WIKI ANALYSIS

Using Gaze-Tracker

**Group members:** Kartikeya Pise(2017csb1086)

Sidhant Jain(2017csb1112)

Sibee Sanchay(2017eeb1170)

**Supervised By:** Neeru Dubey Ma’am

**Instructor:** Dr. S.R.S. Iyengar

**Course Name:** Social Computing

**Course Code:** CS522

# INTRODUCTION

In this project, we have created a web application that outputs the level of proficiency of the reader whenever they read an article on Wikipedia using our web application. We use gaze-tracking software to track the eye movements of the user in order to get the desired data. We track multiple features in the background, for example, the word counts, fixation duration on a particular word, scrolling speed, etc. We train the data obtained using a Linear SVM model and check the performance of our model using metrics like Recall and Precision. We were successful in displaying the proficiency level of the reader at the end of the reading session with additional analysis regarding fixations and gaze movements. The procedure and demo video is uploaded on Youtube.

[Click here to watch our demo video on youtube.](https://youtu.be/SwgzKxQlKz0)

# DETAILS

We have used Django templates and HTML to render our web application and to deliver the output to the reader. After compiling our python code the server starts running and the gaze tracker is opened simultaneously on another window. The gaze tracker(tracker.py file) captures the Eye movements of the reader continuously until he/she stops recording it to end his/her session. After the recording stops, ‘x’ and ‘y’ coordinates and the timestamps created are stored in a file automatically by our code. Then our software uses that data in the generated file to process the information and output the results as shown in the demo video.

# COMPILATION PROCEDURE:

(Mentioned in details at **website/Readme.md)**

## Step 1:

Go to the “website” directory of the project.

command:

cd Gaze-and-Expertise/website

## Step 2:

Install all the requirements needed for the project.

command:

make install

## Step 3:

Initiate the server and open the browser link.

command:

make djrun

## Step 4:

Search any article to read on the corresponding webpage and read the article.

## Step 5:

Stop the recording to view the output.

# **ML M**ODEL

## Procedure:

We asked our colleagues to read a few articles on topics related to their competency and few articles on distant topics while using the gaze-tracker. We selected features based on the general feedback on approaches of reading, like how fast one reads, for how long one looks at one particular term (and their vicinity), scrolling speed, back-scrolling speed, gaze points, etc. Then we assign ‘0’ and ‘1’ scores to each reading session depending on if the topic was of their competency or not, respectively.

## Model:

Before feeding the data to the model, we scaled the features in a standard way for better accuracy. We intended to do a Principal Component Analysis but it did not change the accuracy so we dropped it. We trained the Linear SVM Classifier model with C-factor being 1 and the loss function being squared hinge.

We calculated recall and precision score for test data and both turned out to be 1, i.e. the model is near perfect for the given amount of data.

# MAJOR DEPENDENCIES

## Numpy:

* 1. Since this gaze tracking is a slow process in itself, NumPy is used to provide an array object that is up to 50x faster than traditional Python lists.
  2. It also provides a lot of supporting functions that make working with np arrays very easy.

## Open CV python:

* 1. OpenCV is a huge open-source library for computer vision, machine learning, and image processing.
  2. OpenCV-Python is a library of Python bindings designed to solve computer vision problems. In an easy and faster way.

## Django:

* 1. Django is an open-source framework for backend web applications based on Python.
  2. The community of django is widespread. Making it easy and fast to slow bugs.
  3. Since our gaze tracking features are based on python, it is very convenient to use a backend web development tool like django which is also based on python.

## Pandas:

* 1. A very fast and easy to use python package to process csv files. Which is widely used in our project.

## Wikipedia-API:

* 1. To get access to the wikipedia articles, so that it can be shown to our website user.
  2. Our project also uses this dependency to get the exact words that were read by the user.

# OUTPUT TEST RUN:

## Info:

No.of words read: 23

Ratio Score: -0.0566710452242447

Summary Score: 51.34

Article Score: -905.93

Reading level: Not Expert

## Reading Summary:

In mathematics, a tuple is a finite ordered list (sequence) of elements. There is only one 0-tuple, referred to as the empty tuple.

## Fixations > dbscan:

## Heat Maps: