

Article

Continuous Prediction of Web User Visual Attention on Short Span Windows Based on Gaze Data Analytics

Francisco Diaz-Guerra ^{1,†,‡}  and Angel Jimenez-Molina ^{1,2,*,†,§} ¹ Department of Industrial Engineering, University of Chile, Santiago 8370456, Chile² Engineering Complex Systems Institute, Santiago 8370398, Chile

* Correspondence: angeljim@uchile.cl; Tel.: +56-2-2978-0543

† These authors contributed equally to this work.

‡ Current address: Avenida Beauchef #851, Of. 514, Santiago 8370456, Chile.

§ Current address: Avenida Beauchef #851, Of. 711, Santiago 8370456, Chile.

Abstract: Understanding users' visual attention on websites is paramount to enhance the browsing experience, such as providing emergent information or dynamically adapting Web interfaces. Existing approaches to accomplish these challenges are generally based on the computation of salience maps of static Web interfaces, while websites increasingly become more dynamic and interactive. This paper proposes a method and provides a proof-of-concept to predict user's visual attention on specific regions of a website with dynamic components. This method predicts the regions of a user's visual attention without requiring a constant recording of the current layout of the website, but rather by knowing the structure it presented in a past period. To address this challenge, the concept of visit intention is introduced in this paper, defined as the probability that a user, while browsing, will fixate their gaze on a specific region of the website in the next period. Our approach uses the gaze patterns of a population that browsed a specific website, captured via an eye-tracker device, to aid personalized prediction models built with individual visual kinetics features. We show experimentally that it is possible to conduct such a prediction through multilabel classification models using a small number of users, obtaining an average area under curve of 84.3%, and an average accuracy of 79%. Furthermore, the user's visual kinetics features are consistently selected in every set of a cross-validation evaluation.

Keywords: visual attention prediction; eye-tracker sensor; visual gaze patterns; visual kinetics; human-computer interaction; gaze data analytics



Citation: Diaz-Guerra, F.; Jimenez-Molina, A. Continuous Prediction of Web User Visual Attention on Short Span Windows Based on Gaze Data Analytics. *Sensors* **2023**, *23*, 2294. <https://doi.org/10.3390/s23042294>

Academic Editors: Antonio Fernandez-Caballero, Guangtao Zhai and Enrico Vezzetti

Received: 17 November 2022

Revised: 27 January 2023

Accepted: 13 February 2023

Published: 18 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Understanding the behavior of users' visual attention on websites is an active research area tackled by the fields of computer vision, human-computer interaction, and Web intelligence. By using information gleaned from this understanding, a website can be constructed to deliver information of greater utility and complexity, such as emergent recommendations in regions of a Web interface that capture users' attention, adapt visual stimuli to users' gaze patterns in real time, among other advantages. Traditional methods to study users' visual attention have historically focused on images. However, recently more complex visual stimuli have been considered, such as videos [1–5], virtual reality environments [6,7], egocentric videos [8,9], and websites. Unfortunately, the literature on users' visual attention on websites is generally based on salience maps computed for static Web interfaces [10–13], where the website structure is always known in advance [14]. Therefore, as dynamic and interactive websites become increasingly prevalent, traditional methods of evaluating visual attention, such as those based on static salience maps or pre-determined website structures, are becoming less effective.

It is imperative to gain a deeper understanding of users' visual attention on websites to create an optimal online experience. This requirement can be realized by dynamically

predicting users' visual attention on specific regions of a website. This understanding offers several advantages, such as identifying regions of the website that capture users' attention and making them more prominent; adjusting the layout and design to match users' gaze patterns, or improving the user experience by creating a more intuitive and user-friendly interface. Furthermore, this understanding can aid in optimizing a website's layout to increase user engagement and conversion rates, and addressing usability issues, such as elements on the website that may be causing confusion or frustration.

Users' own browsing actions are the most significant source of difficulty in predicting their visual attention in current dynamic websites. For instance, these actions may yield the display of menus, pop-ups, or banners when clicking certain regions, or movement of the webpage when using the lateral scroll. Given that the interaction with a website generates constant changes in its layout, there exists the need for a prediction that identifies the regions of a user's visual attention without requiring the constant recording of the current layout but knowing the structure it presented in a past period. To address this challenge, this paper introduces the concept of visit intention, defined as the probability that a user, while browsing, will fixate on a specific region of the website in the next period. These regions are known as areas of interest (AOIs) and are defined by specific groupings of the website components.

In this paper, visit intentions to AOIs are obtained through multilabel classifiers that leverage different features extracted from eye-tracking data. The first set of features comprises a user's gaze fixations in different periods over the components of AOIs. Since generally little information per user is available, our approach generates a population-level classification model to improve generalization and reduce the risk of overfitting from training models with few samples. Furthermore, this means that a subset of these features embeds information about the behavior of other users in that of a specific participant. That is, visit intention prediction models integrate, in the first instance, the features calculated from a population-level gaze pattern, allowing a robust training of models of individual visit intentions, increasing their performance. Another set of features comprises the user's visual kinetics, such as gaze position, velocity, and acceleration measurements in the X- and Y-coordinates of the interface. We hypothesize that visual kinetics features enable the prediction models to capture tendencies of the gaze at the beginning of a period, which would inform future eye movement patterns.

This paper thus attempts to answer the following research question:

- RQ: To what extent is it possible to accurately classify a user's visit intention in a certain period in real time to specific AOIs of a website, leveraging population-level general gaze patterns and a user's particular gaze data?

To answer this research question, 51 users participated in an experiment browsing a website with dynamic components in front of an eye-tracking device. This paper shows that by using the population-level gaze patterns of a few users and applying diverse classification methods and feature selection techniques, a user's visit intention to AOIs in a period can be predicted as proposed in RQ (average $AUC = 0.843$, average $ACC = 0.79$). Furthermore, individual users' visual kinetics features are consistently selected in every cross-validation set, confirming the hypothesis.

This paper is organized as follows. Section 2 presents the related work. Section 3 defines key concepts, delivers a formulation of the problem to be addressed, and explains the apparatus used for the experimentation carried out to answer the research question. Section 4 explains the proposed prediction methodology. Section 5 shows an experimental evaluation by applying the proposed methodology to a real dataset of Web users' gaze data. Finally, Section 6 presents the discussions of the work carried out, while the paper is concluded in Section 7.

2. Literature Review

Visual attention is a mechanism that filters relevant areas of a stimulus from the rest of redundant information. Knowledge of a user's attention can be useful in applications

such as coding and data transmission, improvement in recommendation and information delivery systems, and performance evaluation in different visual stimuli, among others.

In the literature, attention is categorized into two functions: bottom-up attention and top-down attention. The first corresponds to the selection of zones of a visual stimulus based on its most salient inherent characteristics in relation to the background. This function of attention operates with input in a crude, involuntary way. The second integrates knowledge of the visual scene, goals or objectives of the user with respect to the stimulus. Moreover, it implements longer-term cognitive strategies on the part of the person [15–17].

Studies focused on top-down attention models are less prominent in the literature than their bottom-up counterparts, since their functionality is influenced by factors external to the stimulus. In addition, calculated responses vary from model to model, i.e., from task to task, making it difficult to generalize a stimulus model. On the other hand, different models of bottom-up visual attention have been proposed. In this work, bottom-up models are studied.

2.1. Attention Models in Static Stimulus

The focus of classic studies on visual attention corresponds to static visual stimuli, that is, stimuli that do not present an alteration during the time period of exposure to a user. In this field, there are multiple studies on visual attention when making “free viewing” on images. Of these, there are mainly three approaches for the study of visual attention: saliency map generation, scanpaths models and the saccade models.

Saliency maps are topographic representations of the visual prominence of a stimulus. That is, they make it easier to understand areas of greater or lesser importance in the selection of points of attention. There are multiple studies in which saliency maps are applied to images. In [18], transformations are made from features present in different natural images to maps of salience, searching for a smaller number of characteristics to use. In [19], the images are convolved with a template and postprocessed to deliver the result. Association methods have also been utilized. In [20], a method called saliency transfer is presented, where a support set of best matches from a large database of saliency annotated images is retrieved, and transitional saliency scores are assigned by warping the support set annotations onto the input image according to computed dense correspondences. In other research, deep learning approaches has been adopted. In [21], a deep convolutional neural network is proposed that is capable of predicting fixations and segmenting objects in different images. In [22], an approach based on convolutional neural networks is employed which incorporates multi-level saliency predictions within a single network, capturing hierarchical saliency information from deep, coarse layers with global saliency information to shallow, fine layers with local saliency response. Other deep learning approaches applied for salient object detection have been reviewed in the literature [23,24] or for co-salient object (group of objects that occur together in an image) [25]. In general, there is a large database of images and different models developed in the state-of-the art approaches [26], as well as studies about their performance metrics [27,28].

Scanpaths are attempts to predict sequences of users’ fixations within a stimulus. While many papers study scanpath generation and saliency map extraction independently, research focusing on the fusion of these approaches is limited. However, some works focus on the identification of scanpaths by using sampling methods on top of saliency maps [10,11]. Other studies have utilized fixation data for the calculation of saliency maps, such as the Attentive Saliency Network (ASNet), developed in [29].

Saccadic models represent another option. In [30], predictions of exploration routes are made in natural images. In [31], the saccadic model considers age ranges of users. This model integrates the interaction between the way in which users observe the visual information and the mental state of each user.

It has been demonstrated that static models present good results; however, in this work, we seek to study the visual attention of users in dynamic environments, which in general are more complex since they consider time as a new variable.

2.2. Attention Models in Dynamic Stimulus

There are models developed for bottom-up attention on dynamic stimuli, that is, environments that present movement or that change their structure when interacting in a certain way with the person. For example, a video presents alterations (frame-to-frame movement), but it does not depend on the interaction. On the other hand, a web page presents dynamism in its structure or content depending on the user's interaction.

In [1], predictions of saliency maps are made in videos, frame-by-frame. For this purpose, recurrent network methods with mixed density are used. In this study, the visual stimulus is known over time, thus allowing it to be broken down into a series of static images. However, the advantage of this approach, in which the saliency map for each frame of the video is generated by the information from the previous frames, is that information is integrated from the user interaction. Other studies have developed the evaluation of attention in videos, such as [32] which compares a model without video annotations, with annotations of foreground objects, and other biologically based annotations, or [33] where a deep learning approach was adopted.

Saliency maps have also been applied to stimuli in web environments. In [34], a saliency map is calculated at the task level. However, most studies have tended to focus on static calculations [10–13]. That is, they do not examine the temporal component of the user's interaction with the website. Finally, there is a dynamic visual attention study [14] in which the site is assumed to be known at all times, and at the same time, it is assumed to know the duration of each subtask conducted by the user.

Although these methods correspond to a dynamic stimulus and contemplate information of user interaction, they generally focus on knowing the stimulus at all times, either by its nature independent of the user or by predicting the duration of subtasks (predicting the structure of the stimulus in certain periods of time). Our approach seeks to generate predictions directly from visual data, without the need for predicting changes in the stimulus or its structure control at all times, seeking to accelerate the processing times of the model in execution.

