Do our eyes mirror our information experience?

Katherine J. Arvanitaki

3979849

MSc Computing Science

Algorithmic Data Analysis

Department of Information and Computing Sciences

Universiteit Utrecht

Master Thesis

Supervisor: dr. dr. Egon L. van den Broek

2nd reader: prof. dr. Remco C. Veltkamp

31/08/2016

Abstract

Information eXperience (IX) is hypothesized to be a complex function of the user's perceived complexity, comprehensibility, and interest. This study links these appraisals to eye behavior of 28 subjects. Eye behavior was operationalized by parameters of the eye's fixations, saccades, and blinks, providing a pattern space. This pattern space was used to develop and validate several models. Random forests, support vector machines, k-nearest neighbors, neural networks, and regression models were used to generate these models. These models predict complexity, comprehensibility and interest, respectively 96.87%, 97.65%, and 90.63% of the cases; but, in parallel, indicate that the relation between the three appraisals is complex. Nevertheless, this research can serve as an initial step towards the foundation of a next-generation wearables that enable true IX, personalized information filtering, access, and retrieval.

Acknowledgements

I have gained many memorable experiences and knowledge during this thesis project. Egon, I am particularly grateful to you for your immense guidance, help, dedication and patience during this project. Frans, I am also very grateful to you for providing me the dataset from you PhD dissertation and for all you help during this project. Moreover, I am extremely grateful to my parents for always being there for me, for supporting me in any possible way and pushing to be better. Lastly, I would like to thank my friends for supporting and inspiring.

Table of Contents

Part I. Research introduction

1) Introduction

When reading an article, people are not only attracted by the subject and context. Although unconsciously, they seek a balance between interest, comprehensibility and complexity (van der Sluis, van den Broek, Glassey, van Dijk, & de Jong, 2014). When people find such an optimal balance, they arrive at their "sweet spot of interest" (Silvia, 2006; van der Sluis, 2013). However, this sweet spot is highly context dependent, relying on the information at hand, its modality, the reader, his emotions and mood, and the environment he is in, to mention a few. Hence, ideally, continuous, real-time adaptation, even stronger than mere personalization, is needed to arrive at the sweet spot (Janssen, van den Broek, & Westerink, 2009), more often than simple coincidence.

Here, we propose to explore eye-tracking as a means to achieve such real-time adaptation. Paulo Coelho wrote in Manuscript Found in Accra, "The eyes are the mirror of the soul and reflect everything that seems to be hidden; and like a mirror, they also reflect the person looking into them". Information processing is not limited to reading articles (or books), it is what humans do constantly throughout various modalities. Most noteworthy is our daily behavior when browsing the internet. Via various strategies (e.g., cookies; Mor, Riva, Nath, & Kubiatowicz, 2015), the information presented to us is already personalized. However, as it is done now, it frames our information via an information bubble (Pariser, 2011). And, although, on the one hand, personalized; on the other hand, it simply provides us more of the same (Van der Sluis, 2013). Also, actual personalization is undermined, as the PageRank algorithm (Franceschet, 2011) and its derivatives provide more information that the majority likes, which is not necessarily the same as what you like.

Based on user's personal interests, filtering and recommendation systems can be improved, beyond their content-, collaborative-, or property-based algorithms (Garcin, Faltings, Donatsch, Alazzawi, Bruttin, & Huber, 2014). Students and scholars can get reading materials that are both educational and linked to their interests (Hidi & Renninger, 2006). Similarly, newspapers can be personalized (Kleinnijenhuis, 1991). Serious Games can be put into practice much more efficiently, when personalized (Deterding et al., 2011; Romero et al., 2015).

With the steep rise of low budget wearables (e.g., Arduino toolkits), also low budget, consumer ready mobile eye trackers have emerged. Moreover, eye-tracking is envisioned to be integrated in various near future electronic devices such as smart phones and smart glasses (e.g., Google glass). Consequently, eye-tracking data can be conveniently linked to other data sources and used to understand the user's reading experience and help to reach reader's sweet spot (Reichle et al., 2010; Bai et al., 2008; Doherty et al., 2010; Bulling & Gellersen, 2010; Jones et al., 2008;).

Eye-trackers are used to define "areas of interest", over specific segments of information and explore the fixations, saccades and blinks that occur at those parts (Jacob and Karn, 2003). As such, eye-tracking features have been used to give an outlook on how people make decisions and reason (Balatsoukas & Ruthven, 2012). To predict syntactic processing complexity, Demberg and Keller used eye-tracking data from 10 participants reading 51,000 words of newspaper text. Rayner (2006) showed that eye movements are sensitive to difficult text passages, as processing times and the number of fixations increased, when text is difficult.

As discussed by Liversedge et al.'s (2011), eye movements have been investigated to study sentence processing since 1967. Cameras were used to examine what part(s) of a sentence the readers fixated. Later in 1982, more advanced hardware was introduced, that could register fixation duration and position. That lead to studies for parsing and sentence

investigation. So, initial studies were focused on separate and specific sentences that were used to test certain theoretical hypotheses. In more recent years (2005), there is a switch to gathering data from eye movements, during the time people read texts. This lead to a less experimental controlled environment, but this way is more normal to comprehend sentences.

Eye trackers are being used to create eye-based human-computer interactive systems, using gaze as a pointing method and to test the dynamic of interfaces, by exploring the areas where users look at in the screen. In virtual reality gaze selection is used over hand pointing (Duchowski, 2007).

Research eye tracking studies for reading and processing, used fixations and gazes as indicators for learning (Mason et al., 2013). Also, an eye tracker was used to examine visual attention while students were solving multiple-choice science questions, in order to identify which parts of the problem, the students paid the most or the least attention (Tsai et al., 2012).

This thesis continues with Section 2 in which we describe the acquired data. Next, we present our methodology and processing pipeline. Section 4 and 5 describe respectively the data and feature preprocessing and the classifiers used, alongside with the results. We end with Section 6 that presents the discussion, including a concise conclusion.

Part II. Modeling / signal processing

2) Data

This thesis uses a set of available eye-tracking data (Van der Sluis et al., 2014). It was recorded using the SMI BeGaze 2.4[1]. The data set includes data from 30 participants (22 males and 8 female), with a mean age of 28.60 (SD = 6.06). All participants had BA or MA degree or

[1] http://www.smivision.com/en/gaze-and-eye-tracking-systems/products/begaze-analysis-software.html

were persuing it. None of them were native English speakers; but, they rated their reading literacy as high (M = 4.63, SD = .62, range 1-5, 5 highest). At the beginning of the experiment, the participants answered a demographics and background questionnaire, which addressed the following items: gender, age, nationality, educational background, prior knowledge, personality traits, English reading proficiency, and visual acuity. These questions were included to verify whether or not there is a correlation between this information and the eye tracking data gathered. The participants read 18 articles in a randomized order and immediately after each article, they answered an experience questionnaire. This questionnaire included nine questions, three questions for each of the three constructs appraised complexity, appraised comprehensibility, and interest (again, see: Van der Sluis et al., 2014). For appraised complexity, two seven-point scales, were used, complex-simple, and easy to read-difficult to read. For appraised comprehensibility, the appraised comprehensibility scale (Silvia, 2008a) was used. For interest, two 7-point differentials were used, interesting-uninteresting and boring-exciting. The 18 articles were selected from articles from The Guardian[2], which were truncated after 1,200 characters and the layout was stripped. The cutoff point was placed before the end of the word at position 1,200, and three dots were added to indicate the story normally would continue. The articles were divided into three levels of complexity (which are: low, medium, and high) and all concerned different topics. The final 18 articles were preselected from the lower, middle and upper part of the distribution of textual complexity and then selected based on suitability. Textual complexity is linked to processing difficulty, meaning, the level of difficulty experienced when processing new information. In order to calculate the textual complexity of the texts, four approaches were used, traditional (regarding word length), familiarity (word frequency), priming (information

[2] http://www.theguardian.com/

density measure, for characters and words), and dependency locality (the cost of processing time based on dependences) (Van der Sluis, et al., 2014).

During this experiment, the aforementioned eye-tracking device was used for all the participants. The eye-tracker has a build in fixation, saccade, and blink detector. The detector uses a dispersion based algorithm (Blignaut, 2009), first detects fixations, with a minimum duration of 80 *ms* (Goldberg, 2000). Humans typically alternate between saccadic eye movements and fixations. A blink is captured by SMI BeGaze as a unique type of fixation, where the horizontal and vertical gaze position equals to zero. The data extracted with SMI BeGaze contains the following events (or features):

- fixations: eye-movements that are identified as a pause, a visual gaze at a location of interest;
- saccades: Rapid (fast) eye-movements between fixations; and
- blinks.

From the completed questionnaires, for each article of each participant, we calculated the average scores for interest, complexity, and comprehensibility.

3) Process pipeline

As a first phase of the analysis, we performed a statistical analysis (Appendix A). The field of pattern recognition, is being researched for at least 60 years and can be divided into four approaches (Jain, 2000; Bishop, 2006):

1) Template matching, regarding samples, pixels and curves, which are recognized using correlations and distance measures, using the criterion classification error.

2) Statistical, that utilizes features, using discriminant functions and as a criterion the classification error.

3) Structural, using primitives (basic patterns), that recognizes rules and grammars and uses as criteria an acceptance error.

4) Neural networks, using samples, features and pixels that recognize using network functions and as a criterion the mean square error.

