

Data modelling and data processing generated by human eye movements

Velin Kralev¹, Radoslava Kraleva², Petia Koprinkova-Hristova³

^{1,2}Department of Informatics, South-West University "Neofit Rilski", Blagoevgrad, Bulgaria

³Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Sofia, Bulgaria

Article Info

Article history:

Received Oct 6, 2020

Revised Mar 6, 2021

Accepted Mar 19, 2021

Keywords:

Data modelling

Data processing

Human eye movements

Rapid application development

ABSTRACT

Data modeling and data processing are important activities in any scientific research. This research focuses on the modeling of data and processing of data generated by a saccadometer. The approach used is based on the relational data model, but the processing and storage of the data is done with client datasets. The experiments were performed with 26 randomly selected files from a total of 264 experimental sessions. The data from each experimental session was stored in three different formats, respectively text, binary and extensible markup language (XML) based. The results showed that the text format and the binary format were the most compact. Several actions related to data processing were analyzed. Based on the results obtained, it was found that the two fastest actions are respectively loading data from a binary file and storing data into a binary file. In contrast, the two slowest actions were storing the data in XML format and loading the data from a text file, respectively. Also, one of the time-consuming operations turned out to be the conversion of data from text format to binary format. Moreover, the time required to perform this action does not depend in proportion on the number of records processed.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Velin Kralev

Department of Informatics

South-West University "Neofit Rilski"

66 Ivan Michailov str., 2700 Blagoevgrad, Bulgaria

Email: velin_kralev@swu.bg

1. INTRODUCTION

One of the important tasks in research is the processing of large data sets that are generated in various experiments. The many possibilities for generating experimental data and their heterogeneous nature lead to certain difficulties in their modeling. Our aim is to model the data so that the result set is easy to analyze. In the present study, we modeled the experimental data according to the relational model [1]-[4], but used client datasets [5] to store, process, and analyze the information. Thus, the summarized experimental data were convenient for analysis, which allowed to formulate the relevant conclusions. After considering the importance of this aspect of research in order to easily share the results obtained, we were motivated to present the research work in this article.

Many approaches to creating visual tasks based on glass patterns (stimuli) are considered in various scientific studies [6], [7]. In recent years, various areas that are related to the analysis of human decision-making processes have been explored [8]-[11]. Techniques in the field of artificial intelligence are widely used [12]. They are used to model experimental processes [13]-[15], to analyze experimental results [16], and to examine a variety of indicators, such as age and/or gender [17], [18]. One important problem associated

with this type of research is the processing of the data used. Different data models are used, for example relational and object-oriented [5], [19], [20], different data access technologies, for example, remote database servers and web services [21], and different software approaches [22], [23].

The subject of the present study is an approach to processing heterogeneous information generated by different devices and different software applications. The data is processed through client datasets and can be stored in XML files or in binary files (used by the client dataset). The advantage of using the xml format is that it is portable and self-describing [24]. The disadvantage of the XML format is the multiple increase in file size due to the addition of self-descriptive tags to the data [25]. When storing data in the binary file format used by client datasets, the file sizes are significantly smaller. This allows faster buffering of data from files in the computer's memory. In addition, the client datasets provide methods for converting data from the internal (binary) format to external text formats (for example, as XML or CSV).

The file converter application was developed for the purposes of this study. It provides the ability to process different types of files related to experiments. Such files are, for example, the stimulus files (BIN), the files with information about the movement of the eyes during the experiments (EYE), the result files from the experimental sessions (DAT) and others. Other features of this application are also loading, processing, filtering, converting and exporting all supported files. Thus, this application is actually a conversion module. This provides the ability to process heterogeneous information (collected from different files (with different file structure)), which in the next step is converted to a common file format (with binary structure). Once the application has converted the heterogeneous information to a common format, it can merge, split, reorganize the data at the structural level and export this data to external files (with a different file structure if necessary). Note that when the data is finally exported, it is already merged into a common file format, unlike its original form, which is heterogeneous. Different programming languages and methods for software development and testing are presented in [26]-[28].