2.3. Attention Models in Egocentric Vision

Egocentric vision is a recently introduced area of study. It details the analysis of attention in videos captured with wearable cameras on a user such that the visual fields during the execution of different tasks can be determined. In general, cameras are mounted on the head or chest of the user, and in addition to recording the environment in which it is carried out, the eye activity is recorded by an eye-tracker.

A recent study on gaze prediction in these environments is [8]. Here, the authors look for the generation of saliency maps in future frames through frame prediction, a similar approach to the one used in [9]. Although these environments are dynamic and depend on user interaction, the possibilities for interaction are so broad that traditional bottom-up models are insufficient for visual prediction [35]. Other models developed do not achieve generalization because they have specific cases for each task [36,37].

As in the case of the dynamic environments described above, the models consider the prediction of a visual stimulus structure or future sub-tasks for the estimation of visual attention. This is because the stimuli in this area are too complex, escaping our framework. In particular, what we are looking for is to study a dynamic stimulus that moves within well-defined structures.

To the best of our knowledge, this is the first work to study attention transition between zones of a dynamic visual stimulus that changes according to the user's behavior, directly from user's visual information, without predicting changes in the stimulus structure.

3. Material and Methods

3.1. Problem Definition

This paper studies the visual attention of a user on a dynamic stimulus, within a bottom-up attention environment, that is, where attention is guided by inherent properties

of the stimulus, and not by factors internal to the user such as previous knowledge of the stimulus or specific tasks to be carried out (top-down attention). To study the above topic, a two-dimensional stimulus is created and is exposed to different users, and each interaction is recorded with eye-tracker instrumentation.

User visual behavior is represented by the recording of gaze data. In this way, each point of gaze x_i is represented by a vector of three components $x_i = [x_i, y_i, T_i]$, where the first two correspond to the horizontal and vertical coordinates of each record over a two-dimensional stimulus, and the third corresponds to the time in interaction. In the following, a signal is referred to as the visual record of a complete interaction.

The eye-tracker records a user's visual signals. This gives us information of gaze, being able to identify fixations and saccades. A fixation is the state in which the eyes remain static, with some micro-movements, for a period of time (100–300 ms), for example, over a word when reading. They are identified as sets of gaze data, and their position is considered as the place where the user keeps their attention. A saccade corresponds to ballistic movements that have a duration between 30 and 80 ms, made between one fixation and another. This is not necessarily done in a straight line.

The development of this work is based on the segmentation of stimulus into AOIs, that is, regions where the researcher is looking for user behavior. The space between the AOI's is called whitespace, which is not necessarily free of stimulus elements.

For the AOI definition, there is an approach where the stimulus is divided into equal size regions, called segmentation in grid. On the other hand, AOI's can be defined in a semantic way, such that common meanings are used between zones assigned to the same AOI. There may be differences in shape and/or size with respect to other AOI's. We use the latter approach in this work given that it is expected to obtain information about the visual attention of the user in certain areas with a dynamic nature.

Therefore, an important factor in the problem is that for each AOI definition, it is necessary to consider grouping the zones of the stimulus that present the same information for the user. For example, grouping two news items within a newspaper allows the AOI to be characterized as news information. However, grouping an advertisement with one news item does not allow the AOI to be given a characteristic meaning.

This paper considers the visual attention of users to the different AOI's defined within a dynamic stimulus. For this, the concept of visit intention is defined, by which it is evaluated in advance such that the AOI visual stops by a user will be verified. In this way, given the visual stimulus, the AOI's $\{A_j | j = 1..n\}$ and a period of time $v^{(t)} = (T_o^{(t)}, T_f^{(t)})$ are given, where the values of the tuple correspond to the time limits of the period, which are indexed by $\{t = 1, 2, \dots\}$, and the visit intention $Iv(A_j, v^{(t)}) \in \{0, 1\}$ is defined as a binary variable that indicates whether or not there is a user decision to visit A_j in that period. The decision to visit an AOI will be measured as the existence of visual fixations of the user within that stimulus zone. The visit intention for n AOI's in a period time is a vector $IV(v^{(t)}) = [Iv(1, v^{(t)}), \dots, Iv(n, v^{(t)})]$.

It is assumed that the vector $IV(v^{(t)})$ is characterized at the beginning of the time period $v^{(t)}$, although in reality, the process of choosing it can take place during all the time before the beginning of the window $v^{(t)}$. What we are looking for in the following sections is the prediction of this vector for different time windows $v^{(t)}$. Time window and period of time are used interchangeably throughout the paper, given the information of the user's behavior in the past time (periods $v^{(s)}$, $s = 1, \dots, t - 1$).

What follows from this section describes the experiment developed to study the bottom-up visual attention of different users on a dynamic visual stimulus based on the estimation of the values of visit intention in different time periods.

3.2. Participants

The experiment is conducted with 51 participants based on a convenience sample. Of these participants, 34 are men and 17 are women, between the ages of 19–35. A large percentage of the participants (88%) are students, and the occupations of the rest include

a research assistant, a participant with a bachelor's degree in Hispanic literature, and four engineers.

The participants signed an informed consent form stating that they permit the use of the data collected and meet the following requirements:

- They are healthy individuals who do not present with diseases that could harm the results of the experiment, they are not under pharmacological treatment and they do not have a history of neurological or psychiatric diseases.
- They do not present with the use of medications or drugs within the past 24 h.
- They present with good vision or corrected vision.

3.3. Apparatus

3.3.1. Instrumentation

The eye-tracker Tobii T120 (Tobii Technology AB, Stockholm, Sweden) is used, which allows tracking of the subject's gaze during the test. This setup corresponds to a screen of 17 inches, with a resolution of 1280×1024 px, where two infrared diodes generate patterns of reflection on the corneas of the user's eyes; the three-dimensional position of each eye is calculated, and therefore, his or her gaze on the screen is determined. At the same time, the pupil sizes of both eyes are recorded over time. The Tobii Studio software allows the adjustment of the user's position during calibration.

The system has a sampling rate of 120 Hz, a typical accuracy of 0.5 Hz, a typical drift of 0.1 degrees, a typical spatial resolution of 0.3 degrees and a typical head movement error of 0.2 degrees.

3.3.2. Web Page

To carry out the study, a dynamic stimulus corresponding to an informative web page is used. This corresponds to new content for the user, as it is content written and edited by the developer, organized in a one page design. To achieve the dynamic character, the static and complete website version cannot be seen entirely on the screen, requiring the use of a scrolling bar by the user. At the same time, dynamic elements within the page, such as drop-down menus or moving elements, are considered.

The web page has different components, that is, the sections of the page that represent an element in its own functions and actions. Web page components can be seen in Figure 1 with different colors, and correspond to the following: "Logo"; "Bar_Menu"; "Title"; "Img" and "New" for each of the seven news items; "Banner", for each of the four advertisements; "Num_Pag" (corresponding to the pages number button); and "Sub_Butt" (corresponding to the user's subscription button). In addition, there are dynamic components, that is, they appear according to the user's interaction with the website. These are the country selection bar, the menu bar presented in Figure 2 and the user registration pop-up window in Figure 3.



Figure 1. Static components of the Web page used in the experiment (content provided in Spanish).



Figure 2. Dynamic component of the Web page used in the experiment: Page drop-down menu (content provided in Spanish).

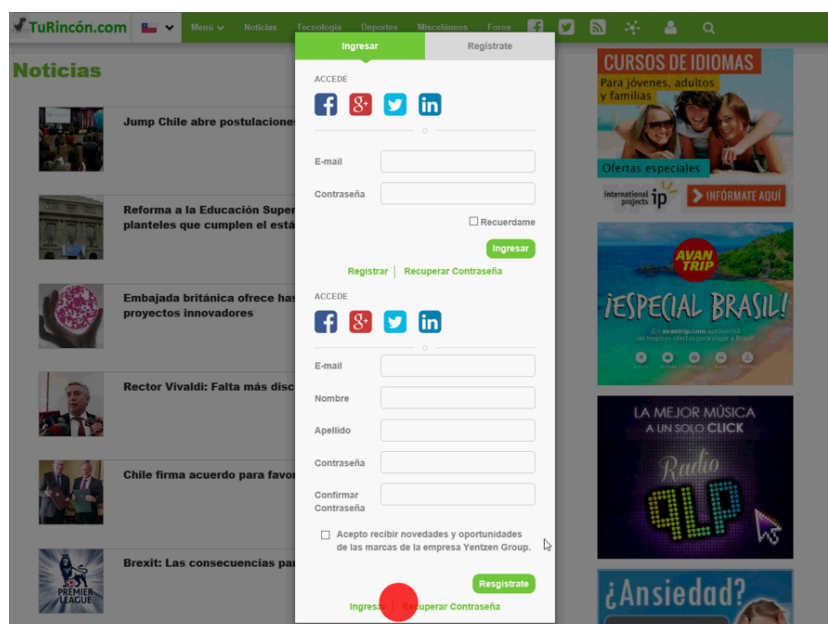


Figure 3. Dynamic component of the Web page used in the experiment: User registration pop-up (content provided in Spanish).

3.4. Task Design

The objective of the experiment is to predict the attention zones of a user's gaze during the exposure of a dynamic stimulus in a bottom-up attention task. To perform this, a fictitious website is chosen, created specifically for the experiment, which is run locally to avoid possible delays in connection to the internet. The web page is detailed in Section 3.3.

A total of three websites are presented to each participant. Each considers the same structure in the elements of the page, but the content varies. Advertisements and news headers appear randomly and without repetition, out of a total of 12 available advertisements and 21 news items. This approach avoids the experience effect of a user with the stimulus and, at the same time, creates a dynamic stimulus with different content for each user. The aim is that users navigate freely on the website and that this, together with the lack of previous experience of the user with the page, allows the study of bottom-up attention.

The content that is chosen, both for news and announcements, is related to student life. Topics include travel, courses, beer, education, music, technology and miscellaneous.

For the best analysis of the data, the control of variables such as luminosity and the user's movement is required, such that an isolated space is necessary to carry out the study. For eye tracking, direct sunlight, which affects the quality of the measurements, should be avoided. In addition, there should be no light from a top source. To avoid these effects of light, the laboratory is isolated from external light with black-out curtains. In addition, measures are taken to avoid all kinds of external interruptions during the experiment.

3.5. Experimental Procedure

During the experiment, only the person in charge of the test and the participant are present in the experimental room. The latter is provided with an explanation of the experiment and asked to read and sign the informed consent form.

The person sits in front of the eye-tracker screen in a manner similar to navigating on a desktop computer, allowing the use of a mouse. The participant is asked to maintain a fixed posture, without moving their head, to avoid data loss.

Each person is prompted to freely browse the web page for as long as they want and to indicate when to finish.

Prior to the tests, each user is subjected to a relaxation stage consisting of the visualization of three videos of four minutes each, consisting of picturesque landscapes with background instrumental music. The second part of this stage asks the participant to take a

deep breath for one minute with closed eyes and soft instrumental music in the background. This stage aims to eliminate the Hawthorne effect (behavior modification as a consequence of knowing they are being studied) and physiological effects. In addition, this stage allows the baseline to be acquired for later treatment of the signals. Then, the page is presented randomly to each participant.

3.6. Descriptive Analysis

A total of 153 signals are extracted from 51 volunteers for the signals of the gaze behavior and event registration (click, display of dynamic elements, scroll, etc.). Fixations with duration less than 100 ms are filtered from the model. Table 1 shows descriptive statistics of the acquired signals.