This having said, this division of approaches can be debated. Neural networks, like feed-forward networks, and multi-layer perceptron (MLP) networks can be considered as part of statistical pattern recognition (Bridle, 1990). Syntactic pattern recognition can be separated from statistical pattern recognition, because a lot of times structural information, cannot always be transferred to a feature vector (Albus et al., 2012).

Moreover, machine learning can be divided to two categories:

1. Supervised learning,

Where the aim is to learn a model from the input (training) data, including a target variable. The target variable, is the label that is used to classify each data point.

2. Unsupervised learning

In this case, the input data does not include a target variable (Alpaydin, 2014; Blum, 1997). In other words, there is no class variable. Instead, the variables of each data point help to identify cluster(s) of data points (Bishop, 2006). Unsupervised learning is also referred to as data clustering, where groups (clusters) of data points are detected (Jain, 2010).

In this study, we will apply statistical pattern recognition and neural networks, using supervised learning. For supervised learning problems, when the goal is to assign each instance to a category, where the number of categories (classes) is finite and discrete, then it is a classification problem. On the other hand, when at least one of the target variables is a continuous number, then it is a regression problem.

Its process is outlined in Figure 1. First, we have the signals, which we accumulated from the eye tracking device. Each signal is going to be filtered, and will go through the process of outlier removal. Moreover, according to the signal, one or all of the process: quantization, sampling rate, and baseline, will be used. The features and parameters outlined in the third and fourth step will be calculated. Then, we will select the appropriate parameters, apply machine learning algorithms, and select the best performing ones. In the next section, these phases will be discussed in more depth.

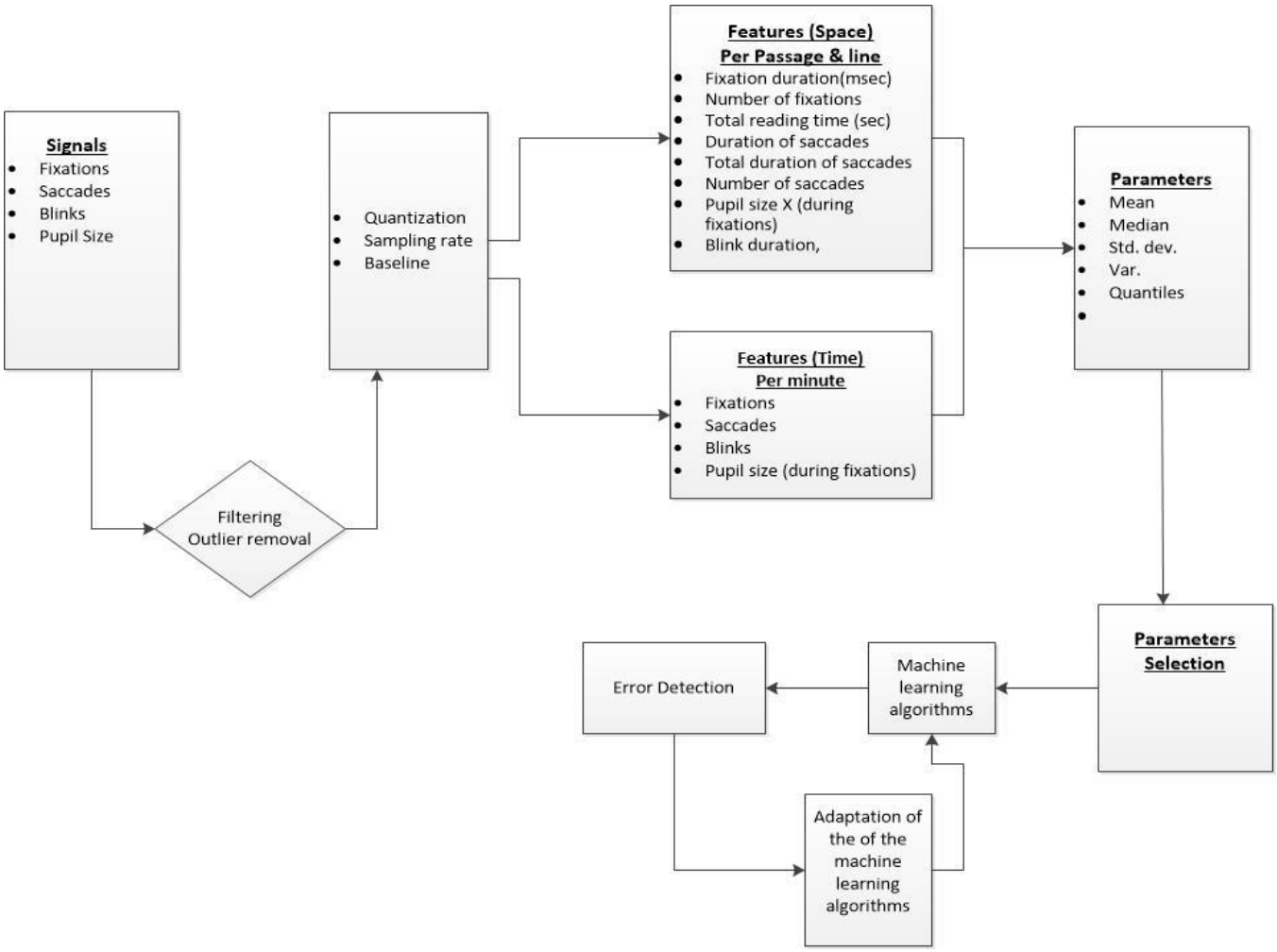

Figure 1: Processing Pipeline

4) Preprocessing

Both the questionnaires and the eye tracking data (i.e., fixations, saccades, and blinks) need to be preprocessed, before it can be used for machine learning. This preprocessing consists of six phases discussed next: i) feature extraction, ii) parameter extraction, iii) parameter selection (or reduction), iv) outlier removal, v) normalization, and vi) data balancing.

a) Feature extraction

On average fixations have a duration of 250 msec. (Ajanki, 2009), ranging from 150 to 300 msec. (Kliegl & Engbert 2005). In line with their suggestions, to detect fixations, the lower

and upper threshold was set on respectively 100 and 300 msec. Additionally, several other fixation features have been proposed, such as fixation duration (i.e., the duration of the first forward fixation on a target word) and gaze duration (i.e., the aggregated duration of all fixations on a target word, when initially encountered) (Cole et al., 2010). However, these features link to specific words, which is a focus the current research does not have. Therefore, these features are not included. For entire passages, Rayner et al. (2006) considered average fixation time, number of fixations, and total reading time. This triplet of features is included. For distinct lines, Inhoff and Rayner measured fixation duration and Ziefle et al. measured the number of fixations (Ziefle et al., 1998; Inhoff et al., 1986). The latter two features have not been included in the current research, because in this paper, the goal is to understand how users perceive different texts as a whole and not unique sentences. Saccades are on average 7–9 characters in size and last 30 msec. (range: 10-100 msec.) (Liversedge & Findlay 2000; Balatsoukas & Ruthven 2012). Saccade duration is one of the three saccade features, complemented with: total time spent in saccades and the number of saccades made (Wiley & Rayner, 2000).

While at rest, people produce 12 and 19 blinks per minute, with a duration between 100 and 400 ms. (Bulling, 2009). An increase in blink frequency, indicates light fatigue. An increase in blink duration, accompanies severe sleepiness.

Pupil size has been used to indicate information processing (Partala & Surakka, 2003). The average pupil size, was measured during text processing and showed that when processing difficult words, pupil dilation is higher (Hyönä et al., 1995).

Of each of these features four parameters are extracted, namely: mean, standard deviation, variance, and median.

b) Parameter selection

To enable efficient classification of eye behavior, the total of features will be assessed on their value in spanning up a pattern space. For this purpose, we apply the linear transform: Linear Discriminant Analysis (LDA) (Jain et al., 2000; Bishop, 2006). Table 1 shows the features selected after the transformations. For each one of the features in Table 1, the four aforementioned parameters were applied.

Table 1: Selected features.

Events	***Features***		
Fixations	Duration of each fixation.	Number of fixations.	Total duration of fixations.
Saccades	Duration of each saccade.	Number of saccades and regressions.	Total duration of saccades.
Blinks	Duration of each blink.	Number of blinks.	Total duration of blinks.
Pupil	average pupil size (x- axis)		

c) Outlier removal

To remove unwanted anomalies in the data, we tested several outlier removal techniques, namely:

- Removal of data +/- *x* standard deviations from the mean, with the values 2, 2.5, and 3 taken for *x*
- Removal of data outside a quantile window, with the quantile set on 10%, 20%, and 30% of the head and tail of the data distribution.

In order to decide which outlier removal approach was best suitable, we applied a linear regression model, for each approach. Considering the summary output values Multiple R-squared and *p-value*, they showed that the removal of data with values: $(mean +/- 2.5SD)$ was most efficient (van Den Broek & Westerink, 2009).

Moreover, data for one article of medium textual difficulty were completely removed, because it was detected from van der Sluis (2014) to be an outlier. So seventeen articles were used for the rest of the research.

d) Normalization

Since people show significant interpersonal variation on behavioral data (van den Broek, 2011), data normalization is required at a participant level. Since the data set we have is extracted from multiple and unique humans reading multiple articles, we are going to apply normalization techniques, for each person separately. The reasoning for this choice is, that each person has different eye-movement regularities. However, normalization is not applied for each article separately, since the goal of this research is to have a robust model, not dependent on a kind of text.