In order to provide access to all collected information (including visual stimulus and result files), it is necessary to choose an approach to modeling the data [29]. The aim is to easily summarize and analyze these data. Two approaches are most commonly used for this purpose [5]. The first approach is to store the data as it is (raw) and then, just before analyzing it, to summarize it. The second approach is that the data is imported into a pre-designed local or remote database, from which the summary information is then extracted for analysis via SQL queries [30]. Access to such a database can be done through a database management system-DBMS [1], [31]. Architectures of multi-user systems based on remote database servers with the ability to perform calculations and analysis of experimental data have been presented in other research [21], [30].

2. RESEARCH METHOD

The experimental data that need to be processed are generated by different applications during the experimental sessions. Therefore, the purpose of this article is not to present an analysis of the experimental results, but to present a way to process the generated data during these experiments. Visual tasks (stimuli) are visualized on a computer display for all participants in the experiment. The stimuli are preliminarily generated, with each stimulus containing exactly 100 frames. In turn, each frame contains exactly 50 points. The rules for generating stimuli are described in detail in [32].

During each experimental session, 140 stimuli were visualized sequentially for each participant in the experiment. The order in which the stimuli were visualized was predefined. During each experimental session, the eye movements of each participant were recorded. This is done with a special device-Saccadometer and specialized software connected to it. The result files contain the information about the eye movements of each participant for each experimental session. These files have a specific file format that is not officially published. Therefore, only specialized software associated with the saccadometer can process these files. However, the information from these files can be exported to external text files (with a .TXT extension, respectively). Table 1 shows the structure of the exported data (as text).

The exported files are characterized by the fact that they are large in size (in the order of tens of megabytes). This is because the eye movements of the participants were scanned at a sampling frequency of 1000 Hz (i.e. 1 kHz). This means that 1000 sets of values were generated in 1 second from the saccadometer. Since each stimulus contains exactly 100 frames, and each frame has a maximum duration of 33 milliseconds, the total time to visualize a stimulus is approximately 3300 milliseconds. After each stimulus, there was a pause of 500 milliseconds before the next stimulus was visualized. At the beginning of each experimental session, the saccadometer was calibrated according to the position of the participant's head. The time for this calibration is approximately 900 milliseconds. On the other hand, each participant could (via some of the mouse buttons) stop prematurely visualizing each stimulus. This usually happened when a participant "recognized" the direction of the stimulus before all the 100 frames of that stimulus were

visualized (i.e., before 3300 milliseconds had passed). Given all these specific features and variable parameters of the experimental sessions determine their different duration. Therefore, in some experimental sessions, the saccadometer can generate up to over 800 000 sets of values. These "sets" of values include all parameters (fields), which were presented in Table 1, respectively - "No", "Eye X", "Eye Y", "Screen", and "Mic".

Table 1. Specification of the structure of TXT files

No.	Column	Description
1	No	The millisecond in which the saccadometer generates the corresponding values.
2	Eye_X	The angle (in degrees) at which the participant's view by the X-axis is shifted.
4	Eye_Y	The angle (in degrees) at which the participant's view by the Y-axis is shifted.
4	Screen	Presence or not of stimulus on the screen. Values above 800 indicate that a stimulus has been displayed on the computer screen at this time.
5	Mic	The value of the microphone signal at the same time (in decibels).

Storing data in text format does not provide a convenient way to process it. These data need to be converted to the relevant base (structural) types. The conversion of data (from text format) can be done before the process of their export (to files with a specific file structure depending on the type of the respective values). Initially, in this process, the data is buffered in a two-dimensional string array (matrix). This data structure has a dimension of 5 columns (as many as the number of exported fields from the text files) and a number of rows as many as the number of generated sets with values from the saccadometer. Since the buffered data are of a character type (string), in the next step it is necessary to convert it to the corresponding base types. In order to implement this step of the data conversion process, a relational scheme "EyeMovments" was created with the following analytical presentation: EyeMovments = {No, EyeX, EyeY, Screen, Mic}. The attributes of this relational schema correspond to the fields of the text file, but their internal representation is of the corresponding base type (domains) depending on the nature of the data. Table 2 presents the structure of the "EyeMovments" relation. The "Class name" column in Table 2 contains the names of the corresponding classes (of the corresponding attributes) used by the client dataset.