Table 1. Descriptive statistics of the acquired signals.

Number of records	153
Recording time (s)	$t_{min} = 16.5 - t_{max} = 399$ $\bar{t} = 78.8 - \sigma_t = 51.9$
Number of saccades	$s_{min} = 62 - s_{max} = 1129$ $\bar{s} = 262 - \sigma_s = 165$
Fixation per record	$F_{min} = 62 - F_{max} = 1130$ $\bar{F} = 262 - \sigma_F = 164.6$
Fixing duration per record (s)	$F_{min} = 0.008 - F_{max} = 3.273$ $\bar{F} = 0.22$

4. Prediction Methodology

Figure 4 shows the methodology for the construction of predictive models of visit intention in limited periods of time. First, components of the web page are established, and then, AOI's are defined. Afterwards, signal records are divided into time windows of a specific length by solving a linear optimization problem. The time windows are then optimally assigned to training and test sets for cross-validation. Labels for each time window are created, and an oversampling balance method is used for each train-test set to avoid overtraining. For each time window, it is possible to create a set of features used for classification. Different methods are tested, and finally, the results are evaluated.

4.1. AOI Definition Criteria

Given the approach of the problem, each defined AOI must have a characteristic meaning; therefore, components of similar semantic meaning were grouped into larger areas. At the same time, the final size of each defined area is considered so that it can be fully visible on the screen by scrolling the page in the corresponding dimensions.

In this way, the web page is separated into five semantic areas shown in Figure 5. A sixth AOI is defined, corresponding to the registration menu that is displayed when a user clicks on the button intended for this functionality, located in the upper navigation bar (see Figure 3).

AOI 1 is dynamic, as the menu bar moves together with the screen movements given by the scroll. The AOI's 2 and 3 are groupings of outstanding news, and the division into three and four news groups is chosen since the resolution of the screen allows these AOI's to be completely displayed when performing a lateral scroll. Following the same logic, AOI's 4 and 5 are generated corresponding to advertising banners within the web page.

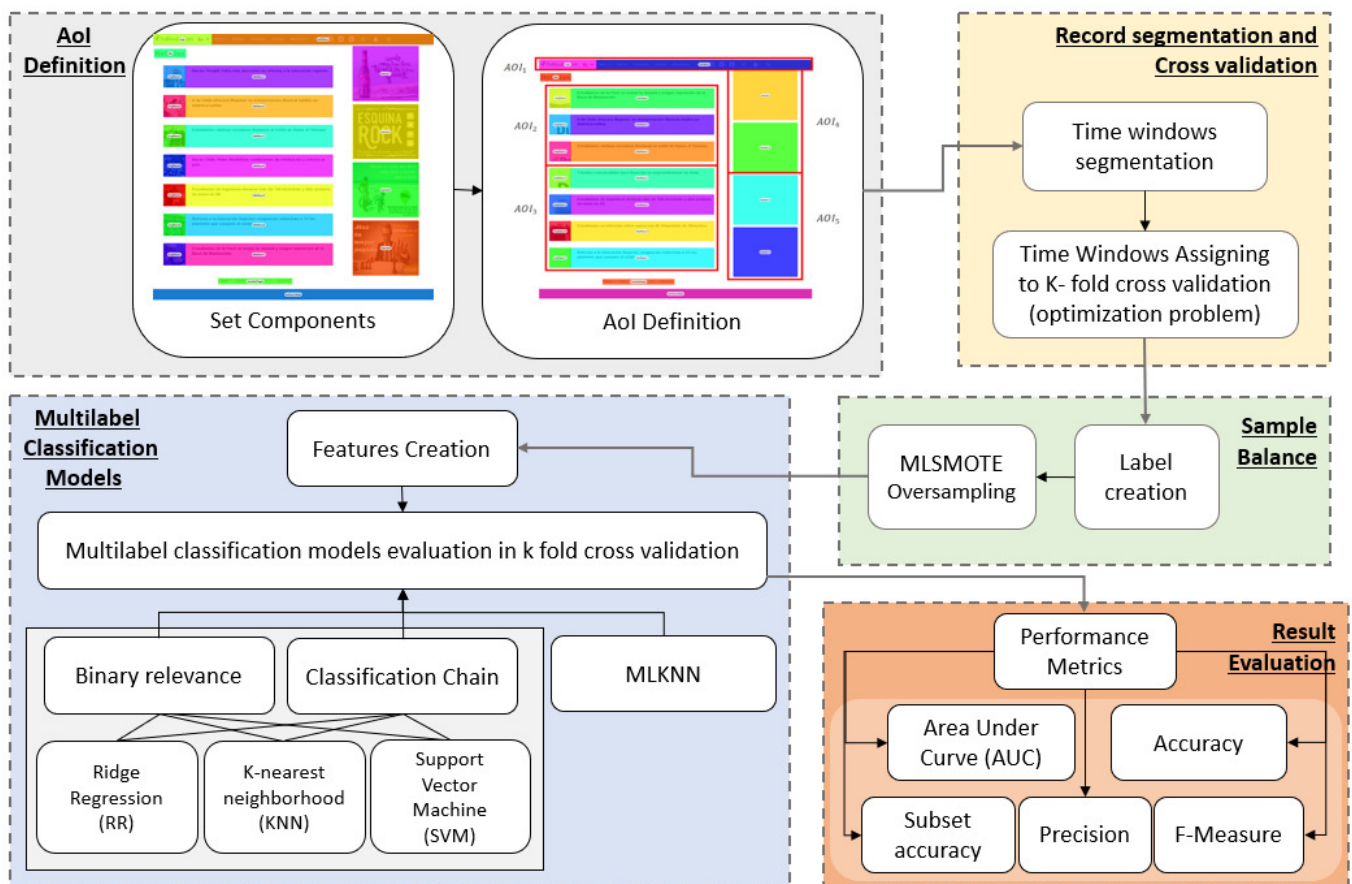


Figure 4. Prediction method. The process consists of five stages that are developed sequentially. Firstly, the areas of interest are defined as sets of components within the website. Subsequently, the signals captured experimentally are segmented into constant length time windows, which are assigned to each of the sets of the k-fold cross validation. As the target variable for each window, a binary label is generated that is one if the area of interest is visited in that temporal window or zero if not, and subsequently MLSMOTE techniques are used for the balance between the different categories. Later, for each time window, a set of features is associated with which the multilabel classification models with k-fold cross validation of MLKNN, Ridge Regression, K-nearest neighborhood (KNN) and support vector machine (SVM) are trained, where the latter three models are trained under the binary relevance and classification chain approaches. Finally, the metrics shown in the “Result evaluation” box are calculated to choose the best model.

AOI 6 has a dynamic nature and is activated by user interaction within the site. When activated, the ocular fixations for the rest of the AOIs located below are deactivated.

Some components of the page are not included in any AOI, as is the case for the subscription bar, the number of pages and the title of news; these are considered to be whitespace.

Thus, each defined AOI meets the conditions of the problem, where AOI 1 acquires the meaning of the navigation menu, areas 2 and 3 refer to news, areas 4 and 5 to advertising and AOI 6 to the user registration menu.

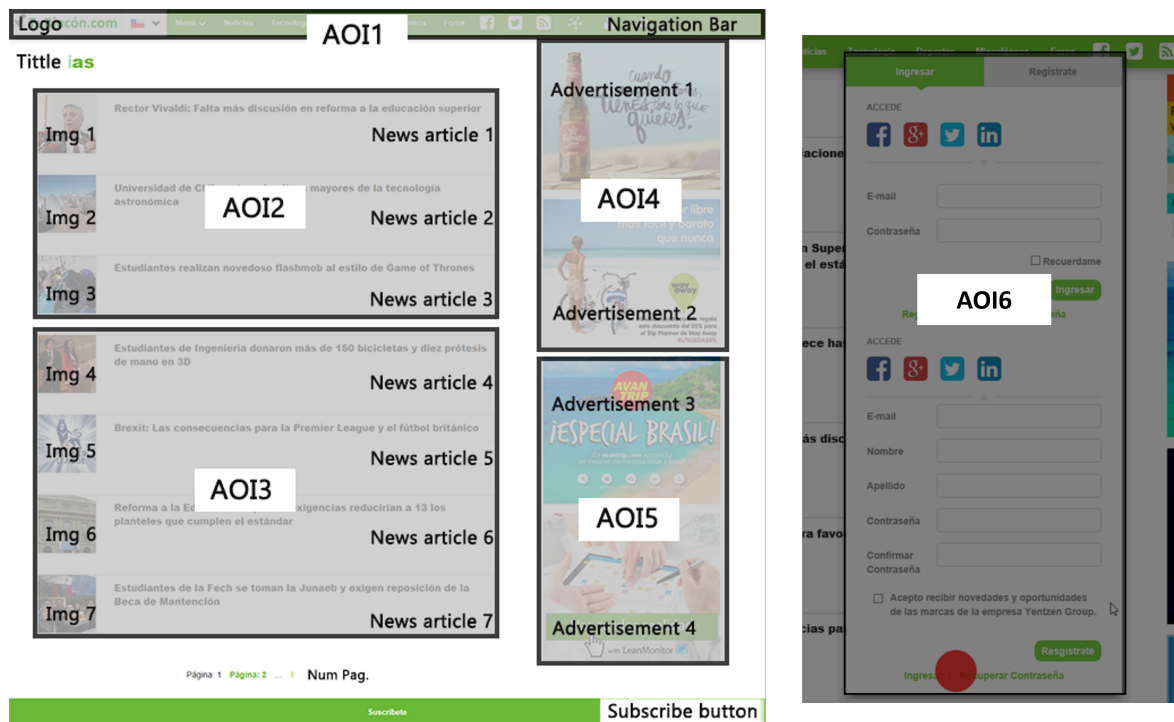


Figure 5. AOIs for the web page. AO1 comprises the page logo and the navigation bar at the top of the web page. AO12, located at the center of the web page, is composed of the titles of three news articles and their respective images (Img 1, Img 2, and Img 3). AO13 is composed of the titles of four news articles and their images (Img 1, Img 2, Img 3, and Img 4). AOIs 4 and 5 consists of two advertisements each one, located at the right-hand side of the web page. AO16 corresponds to the user registration pop-up, which is a dynamic component.

4.2. Time Window Segmentation and Cross-Validation

The models developed with population-level features simulate the continuous acquisition of gaze data over time. To evaluate this acquisition, the gaze signal is segmented into time intervals called windows. These windows have a long τ , given as a parameter. Thus, a long signal T will have $t = 1, \dots, \lfloor T/\tau \rfloor$ time windows $v^{(t)}$.

To achieve external validity of the prediction models, it is necessary to generate training and test datasets that maintain independence from each other. To achieve this generalization, it is considered that both sets cannot share information from the same user. Thus, using the k-fold cross-validation method, a partition per user is performed. In turn, to select which users belong to each of the k-folds, an optimal allocation is performed that balances the number of windows in each fold. Since each user's total browsing time is different, the number of users in each partition varies.

The assignment is made with a mixed integer linear programming model given by Equations (1a) to (1f). Sets $U = 1, \dots, N$ and $KF = 1, \dots, K$ represent the number of users and fold to be used in cross validation, respectively. Variables $\{y_{u,k} | u \in U; k \in KF\}$ are defined with the value one if the user u is assigned to fold k ; otherwise, they are zero. The variables $\{x_k | k \in KF\}$ represent the sum of windows in fold k , given the assignment of users. Thus, each user is assigned to a fold, minimizing the difference between the total sum of windows in each fold k and the average number of windows in all folds. In the best case, all folds will have a number of windows equal to the average \bar{x} .

The restriction (1b) indicates that each user can be assigned to only one of the k-folds. The restriction (1c) defines the variables x_i , each representing the weighted sum between the assigned users and the total number of windows it has in the interactions (P_u). The restriction (1d) defines the arithmetic mean of the number of windows in the folds.

$$\text{minimize } \sum_{k=1}^K |x_k - \bar{x}| \quad (1a)$$

$$\text{subject to } \sum_{k=1}^K y_{u,k} = 1, \quad u \in U, \quad (1b)$$

$$x_k = \sum_{u=1}^N y_{u,k} * P_u, \quad k \in KF, \quad (1c)$$

$$\bar{x} = \frac{\sum_{k=1}^K x_k}{K}, \quad (1d)$$

$$x_k \geq 0, \quad k \in KF, \quad (1e)$$

$$y_{u,k} \in \{0, 1\}, \quad u \in U; k \in KF \quad (1f)$$

To solve (1a–1f), we add new variables $U_i, i = 1, \dots, K$ that represent $|x_i - \bar{x}|$. The target function is changed, and constraints (2b) and (2c) are added. This is how problem (2a–2e) is defined.

$$\text{minimize } \sum_{k=1}^K U_k \quad (2a)$$

$$\text{subject to } U_k \geq x_k - \bar{x}, \quad k \in KF, \quad (2b)$$

$$U_k \geq -(x_k - \bar{x}), \quad k \in KF, \quad (2c)$$

$$U_k \in \mathbb{R}, \quad (2d)$$

$$\{(1b), (1c), (1d), (1e), (1f)\} \quad (2e)$$

4.3. Sample Balance

In multilabel classification problems, it is common that label frequencies are not the same, with some that are more frequently active than others. In this case, since labels are defined by combinations of AOIs, vectors will exist with few combinations, as in the cases in which the dynamic areas of interest are activated, such as the pop-up windows, which are carried out to a lesser extent than the activation of the static AOIs.

The MLSMOTE [38] resampling method is used to solve the sample imbalance problem. This method uses the minority class labels as seeds for the generation of new instances. For each minority class label, it searches for the nearest neighbors by obtaining the characteristics of the new samples using interpolation techniques. Therefore, the new instances are synthetic rather than clones of other data.

4.4. Features Extraction for Multilabel Classification

For multilabel classification, a set of independent, population-level features is generated, which are processed from gaze data signals and navigation history. Indexes $j = 1, \dots, n$ correspond to the defined AOI's and $c = 1, \dots, N$ to the components present in the web page.

The features related to the ocular fixations carried out in each component and AOI are extracted, which are defined in Table 2. These features attempt to capture attention in certain areas given the history of previously visited areas.

Table 2. Predictive ocular fixation model features.

Feature	Definition
$\{AOI_tslv\}_j$	The “time since last visit” features for each AOI are calculated as the difference between the start time of the current window and the time of the last time that user make a fixation in the j-th AOI ($\{AOI\}_j = 1$). Value 0 in case of no previous visit.
$\{AOI_r1\}_j$ $\{AOI_r2\}_j$ $\{AOI_r3\}_j$	Independent time series features. r_q correspond to 1 if the user makes a fixation in the q-th window previous to the current one without temporal accumulation, 0 if not. That is, if $\{AOI\}_j = 1$ for a window $v^{(t)}$, then $\{AOI_rT\}_j = 1$ for the window (v^{t+T}) .
$\{AOI_end_r1\}_j$	Binary feature. Takes the value 1 if the previous window ends with a fixation in the j-th AOI and 0 if not. This feature captures the fact that fixations can be interrupted by transitions between windows.
End_r1	Corresponds to an AOI indicator in which the previous window ends. It is calculated as: $End_r1 = \sum_{j=1}^n j * AOI_end_r1_j$
$\{[Component]_his\}_c$	The “history” feature for each component. Takes the value 1 if user has made a fixation on the component in the past and 0 if not.
$\{[Component]_end\}_c$	This feature indicates for a time window the last component in which user made a fixation in the previous time window.
$\{[Component]_end_r1\}_c$	Similar to $\{AOI_end_r1\}_j$ features, but per component, built from the feature $[Component]_end$.
$\{[Component]_tslv\}_c$	The features “time since last visit” for each component is similar to $\{AOI_tslv\}_j$.
$\{[Component]_atv\}_c$	Features “average time of visits” for each component give the average time in ms of the fixations made on the component by user in their interaction with the web page. If there are no fixations in that component, it is assigned the value 0.
$\{[Component]_atbv\}_c$	The “average time between visits” features for each component represent the average time in ms that it has taken the user to perform a fixation on the component since last fixation on the same component. It is assigned the value 0 in case there are no fixations.
$\{Heat_AOI\}_j$	Using the signals registered in the training set, a feature of frequency of visualizations is generated in each one of the defined AOI’s, in the time windows made by other users $v_j^{(t)}$ in the times corresponding to the analyzed time window $v^{(t)}$, that is: $Heat_AOI(v^{(t)}) = \frac{\sum_{u=1}^{U_{train}} AOI(v_u^{(t)})}{U_{train}}$

The feature “Heat AOI” adds information about the behavior of other users to the behavior of a particular participant. It is expected that for a stimulus, the model will learn to recognize the visual attention patterns of users over time, at the population level. In this way, the weighting of this feature in the classification models allows learning how to use these patterns in predictions.

Subsequently, visual kinematics features are defined, shown in Table 3. Position, velocity and acceleration measurements are calculated in the X- and Y-coordinates of the gaze position over the stimulus. These features seek to incorporate information in relation to the shape of the ocular movement carried out in a period of time before the beginning of the window, which would allow knowing the future movement patterns. The definitions for the Y coordinate features (*coordY*, *MeanY*, *StdY*, *VelY*, *MeanVelY*, *StdVelY*, *AclY*, *MeanAclY* and *StdAclY*) are similar to those of the X coordinate case.

Finally, features related to the eye-tracking function, both fixation and saccade, are added (see Table 4). These features capture the areas to be visited according to the ocular movements registered for the user.

Table 3. Features of visual kinematics.

Feature	Definition
<i>coordX</i>	It is considered the last position of the previous time window, before starting the time window, registered for the user.
<i>MeanX, StdX</i>	Mean and standard deviation of the gaze position in the X-coordinate recorded in the user interaction with web page in the previous time window.
<i>VelX</i>	Derivative from position data at X-coordinate. This feature takes the velocity value corresponding to the last register of the previous time window.
<i>MeanVelX StdVelX</i>	Mean and standard deviation of the speed in X, of the data recorded in the previous time window.
<i>AccX</i>	Derivative from speed data at X-coordinate. This feature takes the value corresponding to the last acceleration register during navigation.
<i>MeanAccX StdAccX</i>	Mean and standard deviation of the acceleration data in the X-coordinate recorded in the previous time window.

Table 4. Features in Eye-Tracking Function.

Feature	Definition
<i>NFix</i>	Number of fixations made by the user during their interaction with web page along time.
<i>TPromFix TMaxFix TMinFix</i>	Average time, maximum time and minimum duration of the fixations made by the user during navigation.
<i>NSac</i>	Total number of saccades made by the user up to the time before the start of the current time window.
<i>APromSac AMaxSac AMinSac</i>	Average, maximum and minimum amplitude of the saccades taken by the user up to the moment before the beginning of the current window.

4.5. Classification Models

In total, five feature selection methods are executed, namely, maximization of mutual information (MLMIM), jointmutual information (MLJMI) [39], min-redundancy max-relevance (MLMRMR) [40], normalized mutual information feature selection (MIFS) [41] and robust feature selection (RFS) [42], on multilabel classification methods chosen because of their widespread use in the literature. The configurations used are briefly described below:

Ridge Regression (RR): This model is used through the multilabel classification approaches of the binary relevance and classification chain, choosing the number of features in the range 5 : 5 : 40, through the methods of feature selection mentioned. Given the features, different values are chosen for the parameter $\lambda = 0.25 : 0.25 : 2$, and then β parameter is estimated.

k-nearest neighbor (KNN): This method is used with the binary relevance and classification chain approaches, setting the number of features to use in the range 5 : 5 : 40 and the values of number of neighbors in the range $K = 5 : 5 : 30$.

Support vector machine (SVM): This method is used with the binary relevance and classification chain approaches, setting the number of features to be used in the range 5 : 5 : 40, chosen through the feature selection methods, and the parameter $C = 0.2 : 0.2 : 2$.

Multilabel K-nearest neighbor (MLKNN): It is used directly without the approaches of binary relevance or the classification chain, setting the number of features to use in the range 5 : 5 : 40, chosen through the methods of feature selection, and the parameter relative to the number of neighbors as $k = 15$.

4.6. Performance Metrics

Models are evaluated by k-fold cross-validation, so the evaluation metrics are independently averaged between the k training and testing sets. To measure the quality of the solution and choose the best configuration, the following metrics are used. Y_i corresponds to the real label vector of the visit intention components, and Z_i corresponds to the prediction visit intention vector. Only in this case does n represent the number of instances to evaluate.

The problem is originally seen as the search for a user's attention zones over time. This approach allows leeway such that the model does not have to achieve the full prediction of the area served by the user. For this reason, the recall metric is not used directly for the evaluation of a model, which establishes the percentage of activation in the labels that are predicted, but it is used harmoniously in the F-measure metric. On the other hand, it is expected that in cases where the model establishes tag activation, these will actually be areas served by users, minimizing the error to the maximum degree. For this reason, the precision metric, which measures the percentage of activation occurrences that actually happen within the total activation proposed by the model, is used directly in the model selection.

