012). Personalized video recommendation based on viewing history with the study on youtube. In *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service, ICIMCS '12*, pages 161–165, New York, NY, USA. ACM.