Table 2. Specification of the attributes of the "EyeMovments" relation

No.	Field name	Class name	Base type	Data type specification
1	No	TIntegerField	Integer	{-2147483648..2147483647}, Format: Signed 32-bit
2	EyeX	TFloatField	Double	{2.23e-308 .. 1.79e+308}, Significant decimal digits: 16, Size in bytes: 8
4	EyeY	TFloatField	Double	{2.23e-308 .. 1.79e+308}, Significant decimal digits: 16, Size in bytes: 8
4	Screen	TWordField	Word	{0..65535}, Format: Unsigned 16-bit
5	Mic	TSmallIntField	SmallInt	{-32768..32767}, Format: Signed 16-bit

The primary key of the "EyeMovment" relation is the attribute "No". The values of this attribute are unique because they correspond to specific milliseconds for a given experimental session. Therefore, the values of this attribute also guarantee the uniqueness of each row in the "EyeMovments" table. The other four attributes - "EyeX", "EyeY", "Screen" and "Mic" have similar meanings as the fields described in Table 1.

The need for automated data processing led to the development of the file converter application. This application has functions for loading, processing and storing data (respectively files) used in the experimental sessions. The file converter application converts the data from strings to the corresponding base types (for example in integer, word, and double). The converted data is buffered in client datasets. Therefore, we will use the abbreviation CDS (short for Client DataSet) to denote them.

3. RESULTS AND DISCUSSION

The subject of this study is the file converter application and its main functions. During the experiments, the application was run on a 64-bit Windows 10 operating system. The hardware configuration has the following characteristics: Intel (R) Core (TM) i5-8265U processor at 1.60-3.90 GHz and 8 GB DDR4 at 2400 MHz random access memory (RAM). Among 264 experimental sessions (containing a total of 38 259 visualized stimuli of 35 participants), 10% were selected (randomly). This formed a representative sample of 26 experimental sessions. The data from all analyzed experimental sessions were initially stored in text files (generated by the software distributed with the saccadometer). Through the file converter

application, these text files were converted and stored both in the internal format used by the client datasets-CDS, as well as in XML files. Table 3 presents the sizes of these files in MB and bytes, respectively. The last row in Table 3 contains the average values of the corresponding columns. The ratio between the sizes of the different files (depending on their format) are shown in Figure 1 (for all analyzed files) and in Figure 2 depending on the average values of their sizes (in megabytes (MB) on the x-axis).