In order to normalize the features among the participants we used this technique: $\log(feature) - (mean\ baseline)/(Stadanrd\ Deviation\ baseline)$ (van Den Broek, 2009). Where the $mean\ baseline = mean(\log(feature))$ and $SD\ baseline = Standard\ Deviation(\log(feature))$. The baselines were calculated based on the eye movements: fixation, saccade, blink and pupil size, during the demographics and background questionnaires for each participant. Also for each participant we used the normalization approach described.

e) Imbalanced Data

The data set we acquired to use in this research is based on 28 humans and each of them has a different appreciation of the 18 articles they read. So, although the articles were equally separated to three categories of textual complexity, the three appraised dimensions (complexity, comprehensibility and interest), were not equally distributed. Therefore, in order to have approximately the same number of instances for

each class of the three dimensions, we considered using a hybrid algorithm to generate instances.

From the figures below (Figure 2 – Figure 13), it is clear that the classes are un-balanced, therefore it is necessary to apply a balancing technique, so that the classifiers can be trained for all the classes equally. Thus, a technique to acquire a more balanced number of instances for each class was required. Further down in this thesis in Section 5, the balancing technique that was applied is described in detail. This is crucial as the original dataset was already small, with only 459 instances.

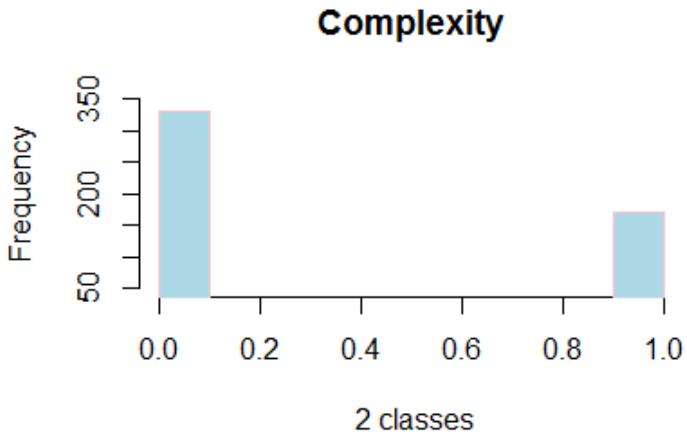

Figure 2: Distribution of 2 classes for the dimension Complexity.

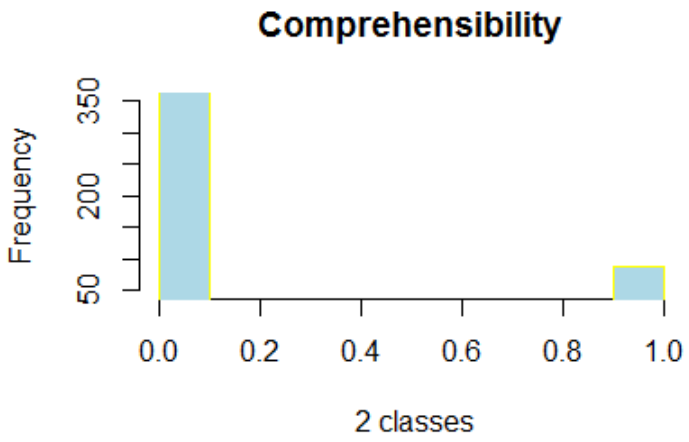

Figure 3: Distribution of 2 classes for the dimension Comprehensibility.

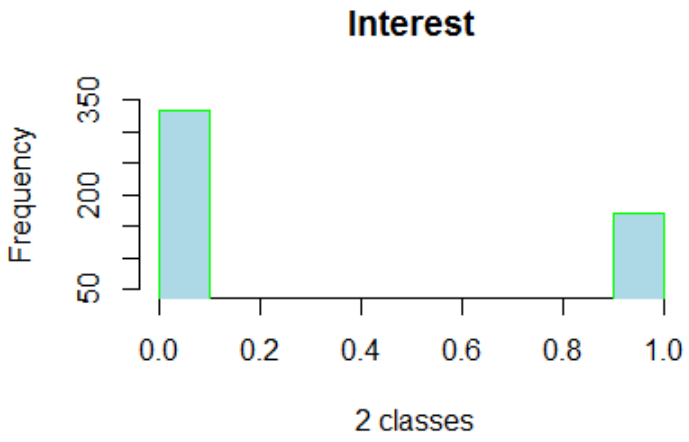

Figure 4: Distribution of 2 classes for the dimension Interest.

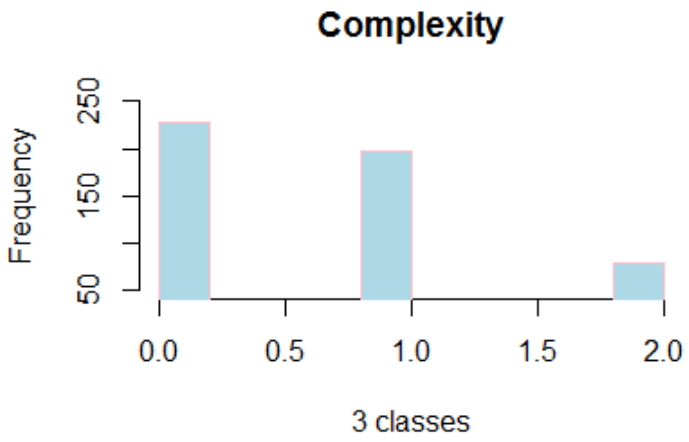

Figure 5: Distribution of 3 classes for the dimension Complexity.

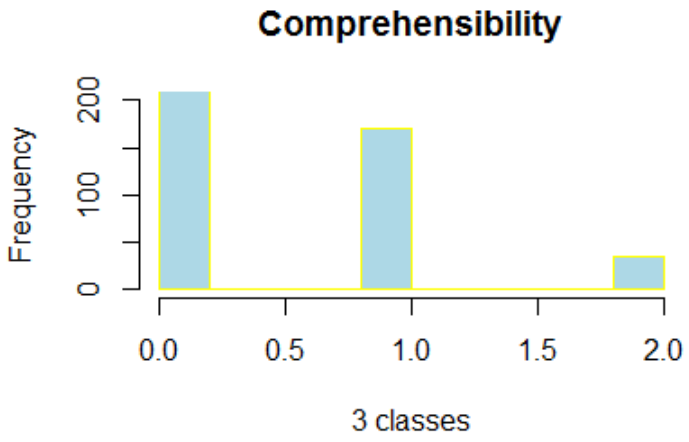

Figure 6: Distribution of 3 classes for the dimension Comprehensibility.

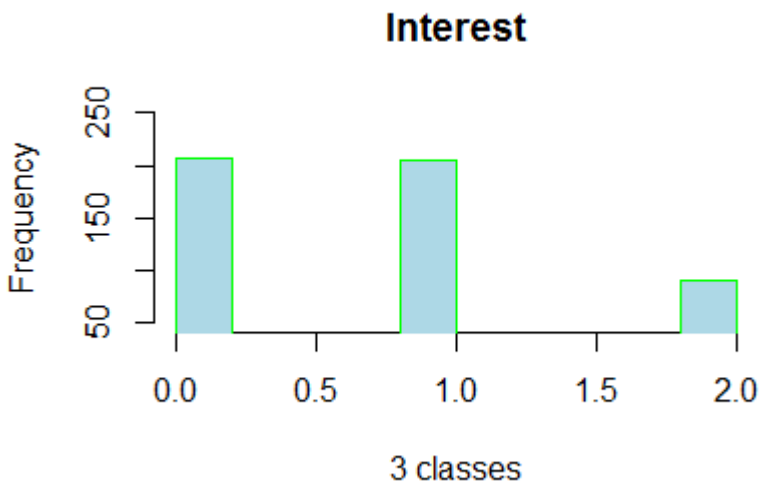

Figure 7: Distribution of 3 classes for the dimension Interest.

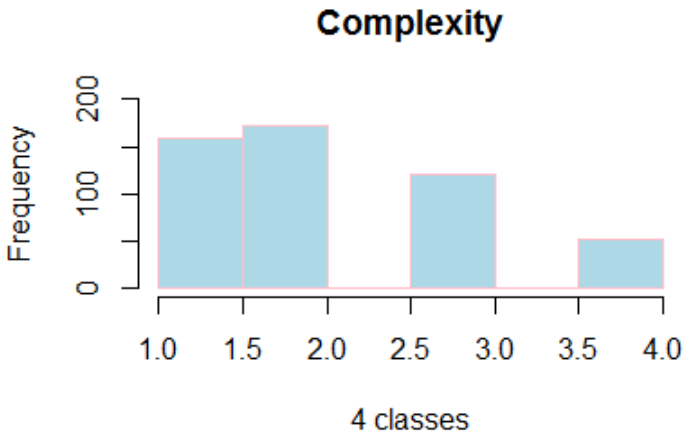

Figure 8: Distribution of 4 classes for the dimension Complexity.

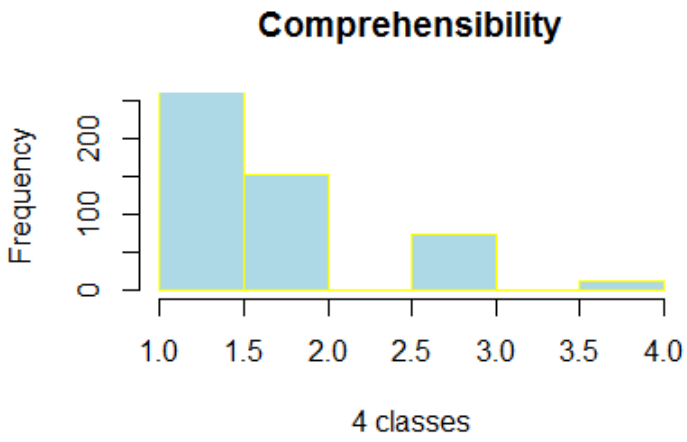

Figure 9: Distribution of 4 classes for the dimension Comprehensibility.

