# Analysing User Gaze on the Web

## Detecting genuinely read parts of text

Author: Bc. Patrik Hlavac
Supervisor: Ing. Marian Simko, PhD.
**Faculty of Informatics and Information Technologies**

## SIMPLE ILLUSTRATION OF PROCESSING ON LAYER LEVELS:

» GAZE visualized by points of fixations

» reading interest measured by index distances



Tomu však bude koniec. Počítače budú komunikovať cez kanál požiadavkami, ktoré nie sú pre človeka čitateľné. Vďaka tomu budú kratšie a prinesú nové možnosti. Kým doposiaľ prehliadač otvoril pre urýchlenie načítavania stránky niekoľko rozhovorov súčasne a v každom z nich pýtal súbor za súborom, po novom urobí spojenie iba jedno a bude zo seba sypať čo potrebuje. Podľa BBC sa dáta zoradia podľa požiadaviek prehliadača a server ich zasa bude musieť popreraďovať tak, aby pri čakaní na vykonanie jednej úlohy mohol zatiaľ posielať výsledok inej, ktorá je hotová. Vznikne tak intenzívny dialóg optimalizovaný...

192 193 194 195 196 197 198
259 i 260 i+1 261 i+2 262 i+3
299 300 301 302 303 304 305
340 341 342 343 344 345 346

» content structure represented by unique elements

» detecting genuinely read passages of text

## Abstract

We propose a method of reading detection from gaze data. Eye tracking devices provide irreplaceable information about a user's gaze. This work deals with the possibilities of identifying user interaction in the educational system. Our algorithm takes into account user's fixation data and maps their coordinates onto single word elements. These are then processed with respect to their relative word distance. Rule-based solution works by considering the sequences in the order of their occurrence. Unlike studies that calculate distance in points that eyes moved around the screen, we consider the distance of words in the vector.

**4 levels of webpage interaction:**

» whether is the user present
» whether is the user watching
» whether is the user reading
» whether is the user learning



A device called "Eye-tracker"

## Our Approach

- **mapping gaze fixation coordinates onto single word elements**
- **processing them with respect to their word distance in the order of their occurrence**
- **detect reading by measuring the "Reading Interest"**

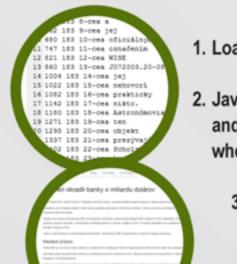| Word position | Content | String length | No. of fixations | Fixations duration | Reading Interest | ... + |
|---|---|---|---|---|---|---|
| 40 | zaujima | 7 | 2 | 0.35 | 1.10 | |
| 41 | sa | 2 | 2 | 0.28 | 1.50 | |
| 42 | o | 1 | 1 | 0.15 | 0.70 | |
| 43 | rozpočet | 8 | 3 | 1.13 | 1.90 | |
| 44 | sprostredkovateľ | 16 | 5 | 0.82 | 2.30 | |
| 45 | medzi | 5 | 4 | 0.57 | 2.20 | |
| 46 | manažérmi | 9 | 3 | 0.60 | 2.00 | |
| ... | | | | | | |

Example of Document Reading Model

Document Reading Model represents all instances of scanned words with records of properties and characteristics of the current session.

We consider the following variable "RI" of reading interest, which maintain the current state of reading intensity. The sequence of fixation determines the variable growth, or penalty.

Variable storing updating value "UV" indicates how the reading interest will be increased due to activity. It is based on the level of words approach, so it subtracting the order of the current word and the previous word in the vector, to determine the specific distance between words.

The basic assumption of reading detection at the level of words is the ability to detect words as unique elements of the code. By pre-processing the source code before the detection we can change the structure of the source code so that each word represents a separate element with a unique label. With this modification we get a number of ROI, which are approximately of the spatial size of a given word.



1. Load page (with text content).

2. JavaScript process the HTML code and transfer the "ContentMap" to server, where it is further processed.

3. "ContentMap" layout simulates the Web page display.

4. Load gaze data and filter the fixations.

5. Remove outliers, for better mapping on content.

6. Align gaze data with content.

7. Expand or shrink gaze data, so they fit content borders.

8. Run algorithm for mapping the fixations.

## Experiment

Based on data from the eye tracker we apply the model, which will be verified explicitly by evaluation questionnaire for each user. Sitting with each user includes solving of the tasks, once prepared and uniformly interpreted to all participants. The participant is asked to read eight articles prepared in the specified order with the intention of learning the parts of his choice. He answers five questions related to clarification whether he knows the answer or whether he at least saw information after every article.

| Participants | Read articles | Answer Familiarity | Answer correctness |
|---|---|---|---|
| 15 | 120 | 0.89 | 0.82 |

We set up experiment in Tobii Studio (Tobii TX300) with webpage articles content in rotation with instruction sheets. This device provides us with three hundred coordinates per second, every row contains additional information about timestamp, media, whether it is fixation, saccade or none and coordinates information for both eyes. User session took slightly less than one hour.

## Conclusions

We propose a new method for genuinely read text detection - based on eye fixations mapped to specific words in the document.
Not every Heatmap shows the reading. The main contribution consists of providing a method which not only shows where the user is looking, but detects the reading!

## Contact

Please do not hesitate to contact me if you have any further questions » patrik@hlavac.sk