Table 3. File sizes depending on their format

File id	File name	Number of rows	TXT file size	TXT size (in bytes)	CDS file size	CDS size (in bytes)	XML file size	XML size (in bytes)
1	AT_M.TXT	745 842	23.3 MB	24 473 600	32.0 MB	33 566 720	94.4 MB	99 012 608
2	AT_F.TXT	641 966	19.6 MB	20 598 784	27.5 MB	28 889 088	81.4 MB	85 417 984
3	DD_C.TXT	882 738	25.7 MB	26 992 640	37.8 MB	39 727 104	109.0 MB	115 224 576
4	DD_M.TXT	906 120	25.5 MB	26 771 456	38.8 MB	40 775 680	112.0 MB	118 312 960
5	DN_M.TXT	605 686	17.6 MB	18 538 496	25.9 MB	27 258 880	76.3 MB	80 035 840
6	DN_S.TXT	615 748	17.6 MB	18 550 784	26.4 MB	27 709 440	77.2 MB	80 990 208
7	DZ_C.TXT	674 560	20.8 MB	21 811 200	28.9 MB	30 355 456	85.8 MB	90 066 944
8	DZ_F.TXT	674 276	21.1 MB	22 208 512	28.9 MB	30 343 168	86.7 MB	90 951 680
9	GB_M.TXT	445 550	14.3 MB	15 003 648	19.1 MB	20 049 920	56.6 MB	59 379 712
10	GB_S.TXT	551 836	17.0 MB	17 846 272	23.6 MB	24 834 048	71.8 MB	75 374 592
11	KA_C.TXT	428 400	12.8 MB	13 447 168	18.3 MB	19 279 872	53.4 MB	56 078 336
12	KA_S.TXT	459 438	13.5 MB	14 209 024	19.7 MB	20 676 608	57.5 MB	60 383 232
13	LB_C.TXT	354 994	10.2 MB	10 747 904	15.2 MB	15 978 496	44.8 MB	46 997 504
14	LB_M.TXT	377 386	11.6 MB	12 206 080	16.1 MB	16 986 112	48.9 MB	51 314 688
15	LD_C.TXT	707 012	20.1 MB	21 180 416	30.3 MB	31 817 728	87.8 MB	92 082 176
16	LD_F.TXT	770 076	22.3 MB	23 453 696	33.0 MB	34 656 256	95.9 MB	100 618 240
17	PV_F.TXT	511 588	15.3 MB	16 056 320	21.9 MB	23 023 616	64.0 MB	67 108 864
18	PV_M.TXT	483 388	15.5 MB	16 343 040	20.7 MB	21 753 856	61.4 MB	64 442 368
19	PM_M.TXT	660 246	19.4 MB	20 422 656	28.3 MB	29 712 384	82.5 MB	86 564 864
20	PM_S.TXT	658 688	18.4 MB	19 324 928	28.2 MB	29 642 752	81.7 MB	85 688 320
21	RN_C.TXT	775 176	22.1 MB	23 265 280	33.2 MB	34 885 632	95.8 MB	100 487 168
22	RN_S.TXT	744 708	22.4 MB	23 494 656	31.9 MB	33 513 472	94.3 MB	98 983 936
23	SN_F.TXT	429 686	12.5 MB	13 205 504	18.4 MB	19 337 216	53.3 MB	55 959 552
24	SN_S.TXT	431 702	12.0 MB	12 607 488	18.5 MB	19 427 328	54.1 MB	56 770 560
25	VG_F.TXT	819 958	24.1 MB	25 313 280	35.1 MB	36 900 864	102.0 MB	107 208 704
26	VG_C.TXT	788 214	22.4 MB	23 556 096	33.8 MB	35 471 360	99.1 MB	103 948 288
Average values		620 961	18.4 MB	19 293 420	26.6 MB	27 945 118	78.0 MB	81 900 150