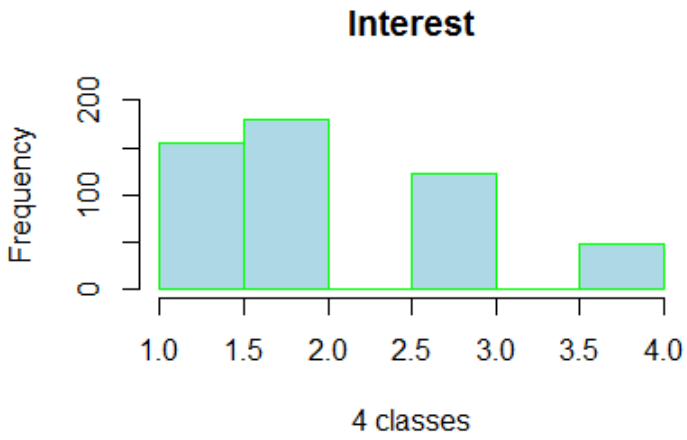

Figure 10: Distribution of 4 classes for the dimension Interest.

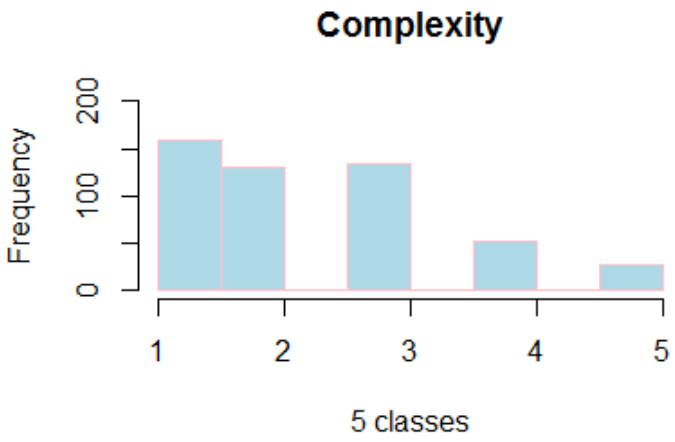

Figure 11: Distribution of 3 classes for the dimension Interest.

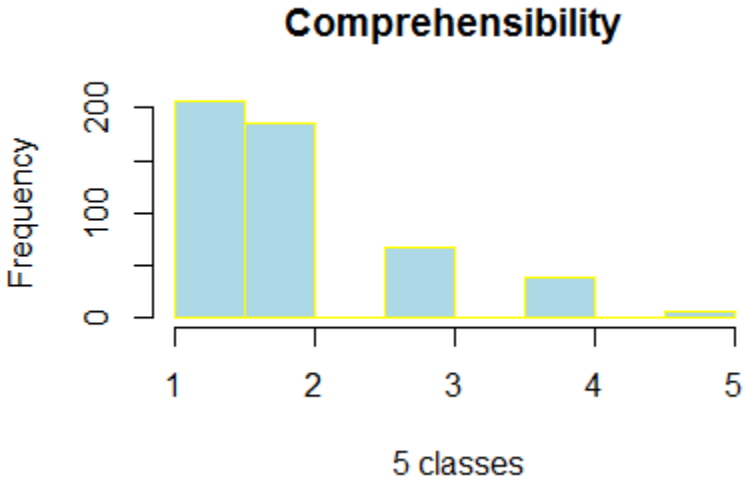

Figure 12: Distribution of 3 classes for the dimension Interest.

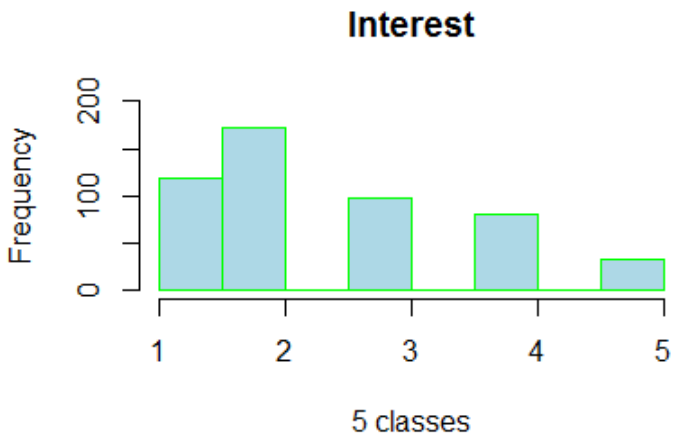

Figure 13: Distribution of 3 classes for the dimension Interest.

5) Classifiers

Utilizing the eye-tracking features we explored, we are going to try to classify the three labels complexity, comprehensibility and interest. We are going to investigate our hypothesis by applying two techniques for prediction, regression and classification. We used R Studio[3] for this section.

[3] https://www.rstudio.com/

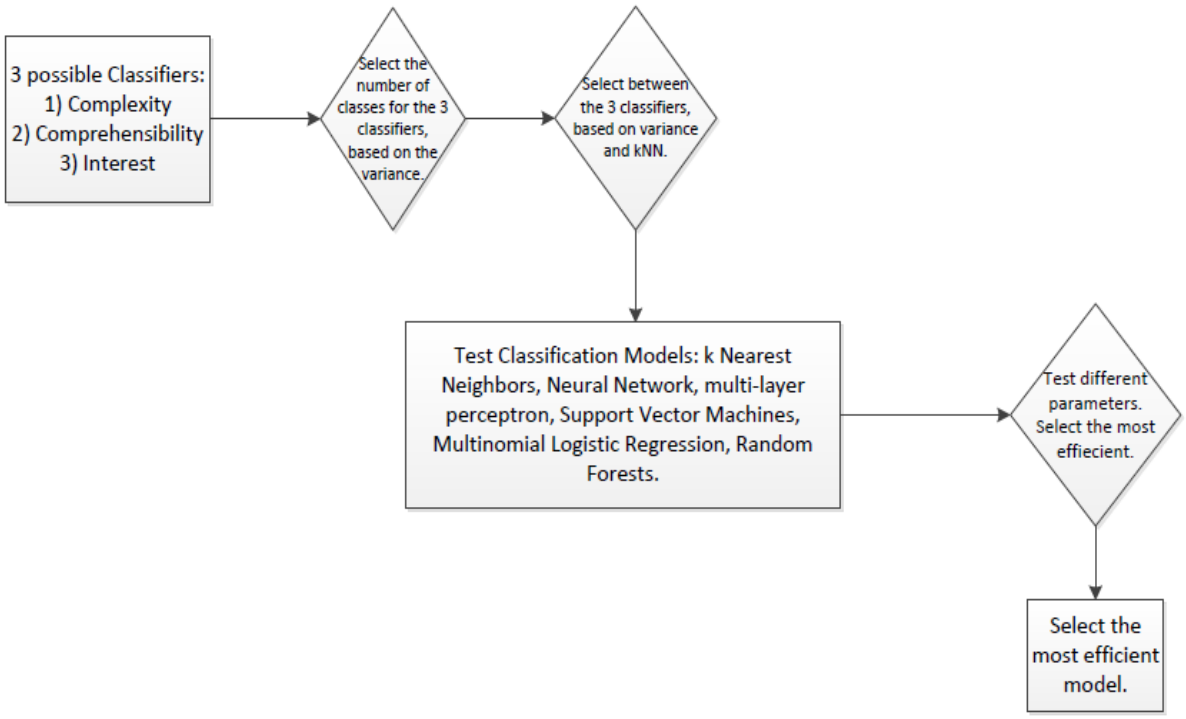

Figure 14: Classification flow diagram.

a) Regression

We initially considered regression to predict the three dimensions, complexity, comprehensibility and interest. But unfortunately, utilizing regression models, resulted to very low prediction rates (< 20%). This is due to the big range of scores for the three labels, which are between [1, 7]. Later on, the scores are divided to classes, in order to avoid this problem and have a more fix classification problem.

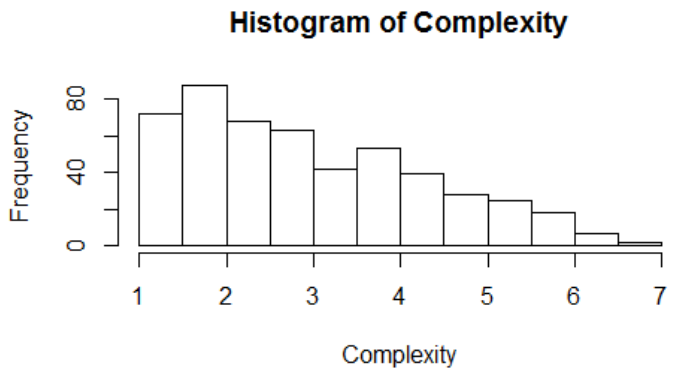

Figure 15: Distributions of the scores [1-7] for complexity.

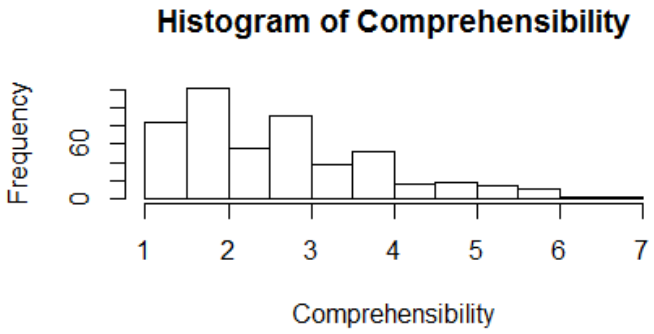

Figure 16: Distributions of the scores [1-7] for comprehensibility.