Area under the curve (AUC): A receiver operating characteristic (ROC) curve demonstrates a classifier's ability to discriminate between positive and negative values by changing a threshold value. When the area below the curve is one, it represents a perfect prediction. In the multilabel classification case, it can be calculated from the macro approach, using the rankings of the instances for each label. Ranking $r(x_i, l)$ is defined as the function for which the instance x_i and the label l returns the degree of confidence of l in the prediction Z_i . In this way, the calculation is given by:

$$AUC = \frac{1}{k} \sum_{l \in \mathcal{L}} \frac{|\{x', x'' : r(x', y_l) \geq r(x'', y_l), (x', x'') \in A\}|}{|X_l| |\bar{X}_l|}$$

$$X_l = \{x_i | y_l \in Y_i\}, \bar{X}_l = \{x_i | y_l \notin Y_i\}, A = X_l \times \bar{X}_l$$

Accuracy (Acc): This is the ratio between the number of correctly predicted labels and the actually active labels, given individually by components. It is averaged across all instances:

$$Accuracy = \frac{1}{n} \sum_{i=1}^n \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}$$

Subset Accuracy (Exact): This is the instances percentage, where all coordinates of each vector that are correctly labeled are compared to the total number of labels, that is, the percentage of instances correctly predicted. In the below formula, the operator $\llbracket \cdot \rrbracket$ corresponds to the bracket of Iverson, which is 1 if the logical proposition is true, and 0 if not.

$$Subset Accuracy = \frac{1}{n} \sum_{i=1}^n \llbracket Y_i = Z_i \rrbracket$$

Precision (Pre): The ratio between the number of correctly classified labels and the total number of labels. Intuitively, it corresponds to the percentage of labels predicted as true and that are really important for the instance.

$$Precision = \frac{1}{n} \sum_{i=1}^n \frac{|Y_i \cap Z_i|}{|Z_i|}$$

F-measure (F-me): It is the average between precision and recall calculated in a harmonic way. It measures how many relevant tags are predicted and how many of the predicted labels are relevant.

$$F\text{-measure} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

where *Recall* is the percentage of correctly predicted labels among all labels:

$$\text{Recall} = \frac{1}{n} \sum_{i=1}^n \frac{|Y_i \cap Z_i|}{|Y_i|}$$

4.7. Methodological Limitations

One of the foremost limitations of this study is the utilization of a stimulus in the form of a free visit to a web page on a desktop computer, which cannot be extrapolated to mobile devices. This is a critical consideration, as the use of mobile devices is on the rise and can significantly impact the browsing experience on a web page, due to variations in screen size, modes of manipulation, and the user's field of view. Additionally, while the free visit task may provide a general understanding of page navigation, it is possible that the results would differ if a different task were employed.

While efforts were made to include a diverse range of participants, it is important to acknowledge that the results may be subject to variation if the sample were more diverse in terms of demographic factors such as age or professional status. Additionally, it is important to note that the results of this study may not be generalizable to other websites that possess distinct characteristics, where interactions may present alternative types of dynamic components. Furthermore, the areas of interest defined in this study are specific to this particular website and may not be applicable to other websites with different components or distribution. However, the methodology proposed in this study can be replicated in other websites. Therefore, it is essential to continue research in this area in the future.

5. Experimental Results

Classification models are executed with different configurations, making a cross-validation of 10 folds. The average results measured in the test sets are given in Section 5.1. Test set samples are not balanced, since synthetic data classification is irrelevant. The results therefore correspond to the original distributions of the labels.

5.1. Main Results

The proposed configurations are executed using the programming given in [43] ("MLC Toolbox"). The results are presented in Table 5, where the best models are summarized (choosing the number of features and parameters) as detailed below:

- Ridge Regression Binary Relevance (RR-BR): Best results are achieved with the RFS feature selection method, using 30 predictive features and the parameter $\lambda = 1.75$ (weighting of the coefficients of the variables in the target function).
- Ridge Regression Classification Chain (RR-CC): Best results are achieved with the RFS feature selection method, using 30 predictive features and the parameter $\lambda = 1.75$ (weighting of the coefficients of the variables in the target function).
- KNN Binary Relevance (KNN-BR): Best results are achieved with the MLMIM feature selection method, using 10 predictive features and the parameter $K = 20$ (neighborhood size).
- KNN Classification Chain (KNN-CC): Best results are achieved with the MLJMI feature selection method, using 10 predictive features and the parameter $K = 15$ (neighborhood size).
- SVM Binary Relevance (SVM-BR): Best results are achieved with the MLMIM feature selection method, using 40 predictive features and the parameter $C = 0.2$ (weighting of the loss function in the target function).

- SVM Classification Chain (SVM-CC): Best results are achieved with the RFS feature selection method, using 30 predictive features and the parameter $C = 0.4$ (weighting of the loss function in the target function).
- MLKNN: Best results are achieved with the MLMIM feature selection method, using 10 predictive features.

Table 5. Results of average metrics in the 10-folds are delivered. The standard deviation of these values within the same sets is given in parentheses.

Model	Metrics					AOI Prediction					
	AUC	Exact	Fscore	Acc	Pre	AOI 1	AOI 2	AOI 3	AOI 4	AOI 5	AOI 6
RR-BR	0.844 (0.006)	0.285 (0.010)	0.658 (0.011)	0.566 (0.010)	0.877 (0.007)	0.755 (0.016)	0.686 (0.012)	0.747 (0.009)	0.725 (0.012)	0.785 (0.013)	0.971 (0.006)
RR-CC	0.844 (0.006)	0.285 (0.010)	0.658 (0.011)	0.556 (0.010)	0.877 (0.007)	0.755 (0.016)	0.686 (0.012)	0.747 (0.009)	0.725 (0.012)	0.785 (0.013)	0.971 (0.006)
KNN-BR	0.843 (0.005)	0.303 (0.008)	0.670 (0.009)	0.571 (0.008)	0.879 (0.004)	0.769 (0.019)	0.702 (0.01)	0.760 (0.008)	0.730 (0.01)	0.782 (0.013)	0.972 (0.004)
KNN-CC	0.839 (0.005)	0.302 (0.006)	0.679 (0.007)	0.579 (0.006)	0.878 (0.003)	0.763 (0.015)	0.710 (0.008)	0.729 (0.01)	0.790 (0.01)	0.79 (0.013)	0.967 (0.004)
SVM-BR	-	0.291 (0.012)	0.663 (0.005)	0.564 (0.013)	-	0.777 (0.014)	0.681 (0.01)	0.757 (0.008)	0.712 (0.011)	0.782 (0.013)	0.969 (0.008)
SVM-CC	-	0.287 (0.009)	0.662 (0.009)	0.561 (0.009)	-	0.775 (0.015)	0.682 (0.009)	0.751 (0.007)	0.712 (0.011)	0.775 (0.014)	0.970 (0.006)
ML-KNN	0.844 (0.005)	0.301 (0.009)	0.675 (0.009)	0.574 (0.009)	0.882 (0.005)	0.767 (0.017)	0.708 (0.01)	0.756 (0.007)	0.732 (0.011)	0.781 (0.013)	0.97 (0.004)

The best result is obtained with the model KNN binary relevance (model KKN-BR), with parameter $K = 20$ (number of neighbors considered by the method) and the selection of 10 features by the MLMIM method, since this maximizes the accuracy and subset accuracy metrics. The features chosen are as follows:

- Ocular fixation model feature: AOI_6_r1 , $AOI_1_end_r1$, $AOI_3_end_r1$, $AOI_6_end_r1$, $AOI_2_end_r1$, AOI_3_r1 , AOI_1_r1 AOI ocular fixations are selected.
- Visual kinematics features: $coordY$, $coordX$, $MeanY$
- Eye-tracking function features: No variables are selected.

Figure 6 graphs the approximation made by the MLMIM method for the calculation of discretized (floor approximation of values) mutual information between X (features) and Y (labels), for variables selected across all folds. Higher values indicate a better relation of the variables with respect to the dependent variable. Possible values are in range $0 \leq I(X;Y) \leq \min(H(X), H(Y))$ for each value, where $H()$ is the entropy and $I(X;Y)$ is mutual information value.

A total of ten features are listed. It is observed that all sets require the use of visual kinetics features of gaze position at the beginning of the time window. In addition, it is observed that the average Y-coordinate recorded in the past is sometimes required. Subsequently, it is required to know the activation of any of the AOIs in the previous time window, or the information about the AOI in which the previous window ends.

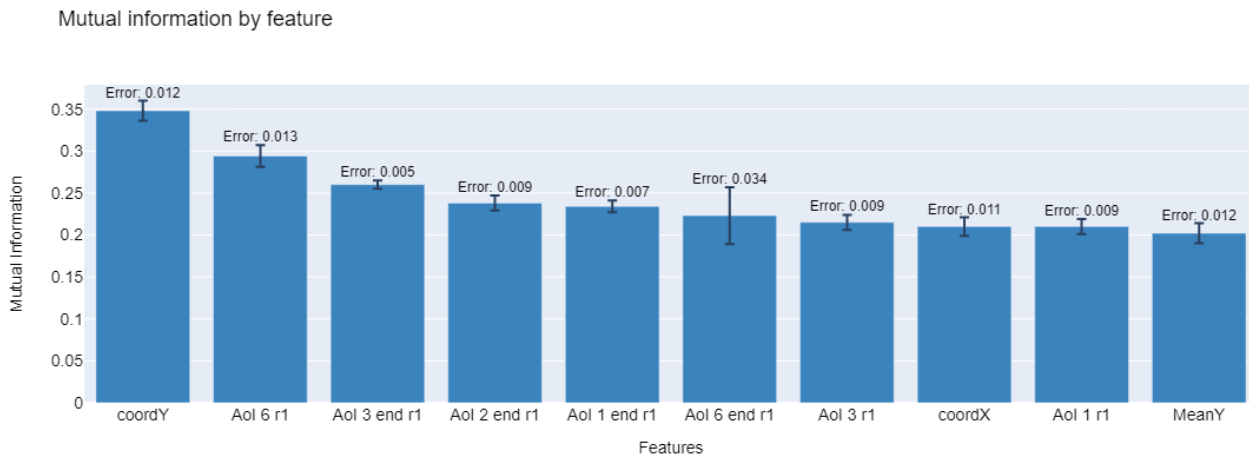


Figure 6. Mutual information for each selected feature by the MLMIM method.

It is interesting to consider that the model only requires knowing the activation of AOIs in a window of time in the past and not the full activity of user interaction. In the sets in which component information is required, the use of the features that indicate the activation of the dynamic components of the website in the previous time window is observed, that is, the menu bar and/or the pop-up registration.

Upon studying the behavior in each of the AOI's (right part of Table 5), it is observed that the model chosen presents the best accuracy values for three AOIs and approaches the best results of the other models for the other AOI's.

The results obtained are validated by comparing them with the biases towards the majority class, which is the proportion of labels with more repetitions with respect to the total. In the formula, m is sample number, MC is majority class and the function inside the argmax calculation is the majority class ratio.

$$MC(AOI_j) = \operatorname{argmax}_{x \in \{0,1\}} \left(\frac{\sum_{q=1}^m \mathbb{1}(AOI_j^{(q)} = x)}{m} \right)$$

For each AOI, the MC ratio values are 0.653, 0.567, 0.513, 0.629, 0.727, and 0.954, for AOI's 1, ..., 6, where the majority class corresponds to the activation of each AOI (1 label), except in the case of AOIs 2 and 3 (0 label). At the same time, the bias to the majority class when considering the complete instance is 0.133 (considering the combination of labels of each AOI of a vector as a label), which corresponds to the activation of AOIs 2 and 3. Therefore, at the individual level and as a whole, the model improves the trivial case of always choosing the majority class.

5.2. Sensitivity Analysis by Time Window Lengths

Sensitivity analysis is carried out, given variations in the time windows' duration to be considered in classification. With this analysis, it is expected that the robustness of the methodology can be evaluated and used for the prediction of visit intentions.

First, models are run with all the combinations again, considering the duration of time windows with $\tau = 3, 10, 15, 20$ s. Table 6 shows the best settings for each prediction model. Table 7 shows the metrics obtained by the best models for each combination.

Table 6. Better configurations for each classification model according to time window duration. FS: Feature selection method, NF: Number of features, P: Parameters.