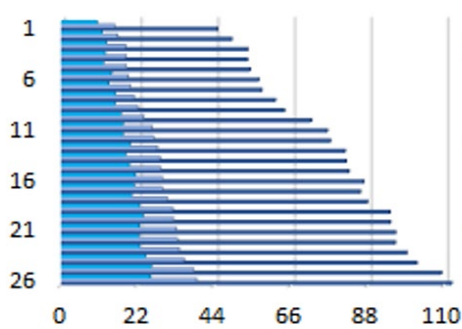


Figure 1. A ratio between sizes of all files

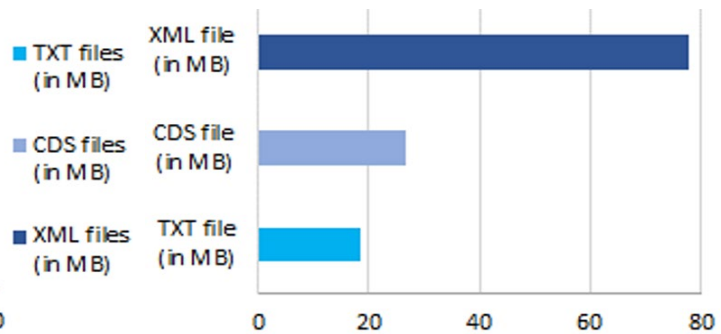


Figure 2. A ratio between average file sizes

The values in Table 3 and the diagrams in Figures 1 and 2 show that the text files have the smallest size. In contrast, XML files have the largest size. The size of the text files after conversion to XML format increases to 4.25 times. However, after converting text files to CDS format, their size increases only up to 1.45 times. In addition, the average aspect ratio of a CDS file size to an XML file size is also very large, approximately three times (exactly 2.93 times). Therefore, the CDS format should be preferred, as the sizes of the CDS files and the TXT files are commensurate. On the other hand, the data in text files are of the character type (although they "look like" numeric values), while the data in CDS format are of the corresponding numeric types. The actions that were performed by the File Converter application and that were analyzed are as follows: time to load a TXT file, time to load a CDS file, time to load an XML file, time to convert data from text to binary CDS format, time to store the data in a CDS file and time to store the data

in an XML file. Due to the multitasking mode of the Windows operating system, in order to more accurately measure the average execution time of each action, five different executions of each action were made for each file. Table 4 presents the averages of the different time measurements for each action.

Table 4. Time to perform the various actions

File id	File name	Number of rows	Load TXT (in ms)	Load CDS (in ms)	Load XML (in ms)	Convert (in ms)	Save to CDS (in ms)	Save to XML (in ms)
1	AT_M.TXT	745 842	5 041 (676)	391 (52)	2 782 (373)	3 031 (406)	937 (126)	9 472 (1 270)
2	AT_F.TXT	641 966	4 368 (680)	346 (54)	2 375 (370)	2 641 (411)	792 (123)	7 890 (1 229)
3	DD_C.TXT	882 738	5 896 (668)	437 (50)	3 250 (368)	3 672 (416)	1 187 (134)	10 856 (1 230)
4	DD_M.TXT	906 120	6 386 (705)	469 (52)	3 367 (372)	8 672 (957)	1 241 (137)	11 359 (1 254)
5	DN_M.TXT	605 686	4 112 (679)	328 (54)	2 234 (369)	5 734 (947)	794 (131)	7 681 (1 268)
6	DN_S.TXT	615 748	4 125 (670)	328 (53)	2 258 (367)	5 922 (962)	748 (121)	7 313 (1 188)
7	DZ_C.TXT	674 560	4 514 (669)	344 (51)	2 514 (373)	2 797 (415)	894 (133)	8 357 (1 239)
8	DZ_F.TXT	674 276	4 529 (672)	359 (53)	2 507 (372)	5 531 (820)	921 (137)	8 475 (1 257)
9	GB_M.TXT	445 550	3 172 (712)	234 (53)	1 656 (372)	1 844 (414)	543 (122)	5 385 (1 209)
10	GB_S.TXT	551 836	3 871 (701)	296 (54)	2 045 (371)	5 015 (909)	737 (134)	6 875 (1 246)
11	KA_C.TXT	428 400	3 016 (704)	219 (51)	1 578 (368)	1 797 (419)	584 (136)	5 128 (1 197)
12	KA_S.TXT	459 438	3 260 (710)	250 (54)	1 687 (367)	4 422 (962)	554 (121)	5 489 (1 195)
13	LB_C.TXT	354 994	2 535 (714)	188 (53)	1 321 (372)	1 484 (418)	432 (122)	4 375 (1 232)
14	LB_M.TXT	377 386	2 692 (713)	203 (54)	1 403 (372)	3 344 (886)	481 (127)	4 513 (1 196)
15	LD_C.TXT	707 012	4 922 (696)	359 (51)	2 592 (367)	2 953 (418)	864 (122)	8 922 (1 262)
16	LD_F.TXT	770 076	5 406 (702)	390 (51)	2 832 (368)	7 141 (927)	1 019 (132)	9 728 (1 263)
17	PV_F.TXT	511 588	3 625 (709)	266 (52)	1 876 (367)	2 125 (415)	618 (121)	6 391 (1 249)
18	PV_M.TXT	483 388	3 447 (713)	256 (53)	1 803 (373)	4 125 (853)	578 (120)	5 693 (1 178)
19	PM_M.TXT	660 246	4 697 (711)	328 (50)	2 427 (368)	2 812 (426)	904 (137)	8 109 (1 228)
20	PM_S.TXT	658 688	4 566 (693)	343 (52)	2 416 (367)	5 656 (859)	884 (134)	8 024 (1 218)
21	RN_C.TXT	775 176	5 375 (693)	391 (50)	2 844 (367)	3 204 (413)	1 016 (131)	9 775 (1 261)
22	RN_S.TXT	744 708	5 016 (674)	391 (53)	2 765 (371)	6 453 (867)	1 004 (135)	9 285 (1 247)
23	SN_F.TXT	429 686	2 953 (687)	219 (51)	1 578 (367)	1 797 (418)	541 (126)	5 369 (1 250)
24	SN_S.TXT	431 702	2 943 (682)	219 (51)	1 594 (369)	3 688 (854)	562 (130)	5 153 (1 194)
25	VG_F.TXT	819 958	5 764 (703)	422 (51)	3 011 (367)	3 375 (412)	1 028 (125)	9 617 (1 173)
26	VG_C.TXT	788 214	5 281 (670)	422 (54)	2 906 (369)	6 986 (886)	1 035 (131)	9 502 (1 206)
Average values		620 961	4 289 (693)	323 (52)	2 293 (369)	4 085 (657)	804 (129)	7 644 (1 228)