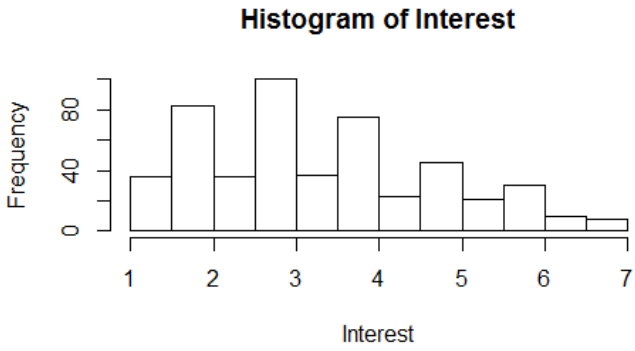

Figure 17: Distributions of the scores [1-7] for interest.

b) Classification

Using a 1-7 scale, the participants judged the texts they read on three dimensions: simple – complex, comprehensible – incomprehensible, and interesting – uninteresting. Their judgements provided the required labels for the classification process. These judgements were grouped into 2, 3, 4, and 5 classes for both single-label and multi-label classification. Figure 14 illustrates the classification flow diagram regarding the rest of this section.

a) Single Label

1. *k-NN*

Given a feature vector x from the test set, the k-NN algorithm finds the k feature vectors in the training set that have the smallest Euclidean distance to x. The class that is represented the most among these k feature vectors will be assigned to x. In case of a tie, a random class is selected from the top most represented classes. In case that there a several feature vectors available for the k^{th} nearest neighbor, all of those vectors will be considered in the decision process (Bishop, 2006). We have run the k-NN classification for k = 1,2, …,50. For each run, we have calculated the relative prediction accuracy.

2. *Support Vector Machines*

Support Vector Machines (SVM) try to maximize the distance from the decision boundary to the data points (Bishop, 2006). The kernel function is used to predict the test (not trained) data. We used 3 different kernels that are available from the SVM classifier (Meyer, 2003). These kernels are: radial, sigmoid and polynomial. We used the values 1 and 100 for the parameter cost and the parameter setting: type = C- classification. We used the library e1071 and the function SVM.

3. *Multi-Layer Perceptron*

A multi-layer perceptron is consisted of multiple layers of nodes and each layer is connected to each other. Besides the input nodes(features), each node is a neuron. (Kruse et al. 2013). The backpropagation error function, uses the next layer to define the error of the previous layer. We used the R library RSNNS, to use the feed-forward network, multi-layer perceptron, trained by error backpropagation.

4. *Neural Networks*

Here, we will test the algorithm based on neural networks (NNET). We use the nnet function from the nnet package of R to train a neural network. The nnet function calculates the

most suitable weights for the neural network and returns the fitted values of the data we use to train our model. The number of hidden layers is defined by the parameter size. By increasing the number of hidden layers, the network becomes more complicated. The relevant parameters are: package of R to train a neural network. The nnet function calculates the most suitable weights for the neural network and returns the fitted values of the data we use to train our model. The number of hidden layers is defined by the parameter size. By increasing the number of hidden layers, the network becomes more complicated. In order for our model to converge, a large number of iterations is needed. The number of iterations needed depends on the model, the number of hidden layers, and the decay. The seed was set at 54321. The parameters that we changed were size and decay, and we always used the following values for the other parameters: skip = T, softmax = T, maxit = 20000. The different values we tried for size were 2, 8, 10, and 12, and for decay we tried 0.0001, 0.001, and 0.01.

5. Random Forests

A random forest is a special type of tree for regression or classification, it is a big collection of (not correlated) decision trees. That instead of using the best split function, for all the features, it selects a subset of the features at random and then proceeds with the split (Liaw, 2002; Segal, 2004). The library *randomForest* was used, and the number of trees was set to 1000 and the parameter type to *classification*.

For all classifiers, the procedure cross-validation (CV) was used, which is used to prevent over-fitting. Here, we used a 5-fold CV method, where the data was split into four subsets and each time three subsets were used for training and one for testing. Meaning that four different test sets were investigated (Bishop, 2006).

The table below, shows the prediction scores for each label and classifier. It is visible that 'Comprehensibility' has the highest accuracy among the three labels. Also, by increasing the

number of classes, the accuracy declines. When using neural networks and having to predict multiple classes, there is the negative factor that individual neurons are trained based on a certain class or classes. This can lead to “ambiguity and/or uncovered feature space regions” (Ou, & Murphey 2007). This explains the very low prediction rates when using neural networks and multiple classes (Table 2).

Table 2: Prediction rates for the three labels (average scores), predicted by each classifier.

	Classifiers					
Labels/Classes	*Random Forest.*	*Multi-layer perceptron.*	*Neural networks.*	*Support Vector Machines.*	*k- Nearest Neighbors.*	*Multinomial logistic regression.*
Complexity/2, Complexity/3, Complexity/4, Complexity/5	2c: 64.04%, 3c: 35.95%, 4c: 26.97%, 5c: 22.47%	2c: 47.19%, 3c: 28.09%, 4c: 26.97%, 5c: 24.72%	2c: 66.29%, 3c: 49.44%, 4c: 13.48%, 5c: 15.73%	2c: 60.67%, 3c: 44.94%, 4c: 29.21%, 5c: 30.34%	2c: 60.67%, 3c: 48.31%, 4c: 28.09%, 5c: 33.71%	2c: 60.67%, 3c: 37.08%, 4c: 28.09%, 5c: 30.34%
Comprehensibility/2, Comprehensibility/3, Comprehensibility/4, Comprehensibility/5	2c: 88.76%, 3c: 42.67%, 4c: 49.44%, 5c: 46.07%	2c: 64.04%, 3c: 41.57%, 4c: 32.58%, 5c: 31.46%	2c: 88.76%, 3c: 59.55%, 4c: 8.99%, 5c: 5.62%	2c: 84.27%, 3c: 46.07%, 4c: 44.94%, 5c: 37.08%	2c: 88.76%, 3c: 56.18%, 4c: 55.06%, 5c: 38.20%	2c: 89.89%, 3c: 53.93%, 4c: 49.44%, 5c: 43.82%
Interest/2, Interest/3, Interest/4, Interest/5	2c: 60.67%, 3c: 42.67%, 4c: 41.57%, 5c: 38.20%	2c: 44.94%, 3c: 35.95%, 4c: 29.21%, 5c: 21.35%	2c: 74.16%, 3c: 40.45%, 4c: 10.11%, 5c: 14.61%	2c: 59.55%, 3c: 40.45%, 4c: 39.33%, 5c: 26.97%	2c: 76.40%, 3c: 42.67%, 4c: 23.60%, 5c: 22.47%	2c: 68.54%, 3c: 39.32%, 4c: 37.08%, 5c: 33.71%

i. Multi-Label

In machine learning, multi-label classification is a problem of multi-output classification, where multiple target labels can be assigned to each instance. Multi-label learning can be phrased as the problem of finding a model that maps inputs *x* to binary vectors *y*, rather than to scalar outputs as in the ordinary classification problem. In order to implement a multi-label classification, we used the package rFerns (Ozuysal et al., 2010), modified supporting multi-label classification.

For the multi-label classification, we use as labels the three average scores complexity, comprehensibility and interest. The aforementioned scores are from 1 to 7, so we divided the scores in two classes 0 and 1, where class 0 concerns the scores [1, 4) and class 1 concerns the scores [4, 7]. We conducted four different multi-label classification approaches. First, by applying all 3 labels and further 2 labels at a time, exploring the possible combinations.

Opposed to the previous one-label classification approaches, the joint dimensions' complexity and interest have the highest prediction rates. The prediction rates for 8 binary classes for the joint three dimensions were lower than 10%. Further, the prediction rate for 4 binary classes, for complexity and comprehensibility was 31.64%, for complexity and interest 24.05% and comprehensibility and interest 11.40% (Appendix B).

We used the SMOTE: Synthetic Minority Over-sampling Technique, to balance the data, by over-sampling the minority class and under-sampling the majority class (Chawla, 2002). The reasoning for that decision is that the original dataset was already not big enough, only 459 instances. SMOTE is a hybrid algorithm that generates instances in order to create balanced class distributions. Here, SMOTE is applied for the purpose of optimizing the classification process. We used the R library DMWR, which has an implementation of the algorithm SMOTE to balance the 3 classes. For each class, different parameters were used, since each label and class had a different number of instances. The table 4 below, demonstrates the prediction rates, using the classifier, Support Vector Machines, and the sigmoid which achieved the best predictions. The train and test set were split to 80 and 20 percent respectively, and divided in such way that each class had even number of instances in the tests sets.

Table 4: Prediction rates for the three labels (average scores), predicted by SVM classifier.

	Complexity	**Comprehensibility**	**Interest**
2 Classes	Score_1_2c: 96. 87% (svm)	Score_2_2c: 97.66% 125/128 (svm)	Score_3_2c: 90.62% (svm)
3 Classes	Score_1_3c: 84.37% (svm)	Score_2_3c: 85.16% (svm)	Score_3_3c: 82.81% (svm)
4 Classes	Score_1_4c: 74.22% (svm)	Score_2_4c: 75.78% (svm)	Score_3_4c: 72.66% (svm)

Using the new data sets created with the SMOTE algorithm mentioned above. The multi-label classification process for the new data sets, cannot be done one on one for each instance and label. The reason for that is that SMOTE was used for each dimension separately, therefore the new hybrid data were different for each dimension. Thus, each instance of each dimension was compared to each other, using the Euclidean distance. Below, Figure 18, shows the distribution for 8 binary classes and Figures 19, 20 and 21 for 4 binary classes.

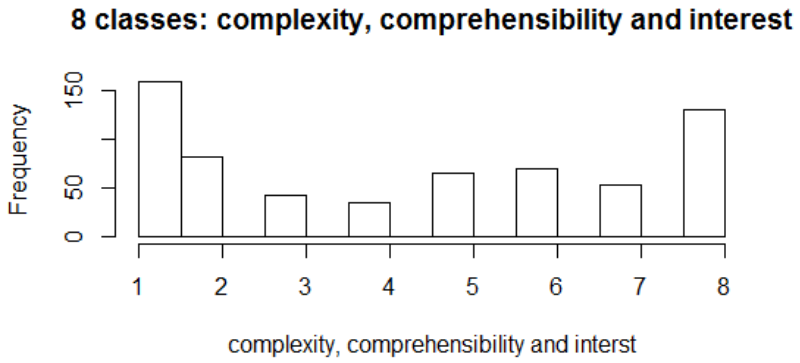

Figure 18: Histogram of 8 classes for complexity, comprehensibility and interest.

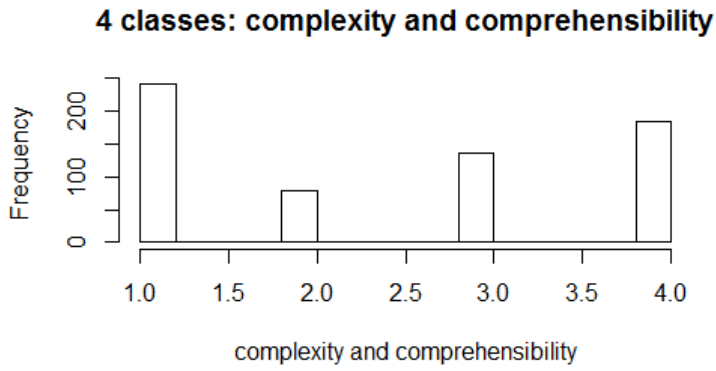

Figure 19: Histogram of 4 classes for complexity and comprehensibility.

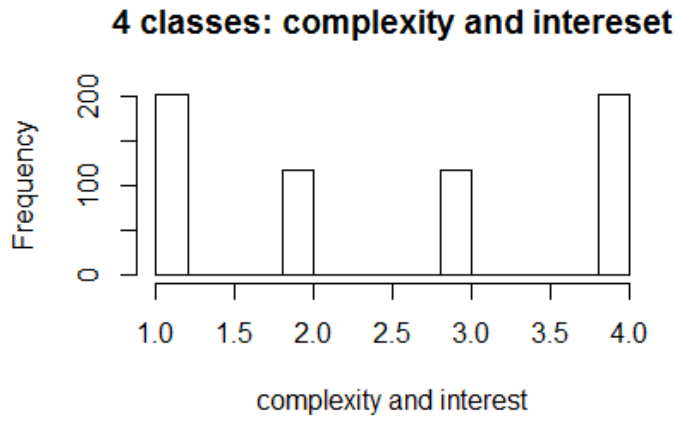

Figure 20: Histogram of 4 classes for complexity and interest.

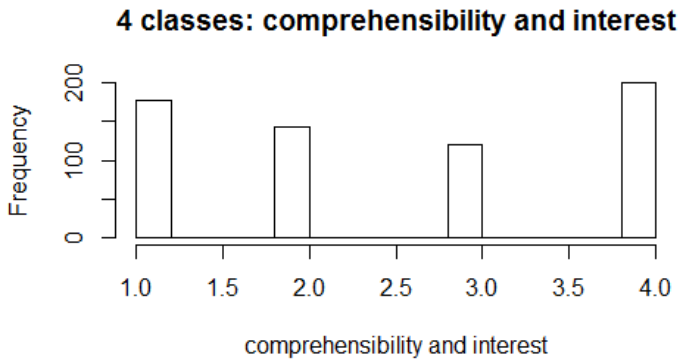

Figure 21: Histogram of 4 classes for comprehensibility and interest.

The prediction rate for four binary classes (complexity - comprehensibility), is 64.84%, for (complexity - interest) the prediction rate is 38.28% and for (comprehensibility – interest) the prediction rate is 43.75 % (Appendix C).

III. Closing

6) Discussion

This research explored whether or not the human eye (Jacob & Karn 2003) can serve as a channel to predict reader's text appreciation. For this aim, an available eye tracking data set from Van der Sluis (2013) was used.

Initially, we united the three appraisals complexity, comprehensibility and interest that make up reader's text appreciation, which resulted in three binary classes, making 8 classes to discriminate between. However, three of these classes suffered from a lack of data that made it impossible to build models upon. Therefore, the 8 classes were reduced to 5 classes, which data was predicted correctly in 37.14% of the cases.

With only 62.5% of the three dimensions filled with data, the three suggested dimensions for text appreciation (Van der Sluis et al., 2014) can be questioned. Therefore, we investigated how much of the data variance could be explained using the combinations of two dimensions. First, we explored the combination complexity and comprehensibility, which enabled 64.84% correct classification. The case where a text is considered as simple and incomprehensible occurred too little which deteriorated the classification rate for this class (i.e., 10%). When the labels complexity and interest were combined, the prediction rate was 38.28% and when comprehensibility and interest were combined, the classification results were the even lower. It is visible that complexity and comprehensibility have a higher correlation, than complexity with interest. According to section 5 (Figures 10, 11, 12), the combination of the dimensions' complexity and comprehensibility, indicates that these two dimensions demonstrate many similarities. Therefore, we suggest to combine the dimensions' complexity and comprehensibility.

Although the results for 2x2 models were better than those of the 2x2x2 models, they still can be considered as moderate at best. Consequently, we also explored the three dimensions separately, resulting in the following prediction rates: *complexity*: 96.87%, 84.38%, and 74.22%; *comprehensibility*: 97.65%, 85.16%, and 75.78%; and *interest*: 90.63%, 82.81%, and 72.66%, all with respectively 2, 3, and 4 classes. These results are good and seem to indicate that the relation between the three dimensions, as proposed in Van der Sluis et al. (2014) needs

further investigation. At least, a simple, linear relation between these three dimensions seems unlikely.

The aforementioned results were achieved with the use of the classification algorithm, Support Vector Machines (SVM) and the Sigmoid kernel. When tested against the other classifiers mentioned in Part II, for the balanced (after the use of SMOTE) SVM algorithm outperformed the other classifiers. The reason for that is the SVM can handle overfitting. It has been reported (Tang et al., 2009), that SVM together with rebalancing algorithms perform better and that imbalanced data can really hinder the performance of SVM.

The analyses and the model featured in this research were conducted with the use of a small and imbalanced dataset. To overcome this problem, with all analysis, the SMOTE algorithm was applied, which generates synthetic samples for the minority class, using *k* nearest neighbors of the instances in the minority class (Chawla et al., 2002). Also, it under-samples the majority class. Together, SMOTE realizes a larger and balanced dataset. However, its downside it that the synthetic samples may overlay among classes (He et al., 2008).

Future research could use an improved version of the SMOTE algorithm: the SMOTEBoost (Synthetic Minority Over-Sampling in Boosting) algorithm, which improves performances on minority classes (He et al., 2010). It would also be interesting to explore alternative algorithms for SMOTE, such as ADASYN: ADAptive SYNthetic sampling approach, which is claimed to outperform SMOTE (He et al., 2008). Another alternative for SMOTE is Cluster Based Synthetic Oversampling (CBSO), which prevents to create synthetic sample that are falsely classified (Barua, Islam & Murase, 2011).

As an alternative to generating new artificial data, the original dataset could also be extended with new data. Nielsen (2010) stated that 20 participants suit the needs for a

quantitative eye tracking studies. With a data set composed of data of 28 participants, Nielsen's requirement is met. However, the number of data points obtained from each participant is very small; in particular, given the data set's imbalance. The most obvious remedy for this problem is not to apply alternatives for SMOTE; but, to harvest more eye tracking data. Preferably from the same 28 participants. However, in practice, this can be problematic. Alternatively, eye tracking data from other, new participants has to be gathered. On the one hand, the interpersonal variance that will be introduced consequently may play a significant role in the process, which could deteriorate the results. On the other hand, this can also be considered as an ultimate test for the system's robustness.

Considering multi-label classification, in real-world implementations, similarly to the case of this thesis, labels can be depended to each other and some may be missing. A probabilistic model was build which can consequently learn and take advantage of multi-label dependencies and manipulates data with missing labels (Bi & Kwok, 2014). This model handles label dependencies by transforming the labels in the original label space, with the probabilistic model. When missing labels occurred, Bi and Kwok (2014), calculated values from the observed labels. Another solution for limited amount of data that is labeled, is the algorithm iMLCU (inductive Multi-Label Classification with Unlabeled data), that showed good results for labeled and unlabeled multi-label data. This algorithm optimizes semi-supervised learning, a strategy that exploits unlabeled data in the learning procedure, together with labeled data (Wu & Zhang, 2013).

In the future, other classification algorithms that perform better or are more suitable for this problem, can be used. Recently, the classification algorithm deep learning is successfully being used for large data sets (LeCun, Bengio & Hinton, 2015). Although, the data set in hand is

not large and deep learning cannot benefit this data model, in the future a different eye tracking data set, might benefit from this data model.

The data set used in this thesis was collected with, meanwhile outdated software and hardware tools. For future studies improved eye tracking methods should be included. For example, recently, promising software and hardware has been released, including the Pupil headset, a real-time eye tracker (Kassner et al., 2014). Its algorithms for pupil detection even works with users wearing eyeglasses or contact lenses and, hence, unconstrained use is possible. Moreover, this platform is “inexpensive and extensible” (Kassner et al., 2014). Additionally, in contrast to the SMI BeGaze 2.4 software, the Pupil software is open source, which makes it par excellence suitable for research purposes (West, Salter, Vanhaverbeke & Chesbrough 2014).

As future work, research for the raw data from the aforementioned eye tracking experiment can be conducted for feature mining. Therefore, features that have not been considered in this research, but have been validated to significant, may lead to a more efficient model.

As noted earlier in this discussion, the dimensions that define text appraisal, adopted in this paper require further investigation. One of the possible problems may have its origin in untangling often ill-defined psychological constructs. Eye tracking studies are also often claimed to unveil people’s workload and attention (Tsai et al., 2007). But, how to know what construct is assessed, remains problematic. Nevertheless, some interesting studies have been conducted in cognitive sciences, which could be related to the current study and, subsequently, investigated further. For example, McDaniel (2000) discovered that stories that are interesting need less attention resources from users and Smith (2001) showed that workload rises when the

complexity of a task increases. These amongst many other studies illustrate that the definition of text appraisal as used in this study is not unlikely. However, in parallel, it illustrates the complexity of such constructs.

If anything, this study illustrates how strikingly complex it is to grasp reader's text appreciation, using eye tracking solely. It also identified limitations of eye tracking studies. Nevertheless, promising eye tracking induced models have been introduced that can unveil and even untangle reader's text appreciation. Further research is encouraged to further explore and strengthen the findings of the present study. This research can be considered as a first, significant step toward a better understanding of the relation between reader's eyes and their text appreciation. As such it can serve as a foundation of a next-generation wearable that enables true personalized information filtering, access, and retrieval.

References

Albus, J.E., Anderson, R.H., Brayer, J.M., DeMori, R., Feng, H.Y., Horowitz, S.L., Moayer, B., Pavlidis, T., Stallings, W.W., Swain, P.H., and Vamos, T., (2012). *Syntactic pattern recognition. applications* (Vol. 14). Berlin, GER: Springer Science & Business Media.

Alpaydin, E. (2014). *Introduction to machine learning*. Cambridge, MA, USA: The MIT press.

Balatsoukas, P., and Ruthven, I. (2012). An Eye Tracking Approach to the Analysis of Relevance Judgments on the Web: The Case of Google Search Engine. *Journal of the American Society for Information Science and Technology*, *63*(9), 1728-1746.

Bai, X., Yan, G., Liversedge, S. P., Zang, C., and Rayner, K. (2008). Reading spaced and unspaced s*and Performance*, *34*(5), 1277.

Barua, S., Islam, M. M., and Murase, K. (2011). A novel synthetic minority oversampling technique for imbalanced data set learning. In *Neural Information Processing* (pp. 735-744). Heidelberg, GER: Springer Berlin Heidelberg.

Bi, W., and Kwok, J. T. (2014). Multilabel Classification with Label Correlations and Missing Labels. In *Proceeding of AAAI Conference on Artificial Intelligence* (pp. 1680-1686). Palo Alto, CA, USA: AAAI Press.

Bishop, C. (2006). *Pattern Recognition and Machine Learning*, (vol. 4). New York, NY, USA: Springer New York.

Blignaut, P. (2009). Fixation identification: The optimum threshold for a dispersion algorithm. *Attention, Perception, & Psychophysics*, *71*(4), 881-895.

Blum, A.L., and Langley, P. (1997). Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence*, *97*(1-2), 245-271.

Bridle, J. S. (1990). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neurocomputing* (pp. 227-236). Heidelberg, GER: Springer Berlin Heidelberg.

Bulling, A., Ward, J. A., Gellersen, H., and Tröster, G. (2009). Eye movement analysis for activity recognition. In *Proceedings of the 11th international conference on Ubiquitous computing* (pp. 41-50). New York, NY, USA: ACM.

Bulling, A., and Gellersen, H. (2010). Toward mobile eye-based human-computer interaction. *Pervasive Computing, IEEE, 9*(4), 8-12.

Cole, M. J., Gwizdka, J., Bierig, R., Belkin, N. J., Liu, J., Liu, C., and Zhang, X. (2010). Linking search tasks with low-level eye movement patterns. In *Proceedings of the 28th Annual European Conference on Cognitive Ergonomics* (pp. 109-116). New York, NY, USA: ACM.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*(1), 321-357.

Demberg, V., and Keller, F. (2008). Data from Eye-Tracking Corpora as Evidence for Theories of Syntactic Processing Complexity. *Cognition*, *109*(2), 193-210.

Deterding, S., Sicart, M., Nacke, L., O'Hara, K., and Dixon, D. (2011). Gamification. using game-design elements in non-gaming contexts. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (pp. 2425-2428). New York, NY, USA: ACM.

Doherty, S., O'Brien, S., and Carl, M. (2010). Eye tracking as an MT evaluation technique. *Machine translation*, *24*(1), 1-13.

Duchowski, A. (2007). *Eye tracking methodology: Theory and practice* (Vol. 373). Berlin, GER: Springer Science & Business Media.

Franceschet, M. (2011). PageRank: Standing on the shoulders of giants. *Communications of the ACM*, *54*(6), 92-101.

Garcin, F., Faltings, B., Donatsch, O., Alazzawi, A., Bruttin, C., and Huber, A. (2014). Offline and online evaluation of news recommender systems at swissinfo.ch. In *Proceedings of the 8th ACM Conference on Recommender systems* (pp. 169-176). New York, NY, USA: ACM.

He, H., Bai, Y., Garcia, E. A., and Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)* (pp. 1322-1328). Hong Kong, CHINA: IEEE.

He, H., Chen, S., Man, H., Desai, S., and Quoraishee, S. (2010). Imbalanced learning for pattern recognition: an empirical study. In *Security+ Defence* (pp. 78330T-78330T). Bellingham, WA, USA: International Society for Optics and Photonics.

Hidi, S., and Renninger, K.A. (2006). The four-phase model of interest development. *Educational Psychologist, 41*(2), 111–127.

Hyönä, J., Tommola, J., and Alaja, A. M. (1995). Pupil dilation as a measure of processing load in simultaneous interpretation and other language tasks. *The Quarterly Journal of Experimental Psychology*, *48*(3), 598-612.

Inhoff, A. W., and Rayner, K. (1986). Parafoveal word processing during eye fixations in reading: Effects of word frequency, *Perception and Psychophysics, 40*(6), 431-439.

Jacob, R. J., and Karn, K. S. (2003). Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. *Mind*, *2*(3), 4.

Jain, A. K., Duin, R. P. W., and Mao, J. (2000). Statistical pattern recognition: A review. *IEEE Transactions on pattern analysis and machine intelligence*, *22*(1), 4-37.

Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern recognition letters*, *31*(8), 651-666.

Jones, M. W., Obregón, M., Kelly, M. L., and Branigan, H. P. (2008). Elucidating the component processes involved in dyslexic and non-dyslexic reading fluency: An eye-tracking study. *Cognition*, *109*(3), 389-407.

Kassner, M., Patera, W., and Bulling, A. (2014). Pupil: an open source platform for pervasive eye tracking and mobile gaze-based interaction. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing: Adjunct publication* (pp. 1151-1160). New York, NY, USA: ACM.

Kleinnijenhuis, J. (1991). Newspaper complexity and the knowledge gap. *European journal of communication*, *6*(4), 499-522.

Kliegl R., and Engbert R. (2005). Fixation durations before word skipping in reading. *Psychonomic Bulletin & Review*, *12*(1), 132-138.

Kruse, R., Borgelt, C., Klawonn, F., Moewes, C., Steinbrecher, M., and Held, P. (2013). Multi-layer perceptrons. In *Computational Intelligence* (pp. 47-81). London, UK: Springer London.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444.

Liaw, A., and Wiener, M. (2002). Classification and regression by randomForest. *R news, 2*(3), 18-22.

Liversedge, S. P., and Findlay, J. M. (2000). Saccadic eye movements and cognition. *Trends in cognitive sciences*, *4*(1), 6-14.

Liversedge, S., Gilchrist, I., and Everling, S. (2011). *The Oxford handbook of eye movements*. Oxford University Press.

Mason, L., Pluchino, P., Tornatora, M. C., and Ariasi, N. (2013). An eye-tracking study of learning from science text with concrete and abstract illustrations. *The Journal of Experimental Education*, *81*(3), 356-384.

McDaniel, M. A., Waddill, P. J., Finstad, K., and Bourg, T. (2000). The effects of text-based interest on attention and recall. *Journal of Educational Psychology*, *92*(3), 492.

Mor, N., Riva, O., Nath, S., and Kubiatowicz, J. (2015). Bloom Cookies: Web Search Personalization without User Tracking. In *NDSS.*

Meyer, D., Leisch, F., and Hornik, K. (2003). The support vector machine under test. *Neurocomputing*, *55*(1–2), 169–186.

Ozuysal, M, Calonder, M, Lepetit, V., and Fua P. (2010). Fast Keypoint Recognition using Random Ferns, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3), 448-461.