Model		Time Window Duration (Seconds)			
		$\tau = 3$ s	$\tau = 10$ s	$\tau = 15$ s	$\tau = 20$ s
RR-BR	FS	RFS	MIFS	RFS	F-Score
	NF	35	30	5	5
	P	$\lambda = 0.25$	$\lambda = 1$	$\lambda = 1$	$\lambda = 1.5$
RR-CC	FS	RFS	MLJMI	RFS	F-Score
	NF	35	40	5	5
	P	$\lambda = 0.25$	$\lambda = 0.25$	$\lambda = 1$	$\lambda = 1.5$
KNN-BR	FS	MLMRMR	MLMIM	F-Score	F-Score
	NF	10	40	25	10
	P	$K = 20$	$K = 30$	$K = 25$	$K = 30$
KNN-CC	FS	MLJMI	MLJMI	RFS	F-Score
	NF	25	35	20	10
	P	$K = 10$	$K = 10$	$K = 25$	$K = 25$
SVM-BR	FS	MLMIM	F-Score	F-Score	F-Score
	NF	40	40	5	10
	P	$C = 0.2$	$C = 0.2$	$C = 0.4$	$C = 0.2$
SVM-CC	FS	RFS	MIFS	RFS	F-Score
	NF	35	40	5	5
	P	$C = 1.6$	$C = 0.4$	$C = 2$	$C = 0.4$
ML-KNN	FS	MLJMI	MLJMI	RFS	RFS
	NF	15	40	20	30
	P	$K = 15$	$K = 15$	$K = 15$	$K = 15$

Table 7. Sensitivity analysis metrics on the duration of time windows for the best models.

Model and Metrics		Time Window Duration (Seconds)				
		$\tau = 3$ s	$\tau = 5$ s	$\tau = 10$ s	$\tau = 15$ s	$\tau = 20$ s
RR-BR	AUC	0.876	0.844	0.820	0.788	0.792
	Exact	0.416	0.285	0.198	0.271	0.363
	Fscore	0.683	0.658	0.719	0.781	0.850
	Acc	0.609	0.566	0.606	0.686	0.771
	Pre	0.893	0.877	0.881	0.861	0.913
RR-CC	AUC	0.876	0.844	0.825	0.788	0.792
	Exact	0.416	0.285	0.192	0.271	0.363
	Fscore	0.683	0.658	0.728	0.781	0.850
	Acc	0.609	0.556	0.614	0.686	0.771
	Pre	0.893	0.877	0.885	0.861	0.913
KNN-BR	AUC	0.865	0.843	0.824	0.813	0.804
	Exact	0.423	0.303	0.194	0.233	0.343
	Fscore	0.687	0.670	0.733	0.789	0.845
	Acc	0.613	0.571	0.621	0.687	0.763
	Pre	0.889	0.879	0.887	0.895	0.908
KNN-CC	AUC	0.865	0.839	0.815	0.801	0.800
	Exact	0.424	0.302	0.200	0.246	0.360
	Fscore	0.696	0.679	0.711	0.775	0.844
	Acc	0.621	0.579	0.601	0.674	0.764
	Pre	0.888	0.878	0.881	0.880	0.909

Table 7. Cont.

Model and Metrics		Time Window Duration (Seconds)				
		$\tau = 3$ s	$\tau = 5$ s	$\tau = 10$ s	$\tau = 15$ s	$\tau = 20$ s
SVM-BR	AUC	-	-	-	-	-
	Exact	0.402	0.291	0.174	0.176	0.169
	Fscore	0.686	0.663	0.669	0.722	0.701
	Acc	0.609	0.564	0.554	0.612	0.601
	Pre	-	-	-	-	-
SVM-CC	AUC	-	-	-	-	-
	Exact	0.412	0.287	0.200	0.263	0.305
	Fscore	0.690	0.662	0.680	0.758	0.787
	Acc	0.613	0.561	0.568	0.660	0.702
	Pre	-	-	-	-	-
ML-KNN	AUC	0.871	0.844	0.823	0.805	0.809
	Exact	0.420	0.301	0.193	0.244	0.365
	Fscore	0.694	0.675	0.731	0.782	0.839
	Acc	0.618	0.574	0.619	0.681	0.759
	Pre	0.891	0.882	0.885	0.883	0.903

It can be seen that the settings chosen for each model are not the same for all the values of the temporary duration of the window (τ). The best results are obtained with the use of an RR-BR model for longer time windows. It is interesting to see that the case of $\tau = 3$ obtains a better Exact metric.

First, the values of the subset accuracy (Exact) and accuracy (Acc) metrics are inspected for different values of τ . Figures 7 and 8 shows the graphs of these values for each classification model configuration.

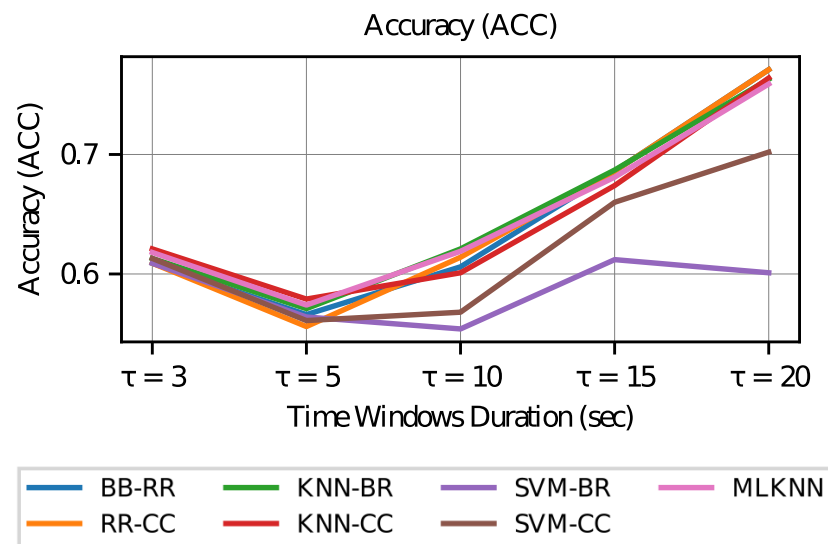


Figure 7. Accuracy vs. Time Window Duration.

From these metrics, it can be seen that if a model with the best estimates of the activation of each AOI is desired, then the case of a longer duration in the time windows ($\tau = 20$) is preferred, since in this case the accuracy (which measures performance by the individual label of each AOI) increases in all models. However, when studying the subset accuracy, that is, the correctness in the estimation of the activation of labels in a vector way (all the AOI's within a time window), the duration of the most extreme time windows is preferred, that is, this metric increases with time windows of shorter duration ($\tau = 3$) or longer duration ($\tau = 20$).

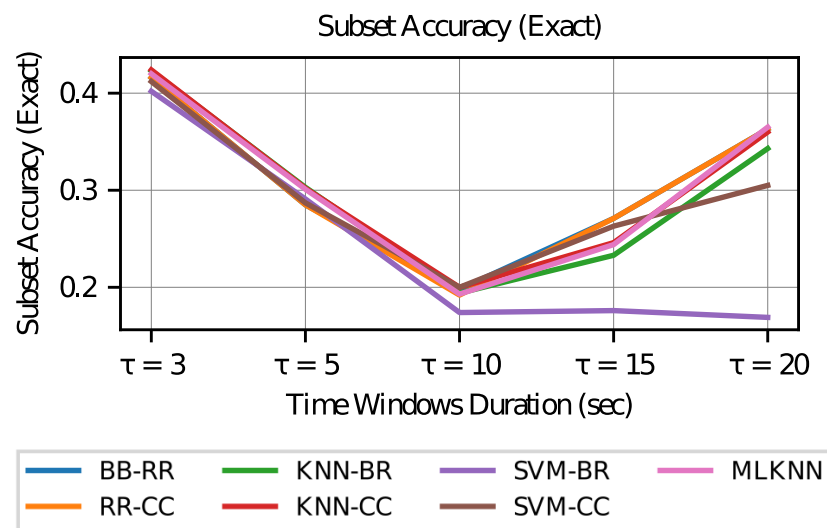


Figure 8. Subset Accuracy vs. Time Window Duration.

To analyze the above, it can be considered that if all models achieve the same level of training (which is not necessarily true), the vectors of the visit intention in smaller and larger time windows show less variability. As the size of the window increases to infinity, the variability in the visit intention tends to zero since it is expected to take the value 1 in all AOI's (although this may not be true in cases where AOI 6 is not visited). In the case of a very small window (tending to zero), the intention to visit all AOI's should tend to zero as there is no time to make transitions to other AOI's.

Subsequently, to better understand the results, the precision is studied. In Figure 9, it is observed that the values for precision appear to be stable according to the lengths of the windows, with the exception of the classifiers based on ridge regression, which tend to decrease slightly more when choosing $\tau = 15$.

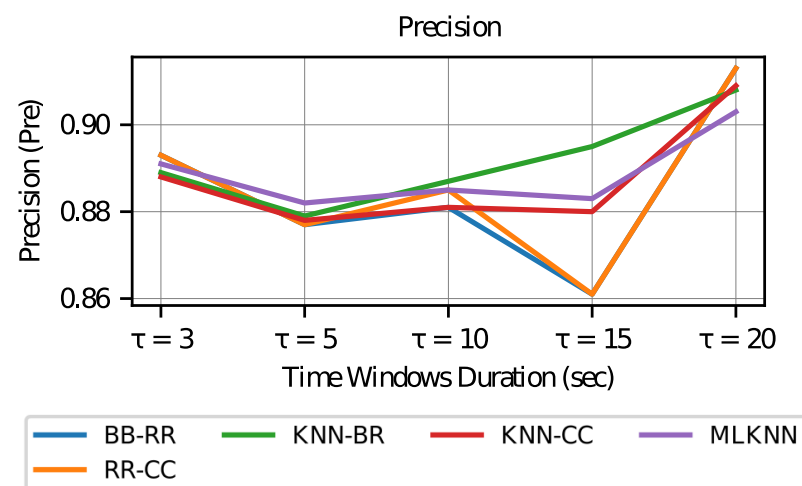


Figure 9. Precision vs. Time Window Duration.

In all cases, precision values greater than 0.8 are achieved, a value commonly considered to be good in the literature. This result means that in the label activation predicted by the model, more than 80% of the cases are performed correctly. In the case of $\tau = 20$, it achieves better accuracy, which may be related to the fact that there are more active labels (as in the case of windows tending to infinity discussed in the previous paragraph), which causes the existence of a greater number of true positive values.

In the case of the F-measure values (see Figure 10), a high trend is observed as the duration of the time windows increases. This result indicates that for longer time windows, greater harmony is achieved between precision and recall.

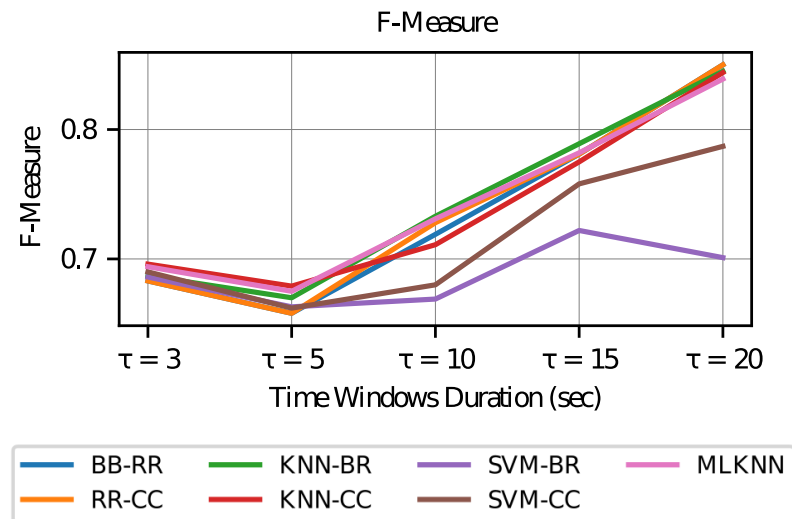


Figure 10. F-Measure vs. Time Window Duration.

Finally, when evaluating the area under the curve (AUC) graph in Figure 11, it can be observed that it decays for values with longer time windows. This observation indicates that the number of times that the ranking value (confidence given to a certain label) of the positive labels (corresponding to the activation of an AOI) exceeds the ranking for a negative label (that is, no activation of an AOI) is lower. Therefore, the power of discrimination between values predicted as positive (true positives) and negative values that are classified as positive (false positive) is lower in longer time windows.

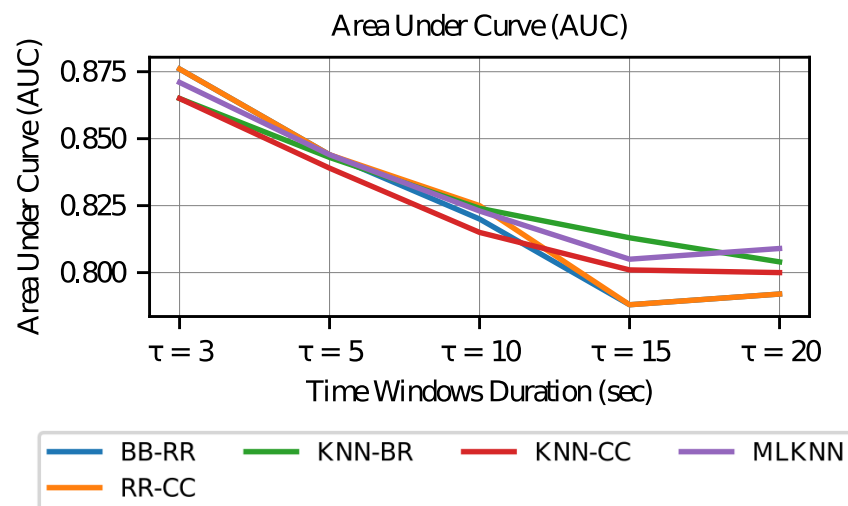


Figure 11. Area Under Curve vs. Time Window Duration.

The drop in AUC values shows that longer time window models have worse discrimination between classes. However, in all cases, values above 0.8 are obtained, which is considered good in the literature.

Given the above, it is concluded that at longer time windows, the accuracy and subset accuracy will tend to improve, which does not necessarily reflect a good model performance. It is necessary to consider the discriminative power between classes of the model, for example, from metrics such as AUC. On the other hand, it is important to

consider the exploratory analysis of the labels (individual and vectorial) in each case (τ chosen), since it is necessary to make a trade-off between the variability of the labels and the time used to make the predictions.

From previous results, it is considered that the estimation of shorter windows is better, since it gives more detailed information of the visual attention in a better discretized time. However, to ensure that the results of the model are valuable, the use of $\tau = 3$ must be evaluated both with respect to the time required by the model for its execution (which does not exceed the duration of the corresponding window) and the variability of the label sufficiency. By the previous criteria, the model from the beginning was considered with $\tau = 5$ as the main model.

5.3. Time Overhead Calculation

The window processing time for the multilabel classification model chosen is given by the signal processing used in the prediction (cleaning of the signals and characteristics calculation), in addition to the execution of the classification.

It is interesting to study the classification times given the duration of the time windows, since on the same equipment, the methodology of the algorithm indicates the speed of computation. The processing times of characteristics will depend on the amount chosen by the model in each of the classifiers considered, according to the duration of the windows, in addition to the processing equipment.

To compare classifiers' performance, all combinations are run to classify windows of different time duration. In particular, windows of length $\tau = \{3, 10, 15, 20\}$ s are considered, in addition to the case of time windows of 5 s. Table 8 shows the mean and standard deviation of execution times of the classifiers in seconds.

Table 8. Time processing of the classification models for different window lengths.

Model	Evaluation Set	Time Window Duration (Seconds)									
		$\tau = 3$ s		$\tau = 5$ s		$\tau = 10$ s		$\tau = 15$ s		$\tau = 20$ s	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
RR-BR	Train	3.008	0.085	1.380	0.051	3.694	0.145	1.497	0.060	0.402	0.004
	Test	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
RR-CC	Train	3.020	0.058	1.392	0.051	0.097	0.010	1.464	0.043	0.378	0.008
	Test	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
KNN-BR	Train	3.252	0.037	0.548	0.020	0.400	0.016	0.356	0.004	0.355	0.006
	Test	1.294	0.007	0.481	0.012	0.092	0.004	0.027	0.002	0.016	0.002
KNN-CC	Train	0.002	0.002	0.000	0.000	0.000	0.000	1.427	0.033	0.395	0.009
	Test	1.302	0.007	0.436	0.004	0.123	0.004	0.111	0.014	0.020	0.002
SVM-BR	Train	3.730	0.052	1.773	0.005	1.205	0.012	0.636	0.010	0.445	0.006
	Test	0.013	0.007	0.008	0.003	0.000	0.000	0.002	0.002	0.002	0.002
SVM-CC	Train	4.511	0.054	1.870	0.031	3.905	0.049	1.559	0.047	0.488	0.016
	Test	0.002	0.002	0.005	0.002	0.003	0.002	0.000	0.000	0.002	0.002
ML-KNN	Train	0.000	0.000	0.538	0.008	0.000	0.000	1.442	0.042	1.992	0.083
	Test	0.300	0.018	0.081	0.002	0.008	0.003	0.020	0.010	0.013	0.008

Execution times for ridge regression models are less than one millisecond in the test sets. This outcome is because the model is based on the storage of the weighting parameters of each variable and only requires a multiplication with the features of the test sets to obtain the result.

Most expensive computational models correspond to those using the KNN base classifier. This is because for a new window in the test set, its classification requires searching for the neighborhood in the training set according to the size given by the parameters of the model.

In Figure 12, it is observed that the execution time decreases with increasing time window length; this effect occurs because with longer times, test sets are generated with a smaller number of temporary window samples.

Finally, it is observed that the classification times are less than the duration of each time window present in the test sets, so they are applicable in an online environment. This approach considers many windows, where the averages in the test sets are 398, 239, 120, 80, and 60 for duration τ of 3, 5, 10, 15 and 20 s, respectively.

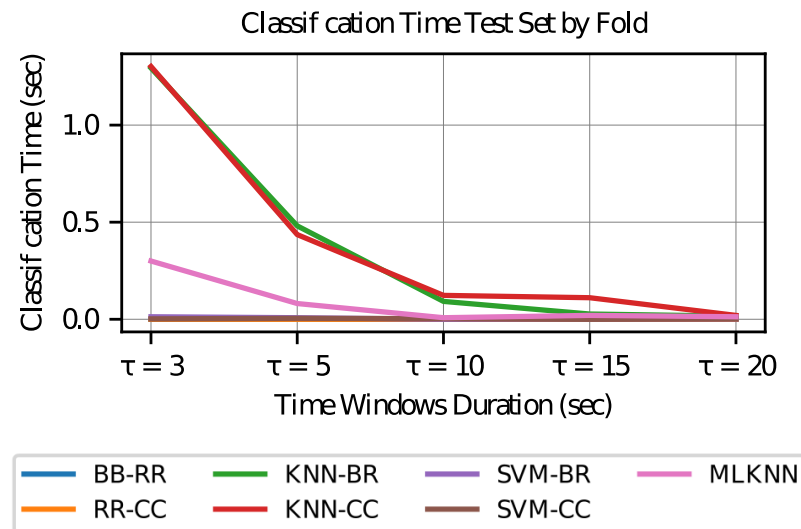


Figure 12. Classification Time Test Set by Fold.

6. Discussion

This paper proposes a method to predict users' visual attention on specific regions of a website called visit intention. This method has been developed in a controlled website where the web objects and their content are known. However, its layout varies, under certain limits, according to the user's interaction. The proposed method for predicting the visit intention does not require knowledge of the website's structure at all periods. However, the underlying idea is that the model learns to recognize the structure implicitly according to the data of the user's behavior and a compilation of visual attention in previous periods.

The trained multilabel classification models allow for predicting visit intentions by jointly incorporating the population-level features of all the AOIs present on the website within each period. In this way, these models comprise a user's visual attention as a whole in all AOIs of the website. The prediction is mostly made by using knowledge of the user's past behavior. In particular, the models use position kinematic features, which consist of the location of the user's gaze at the end of the previous period. With this, the model is encouraged to learn the closest AOIs for the user in a new period based on the current position. The results deliver a total of 10 characteristics in each of the training-test partitions (see Figure 6). In this model, no eye-tracker feature function variables are used, and of the rest, 73% correspond to eye-fixation characteristics and 27% to kinematic characteristics. Indeed, other multilabel classification models can be evaluated, as can neural network models based on deep learning, which is part of our future work.

One limitation of our visit intention approach is that, because of its experimental control, it is not suitable for use in predictions of visual attention in egocentric viewing environments, such as those analyzed in Section 2.3. The reason is that in these environments the stimulus, such as the frames of a video, tends to be less controlled and can suffer considerable variability.

A second limitation is that the study is carried out for the case of bottom-up attention, because users browse a previously unknown website. Therefore, the challenge of predicting visit intention in top-down cases remains open. The reason is because in these cases users tend to seek out certain information in specific regions of the website, depending on clearly defined objectives. At the same time, their website browsing behavior depends on their

knowledge about the layout that each user possesses (learned from interactions with other similar websites).

On the other hand, the sample of participants included in the experimentation corresponds, for the most part, to a convenience sample of engineering students, who one would expect to demonstrate greater fluency in website browsing. The study of results for other types of users would allow a better generalization of the models and, at the same time, could provide useful information regarding their improvement, for example, when considering clusters of similar users in the estimation of parameters. Moreover, this sample corresponds to people from the same location with shared cultural characteristics and native speech corresponding to that of the site studied.

Bearing in mind the above limitations, and although a specific website has been used, a proof-of-concept for the proposed method was achieved. Nevertheless, it is necessary to evaluate these models for other stimuli, which would ensure their generalization, depending on the quality of the results obtained.

Regarding the scalability of the solution, it has been demonstrated that the execution of the classification models takes less time than the duration of the periods that are being predicted. The transmission of data and the calculation of features from the eye-tracker data are factors that must be considered in the selection of the period lengths; however, in this paper, these factors have not been studied in greater depth. Depending on the results of such a scalability study, the use of these approaches in the prediction would allow considering other types of platforms, such as applications in mobile web environments and tablets. This requires experimentation with new visual stimuli in a controlled manner and under these new environments.

7. Conclusions and Future Work

A predictive model is proposed for the concept of visit intention, providing information on the areas of interest of a website that will be visited by a user over a given time period. A proof-of-concept of the proposed method has been achieved by experimenting with a specific informative website that involves dynamic components. The results demonstrate the feasibility of creating these predictive models and test the hypothesis stating that the considered features are useful predictors of visual attention.