Note that the values in parentheses indicate the average time for an action if it were to be executed with 100 000 records (rows). These values are necessary because the analyzed files have different durations. In order to make an analysis of the times for performing the different actions, it is necessary to correlate these times to a single duration (in this case 100 000 records). The last row of Table 4 shows the average values for all columns. Figure 3 shows a chart of the average execution time for loading different types of files-CDS, XML, and TXT. Figure 4 shows a chart of the average times to perform all actions (in milliseconds).

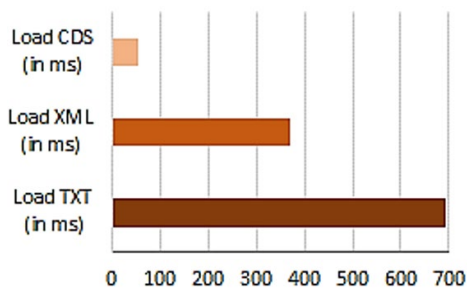


Figure 3. Average loading time by file types

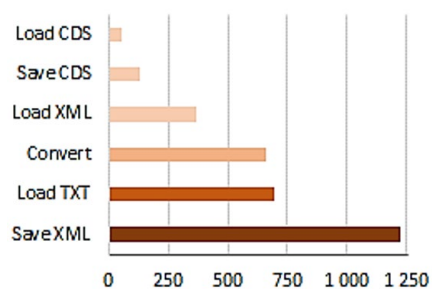


Figure 4. Average time to perform actions

Table 4 and Figure 3 show that the file converter application loads TXT files the slowest. Loading the XML files (although they are largest) takes less time than loading the TXT files. The application loads CDS files the fastest, and compared to loading TXT files this process takes about 13 times shorter time. Compared to loading TXT files, this process takes about 13 times fewer, respectively, compared to loading XML files, this process takes about 7 times shorter time. Figure 4 shows that the two fastest actions are respectively when loading the data from CDS files and when storing the data in CDS files. Given that CDS

files are also relatively small in size, it can be concluded that they are the most suitable for use, both in terms of their size and in terms of speed in processing their data. Figure 4 also shows that one of the time-consuming operations is the conversion of the data from text format to binary CDS format (used by the client datasets). A more in-depth analysis of the values in Table 4 ("Convert (in ms)" column) shows that this action does not depend on the number of records processed. Figure 5 presents this trend.

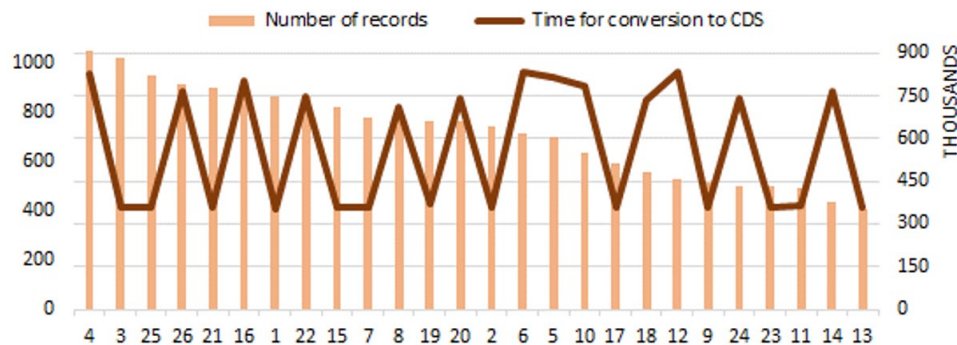


Figure 5. Influence of the number of records on the time for converting text data to binary format

The indices (id) of the analyzed files are located on the x-axis of the diagram in Figure 5. The files are arranged in descending order of size, respectively 4, 3, 25, ..., 14, 13 (the right y-axis (number of records in thousands)). On the left y-axis values in milliseconds, corresponding to the times for converting 100 000 records from text format to CDS format are visualized. The two graphs in the diagram in Figure 5 are synchronized, i.e., for the same file both the number of all records and the time required for conversion 100 000 of records are shown. Figure 5 shows that the time to convert the data from text format to CDS format does not depend only on the number of records in the files. For example, it can be seen that for a file with id = 4, the time required to convert 100 000 records (from text format to CDS format) is 957 milliseconds. However, for a file with id = 3, the conversion time per 100 000 records is only 416 milliseconds, approximately twice shorter. The situation is similar for files with numbers 26 and 25, where the conversion times per 100 000 records are 886 milliseconds (for a file with id = 26) and 412 milliseconds (for a file with id = 25), respectively. These differences in values are obtained by converting the data from text format to CDS format after performing a recognition algorithm for the beginning and end of each stimulus. The algorithm analyzes the stored data related to the eye movements of the respective participant in the experiment. During the data analysis, the recognition algorithm takes into account three different factors, respectively: decision time by the respective participant (for the respective stimulus) and the values of the microphone and "Screen" signal.

4. CONCLUSION

One of the important tasks in research is the processing of data that is generated during various experiments. The large number of possibilities for generating experimental data and the heterogeneous nature of these data lead to certain difficulties in their modeling and processing. The present study was related to the modeling of data and analysis of the time for their processing by performing various actions. The data were generated by a special device-saccadometer and were processed through the application developed for this purpose - File Converter. An approach based on the relational model for organizing data was used, but the storage and processing of these data was done through client datasets. In this way, the summarized data were convenient for analysis, which allowed to draw the appropriate conclusions. The experiments were performed with 26 randomly selected files from a total of 264 experimental sessions. This represents 10% of all available experimental files. The data from each experimental session was stored in three different formats, respectively text, binary (internal format supported by client datasets) and XML based. The results showed that in terms of file sizes, the text format is the most compact. The binary CDS format is comparable to the text format, while the XML-based format is approximately 3 to 4 times larger than the other two formats (when storing the same data). Another important feature of the formats is that in the text format the data are of a character type, while in CDS form the data are of the corresponding numeric types. Several things were identified in relation to the data actions performed. First, the two fastest actions are respectively when the File Converter application loads the data from CDS files and when storing the data in CDS files.