Pariser, E. (2011). *The filter bubble: What the Internet is hiding from you*. London, UK: Penguin UK.

Partala, T., and Surakka, V. (2003). Pupil size variation as an indication of affective processing. *International journal of human-computer studies*, *59*(1), 185-198.

Rayner, K., Chace, K. C., Slattery, T. J., and Ashby, J. (2006). Eye Movements as Reflections of Comprehension Processes in Reading. *Scientific Studies of Reading*, *10*(3), 241-255.

Reichle, E. D., Reineberg, A. E., and Schooler, J. W. (2010). Eye movements during mindless reading. *Psychological Science*, *21*(9), 1300-1310.

Romero, M., Usart, M., and Ott, M. (2015). Can Serious Games Contribute to Developing and Sustaining 21st Century Skills? *Games and Culture*, *10*(2), 148-177.

Segal, M. R. (2004). *Machine learning benchmarks and random forest regression*. San Francisco, CA, USA: Center for Bioinformatics & Molecular Biostatistics.

Silvia, P.J. (2006). *Exploring the psychology of interest*. New York, NY, USA: Oxford University Press.

Silvia, P.J. (2008a). Appraisal components and emotion traits: Examining the appraisal basis of trait curiosity. *Cognition & Emotion, 22*(1), 94–113.

Smith, M. E., Gevins, A., Brown, H., Karnik, A., and Du, R. (2001). Monitoring task loading with multivariate EEG measures during complex forms of human-computer interaction. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *43*(3), 366-380.

SMI (SensoMotoric Instruments). BeGazeTM software manual, version 2.4, document version 1.02.10.

Tang, Y., Zhang, Y. Q., Chawla, N. V., and Krasser, S. (2009). SVMs modeling for highly imbalanced classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39*(1), 281-288.

Tsai, Y. F., Viirre, E., Strychacz, C., Chase, B., and Jung, T. P. (2007). Task performance and eye activity: predicting behavior relating to cognitive workload. *Aviation, space, and environmental medicine*, *78*(Supplement 1), B176-B185.

Tsai, M. J., Hou, H. T., Lai, M. L., Liu, W. Y., & Yang, F. Y. (2012). Visual attention for solving multiple-choice science problem: An eye-tracking analysis. *Computers & Education*, *58*(1), 375-385.

Van den Broek, E. L., Janssen, J. H., and Westerink, J. H. (2009). Guidelines for affective signal processing (ASP): From lab to life. In *Proceedings of the IEEE International Conference on Affective Computing & Intelligent Interaction* (pp. 704-709). Washington, D.C., USA: IEEE Computer Society.

Van der Sluis, F., Van den Broek, E.L., Glassey, R.J., Van Dijk, E.M.A.G., and De Jong, F.M.G. (2014). When complexity becomes interesting. *Journal of the American Society for Information Science and Technology*, *65*(7), 1478-1500.

Van der Sluis, F., Glassey, R. J., and van den Broek, E. L. (2012). Making the news interesting: Understanding the relationship between familiarity and interest. In *Proceedings of the 4th Information Interaction in Context Symposium* (pp. 314-317). New York, NY, USA: ACM.

Wiley, J., and Rayner, K. (2000). Effects of titles on the processing of text and lexically ambiguous words: Evidence from eye movements. *Memory Cognition, 28*(6), 1011-1021.

West, J., Salter, A., Vanhaverbeke, W., and Chesbrough, H. (2014). Open innovation: The next decade. *Research Policy*, *43*(5), 805-811.

Wu, L., & Zhang, M. L. (2013). Multi-Label Classification with Unlabeled Data: An Inductive Approach. In *Proceedings of The 5th Asian Conference on Machine Learning ACML* (pp. 197-212). Canberra, ACT, AUS: ACML2013.

Appendices

Appendix A

Initially, before preprocessing the data set, we did a statistical analysis. A within-subjects Multivariate ANalysis Of Variance (MANOVA) was conducted to test for an effect, within the 18 texts, of the textual complexity and the average scores of the appraised complexity, comprehensibility, and interest and the eye features.
There was a statistically difference in eye movements and the appraisals based on textual complexity, F (52, 58) = 2.22, p = .002; *Wilk's* Λ = 0.111, partial η^2 = .666.

There was a statistically difference in eye movements and appraisals based on the different 18 texts, F (442,6256) = 1.283, p < .0005; *Wilk's* Λ = 0.290, partial η^2 = 0.07.

From the MANOVA analysis above, the not positive, because there is not significant difference in variance, therefore we will do some additional preprocessing and use different algorithms to improve our model.

Appendix B

Table 1 below, shows the classification performance of the test set, for eight(binary) classes, regarding all three dimensions. The diagonal of the table (grey cells), displays the correctly classified instances.

Table 1: Prediction rates for 8 (binary) classes, with each triplet, (blue, red , green) representing the dimensions' complexity, comprehensibility and interes t.

	Predictions							
Targets	0,0,0	0,1,0	0,0,1	0,1,1	1,1,0	1,1,1	1,0,1	1,0,0
0,0,0	2	8	13	8	2	7	1	
0,1,0								
0,0,1	1	2	5	1	2	3		
0,1,1		1						
1,1,0		2	2			3		
1,1,1						2		
1,0,1		2	3			1		
1,0,0		3	6		2	7		

Table 2 below, shows the classification performance of the test set, for four (binary) classes, regarding complexity and comprehensibility. The diagonal of the table (grey cells), displays the correctly classified instances.

Table 2: Prediction rates for 4 (binary) classes, with each doublet, (blue and red) representing the dimensions' complexity, comprehensibility.

	Predictions			
Targets	0,0	0,1	1,0	1,1
0,0	19	17	2	17
0,1		1		
1,0	9	5		10
1,1	3	1		5

The table 3 below, shows the classification performance of the test set, for four (binary) classes, regarding complexity and interest. The diagonal of the table (grey cells), displays the correctly classified instances.

Table 3: Prediction rates for 8 (binary) classes, with each doublet, (blue, red , green) representing the dimensions' complexity, complexity and interest.

	Predictions			
Targets	0,0	0,1	1,0	1,1
0,0	8	21	2	10
0,1	4	5	2	4
1,0	4	8	3	10

1,1	2	3		3

Table 4 below, shows the classification performance of the test set, for four (binary) classes, regarding comprehensibility and interest. The diagonal of the table (grey cells), displays the correctly classified instances.

Table 4: Prediction rates for 4 (binary) classes, with each doublet, (red and green) representing the dimensions' comprehensibility and interest.

	Predictions			
Targets	0,0	0,1	1,0	1,1
0,0	1	21	15	22
0,1	1	8	6	5
1,0	1	2	1	3
1,1	1		1	

Appendix C

Multi-label Balanced (SMOTE-data)

The following tables (5-8), show results regarding multi-label classification after the use of the SMOTE algorithm.

$2^2 = 4$ classes (complexity - comprehensibility)

Table 5 below shows the predictions for the combination of the labels complexity and comprehensibility. Where 1: Simple- Comprehensible, 2: Simple- Incomprehensible, 3: Complex- Comprehensible, 4: Complex- Incomprehensible.

Table 5: Prediction rates for 4 classes, representing the dimensions' complexity and comprehensibility.

	Predictions			
Targets	***1***	***2***	***3***	***4***
1	46	4	1	3
2	9	1	0	0
3	6	1	16	9
4	2	0	10	20

$2^2 = 4$ classes (complexity - interest)

Table 6 below shows the predictions for the combination of the labels complexity and comprehensibility. Where 1: Simple- Interesting, 2: Simple- Uninteresting, 3: Complex- Interesting, 4: Complex- Uninteresting.

Table 6: Prediction rates for 4 (binary) classes, representing the dimensions' complexity and interest.

	Predictions			
Targets	***1***	***2***	***3***	***4***
1	21	14	2	4
2	13	6	2	2
3	7	1	19	16

4	4	1	14	3

2^2 = 4 classes (comprehensibility - interest)

Table 7 below shows the predictions for the combination of the labels complexity and comprehensibility. Where 1: Comprehensible - Interesting, 2: Comprehensible - Uninteresting, 3: Incomprehensible - Interesting, 4: Incomprehensible - Uninteresting.

Table 7: Prediction rates for 4 classes, representing the dimensions' comprehensibility and interest.

	Predictions			
Targets	***1***	***2***	***3***	***4***
1	25	13	4	1
2	13	5	1	2
3	3	3	10	21
4	2	1	8	16

2^3 = 8 classes (complexity - comprehensibility - interest)

Simple-complex, comprehensibility-incomprehensibility, interest-uninteresting

Table 7 below shows the predictions for the combination of the labels complexity and comprehensibility. Where 1: Simple-Comprehensible-Interesting, 2: Simple-Comprehensible-Uninteresting, 3: Simple-Incomprehensible - Interesting, 4: Simple-Incomprehensible-Uninteresting, 5: Complex -Comprehensible - Interesting, 6: Complex-Comprehensible-Uninteresting, 7: Complex-Incomprehensible-Interesting, 8: Complex-Incomprehensible-Uninteresting.

Table 8: Prediction rates for 8 (binary) classes, representing the dimensions' complexity and interest.

	Predictions (40/105)							
Targets	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***
1	15	7	6	1	0	1	0	2
2	9	10	3	0	1	1	0	2
3	2	3	1	0	0	0	0	1
4	1	3	0	1	0	1	0	0
5	2	2	0	0	2	5	2	6
6	0	2	0	1	1	2	0	4
7	1	2	0	0	1	1	2	3
8	2	0	0	0	1	4	1	10