The proposed methodology bears an advantage in that it is not required to know the structure of the website in each period, as in other studies reported in the literature; rather, learning the dynamic changes through the gaze data registered for a user, estimating models through population behavior as characteristics and employing a user's visual kinetics to train individual models for each instance of prediction are utilized. The model is applicable from an analysis of execution times, and criteria have been provided in the selection of the window duration used in the segmentation of the total time of user interaction, preferring the use of windows of $\tau = 5$ s for this case.

As future work, it is expected that these results can be integrated into more advanced models of visual attention, in addition to the study of improvements based on models for segmentation of users and the use of deep learning methods. On the other hand, it is necessary to explore in greater detail the impact of the variables used in the models, to find explainable patterns of the relationship with visit intentions. Furthermore, it is interesting to consider the study of the methodology of this style in other visual environments (e.g., mobile devices) and in cases related to specific tasks (e.g., searching for a news item on the site).

Author Contributions: Conceptualization, A.J.-M.; methodology, F.D.-G. and A.J.-M.; software, F.D.-G. and A.J.-M.; validation, F.D.-G. and A.J.-M.; formal analysis, F.D.-G. and A.J.-M.; investigation, F.D.-G. and A.J.-M.; resources, A.J.-M.; data curation, F.D.-G.; writing—original draft preparation, F.D.-G. and A.J.-M.; writing—review and editing, A.J.-M.; visualization, F.D.-G.; supervision, A.J.-M.; project administration, A.J.-M.; funding acquisition, A.J.-M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by Proyecto Superintendencia de Seguridad Social (SUSESO) grant number IST-2019, and CORFO program IMA+ Innovacion en Manufactura Avanzada. The second author gratefully acknowledge financial support from ANID PIA/PUENTE AFB220003 (Instituto Sistemas Complejos de Ingenieria) and ANID, FONDECYT 11130252, FONDEF IT21I0059. This article is part of the work of the Red Internacional para el Fortalecimiento y Desarrollo de la Salud Mental Digital, Folio FOVI220026.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Ethics and Biosafety Committee for Research of the Faculty of Physical Sciences and Mathematics at the University of Chile (protocol code #05 and date of approval 21 June 2013.)

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to their containing information that could compromise the privacy of research participants.

Acknowledgments: Research assistantship from Cristian Retamal is greatly acknowledged. Conversations with Tonino Jiménez-Fuenzalida have benefited this research.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bazzani, L.; Larochelle, H.; Torresani, L. Recurrent mixture density network for spatiotemporal visual attention. *arXiv* **2016**, arXiv:1603.08199.
2. Guo, C.; Ma, Q.; Zhang, L. Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
3. Zhao, J.; Siagian, C.; Itti, L. Fixation bank: Learning to reweight fixation candidates. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3174–3182.
4. Zhai, Y.; Shah, M. Visual attention detection in video sequences using spatiotemporal cues. In Proceedings of the 14th ACM International Conference on Multimedia, Santa Barbara, CA, USA, 23–27 October 2006; pp. 815–824.
5. Fang, Y.; Wang, Z.; Lin, W.; Fang, Z. Video saliency incorporating spatiotemporal cues and uncertainty weighting. *IEEE Trans. Image Process.* **2014**, *23*, 3910–3921. [[CrossRef](#)] [[PubMed](#)]
6. Sitzmann, V.; Serrano, A.; Pavel, A.; Agrawala, M.; Gutierrez, D.; Masia, B.; Wetzstein, G. Saliency in VR: How do people explore virtual environments? *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 1633–1642. [[CrossRef](#)] [[PubMed](#)]
7. Marmitt, G.; Duchowski, A. Modeling Visual Attention in VR: Measuring the Accuracy of Predicted Scanpaths. Ph.D. Thesis, Clemson University, Clemson, SC, USA, 2002.
8. Huang, Y.; Cai, M.; Li, Z.; Sato, Y. Predicting gaze in egocentric video by learning task-dependent attention transition. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 789–804.
9. Zhang, M.; Ma, K.; Lim, J.; Zhao, Q.; Feng, J. Deep future gaze: Gaze anticipation on egocentric videos using adversarial networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4372–4381.
10. Shen, C.; Zhao, Q. Webpage saliency. In Proceedings of the 13th European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 33–46.
11. Shen, C.; Huang, X.; Zhao, Q. Predicting eye fixations on webpage with an ensemble of early features and high-level representations from deep network. *IEEE Trans. Multimed.* **2015**, *17*, 2084–2093. [[CrossRef](#)]
12. Li, Y.; Zhang, Y. Webpage Saliency Prediction with Two-stage Generative Adversarial Networks. *arXiv* **2018**, arXiv:1805.11374.
13. Li, J.; Su, L.; Wu, B.; Pang, J.; Wang, C.; Wu, Z.; Huang, Q. Webpage saliency prediction with multi-features fusion. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 674–678.
14. Xu, P.; Sugano, Y.; Bulling, A. Spatio-temporal modeling and prediction of visual attention in graphical user interfaces. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, San Jose, CA, USA, 7–12 May 2016; pp. 3299–3310.
15. Katsuki, F.; Constantinidis, C. Bottom-up and top-down attention: Different processes and overlapping neural systems. *Neuroscientist* **2014**, *20*, 509–521. [[CrossRef](#)] [[PubMed](#)]
16. Rutishauser, U.; Walther, D.; Koch, C.; Perona, P. Is bottom-up attention useful for object recognition? In Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 27 June–2 July 2004.
17. Connor, C.E.; Egeth, H.E.; Yantis, S. Visual attention: Bottom-up versus top-down. *Curr. Biol.* **2004**, *14*, R850–R852. [[CrossRef](#)] [[PubMed](#)]
18. Johnson, S.; Subha, T.D. A Study on Eye Fixation Prediction and Salient Object Detection in Supervised Saliency. *Mater. Today Proc.* **2017**, *4*, 4169–4181. [[CrossRef](#)]
19. Tang, H.; Chen, C.; Bie, Y. Prediction of human eye fixation by a single filter. *J. Signal Process. Syst.* **2017**, *87*, 197–202. [[CrossRef](#)]

20. Wang, W.; Shen, J.; Shao, L.; Porikli, F. Correspondence driven saliency transfer. *IEEE Trans. Image Process.* **2016**, *25*, 5025–5034. [[CrossRef](#)]
21. Kruthiventi, S.S.; Gudisa, V.; Dholakiya, J.H.; Venkatesh Babu, R. Saliency unified: A deep architecture for simultaneous eye fixation prediction and salient object segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 5781–5790.
22. Wang, W.; Shen, J. Deep visual attention prediction. *IEEE Trans. Image Process.* **2017**, *27*, 2368–2378. [[CrossRef](#)]
23. Wang, W.; Lai, Q.; Fu, H.; Shen, J.; Ling, H.; Yang, R. Salient object detection in the deep learning era: An in-depth survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 3239–3259. [[CrossRef](#)]
24. Fu, K.; Fan, D.-P.; Ji, G.-P.; Zhao, Q.; Shen, J.; Zhu, C. Siamese network for RGB-D salient object detection and beyond. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 5541–5559. [[CrossRef](#)]
25. Fan, D.; Li, T.; Lin, Z.; Ji, G.-P.; Zhang, D.; Cheng, M.-M.; Fu, H.; Shen, J. Re-thinking co-salient object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 4339–4354.
26. Bylinskii, Z.; Judd, T.; Borji, A.; Itti, L.; Durand, F.; Oliva, A.; Torralba, A. MIT Saliency Benchmark. Available online: <http://saliency.mit.edu/> (accessed on 16 November 2022).
27. Bylinskii, Z.; Judd, T.; Oliva, A.; Torralba, A.; Durand, F. What do different evaluation metrics tell us about saliency models? *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 740–757. [[CrossRef](#)]
28. Le Meur, O.; Baccino, T. Methods for comparing scanpaths and saliency maps: Strengths and weaknesses. *Behav. Res. Methods* **2013**, *45*, 251–266. [[CrossRef](#)]
29. Wang, W.; Shen, J.; Dong, X.; Borji, A.; Yang, R. Inferring salient objects from human fixations. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 1913–1927. [[CrossRef](#)]
30. Wang, W.; Chen, C.; Wang, Y.; Jiang, T.; Fang, F.; Yao, Y. Simulating human saccadic scanpaths on natural images. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011; pp. 441–448.
31. Meur, O.L.; Coutrot, A.; Liu, Z.; Roch, A.L.; Helo, A.; Rama, P. Computational model for predicting visual fixations from childhood to adulthood. *arXiv* **2017**, arXiv:1702.04657.
32. Wang, W.; Shen, J.; Lu, X.; Hoi, S.C.H.; Ling, H. Paying attention to video object pattern understanding. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 2413–2428. [[CrossRef](#)]
33. Xie, J.; Cheng, M.-M.; Ling, H.; Borji, A. Revisiting Video Saliency Prediction in the Deep Learning Era. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 220–237.
34. Zheng, Q.; Jiao, J.; Cao, Y.; Lau, R.W. Task-driven webpage saliency. In Proceedings of the 15th European Conference on Computer Vision (ECCV 2018), Munich, Germany, 8–14 September 2018; pp. 300–316.
35. Yamada, K.; Sugano, Y.; Okabe, T.; Sato, Y.; Sugimoto, A.; Hiraki, K. Can saliency map models predict human egocentric visual attention? In Proceedings of the 2010 Asian Conference on Computer Vision, Queenstown, New Zealand, 8–9 November 2010; pp. 420–429.
36. Yamada, K.; Sugano, Y.; Okabe, T.; Sato, Y.; Sugimoto, A.; Hiraki, K. Attention prediction in egocentric video using motion and visual saliency. In Proceedings of the Pacific-Rim Symposium on Image and Video Technology, Gwangju, Republic of Korea, 20–23 November 2011; pp. 277–288.
37. Li, Y.; Fathi, A.; Rehg, J.M. Learning to predict gaze in egocentric video. In Proceedings of the 2013 IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; pp. 3216–3223.
38. Charte, F.; Rivera, A.J.; del Jesus, M.J.; Herrera, F. MLSMOTE: Approaching imbalanced multilabel learning through synthetic instance generation. *Knowl.-Based Syst.* **2015**, *89*, 385–397. [[CrossRef](#)]
39. Sechidis, K.; Nikolaou, N.; Brown, G. Information theoretic feature selection in multi-label data through composite likelihood. In Proceedings of the Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), Joensuu, Finland, 20–22 August 2014; pp. 143–152.
40. Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *27*, 1226–1238. [[CrossRef](#)] [[PubMed](#)]
41. Jian, L.; Li, J.; Shu, K.; Liu, H. Multi-Label Informed Feature Selection. In Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI), New York, NY, USA, 9–15 July 2016; pp. 1627–1633.
42. Nie, F.; Huang, H.; Cai, X.; Ding, C.H. Efficient and robust feature selection via joint ℓ_2 , 1-norms minimization. In Proceedings of the 23rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 6–9 December 2010; pp. 1813–1821.
43. Kimura, K.; Sun, L.; Kudo, M. MLC Toolbox: A MATLAB/OCTAVE Library for Multi-Label Classification. *arXiv* **2017**, arXiv:1704.02592.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.