Second, the two slowest actions are storing the data in XML format and loading the data from a text file, respectively. Third, one of the time-consuming operations is the conversion of data from text format to binary CDS format. Moreover, the time required to perform this action does not depend in proportion on the number of records processed. This is because during the conversion of data from text format to binary CDS format, an algorithm for recognizing the beginning and end of each stimulus is executed in parallel. The execution time of this algorithm is influenced by various factors, such as the decision time of a participant (for a given stimulus), the values of the microphone signal and the values of the "Screen" signal. A direction for expanding the research is the design and development of software for recreating the experimental sessions. This specialized software must be able to process and visualize both the stimuli and the eye movements of the participants on the computer screen.

REFERENCES

- [1] C. J. Date, "Database Design and Relational Theory (Theory in Practice)," *O'Reilly Media*, 2012.
- [2] S. Bergamaschi, F. Guerra, M. Interlandi, R. Trillo-Lado, and Y. Velegrakis, "Combining user and database perspective for solving keyword queries over relational databases," *Information Systems*, vol. 55, pp. 1-19, 2016, doi: 10.1016/j.is.2015.07.005.
- [3] G. Manjunath, M. N. Murty, and D. Sitaram, "Combining heterogeneous classifiers for relational databases," *Pattern Recognition*, vol. 46, no. 1, pp. 317-324, 2013, doi: 10.1016/j.patcog.2012.06.015.
- [4] G. M. Afify, A. E. Bastawissy, and O. M. Hegazy, "A hybrid filtering approach for storage optimization in main-memory cloud database," *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 329-337, 2015, doi: 10.1016/j.eij.2015.06.007.
- [5] R. S. Krалева, V. Krалев, N. Sinyagina, and P. Koprinkova-Hristova, "Design and analysis of a relational database for behavioral experiments data processing," *International Journal of Online Engineering*, vol. 14, no. 2, 2018, Art. No. 117, doi: 10.3991/ijoe.v14i02.7988.
- [6] R. Marandi, P. Madeleine, Ø. Omland, N. Vuillerme, and A. Saman, "Eye movement characteristics reflected fatigue development in both young and elderly individuals," *Scientific Reports*, vol. 8, no. 1, 2018, Art. No. 13148, doi: 10.1038/s41598-018-31577-1.
- [7] R. Ratcliff, "A diffusion model account of response time and accuracy in a brightness discrimination task: fitting real data and failing to fit fake but plausible dat," *Psychonomic Bulletin and Review*, vol. 9, pp. 278-291, 2002, doi: 10.3758/BF03196283.
- [8] R. Ratcliff and J. N. Rouder, "Modeling response times for two-choice decisions," *Psychological Science*, vol. 9, no. 5, pp. 347-356, 1998, doi: 10.1111/1467-9280.00067.
- [9] J. Drugowitsch, G. C. DeAngelis, E.M. Klier, D. E. Angelaki, and A. Pouget, "Optimal multisensory decision-making in a reaction-time task," *Elife*, vol. 14, no. 3, 2014, Art. No. e03005, doi: 10.7554/eLife.03005.001.
- [10] S. D. Brown and A. Heathcote, "The simplest complete model of choice response time: linear ballistic accumulation," *Cognitive Psychology*, vol. 57, no. 3, pp. 153-178, 2008, doi: 10.1016/j.cogpsych.2007.12.002.
- [11] R. Ratcliff and P. Smith, "Perceptual discrimination in static and dynamic noise: the temporal relation between perceptual encoding and decision-making," *Journal of Experimental Psychology: General*, vol. 139, no. 1, pp. 70-94, 2010, doi: 10.1037/a0018128.
- [12] P. Koprinkova-Hristova, M. Stefanova, B. Genova, N. Bocheva, R. Krалева, and V. Krалев, "Features Extraction from Human Eye Movements Via Echo State Network," *Neural Computing and Applications*, vol. 32, no. 9, pp. 4213-4226, 2020, doi: 10.1007/s00521-019-04329-z.
- [13] P. Bayerl and H. Neumann, "Disambiguating visual motion through contextual feedback modulation," *Neural Computation*, vol. 16, no. 10, pp. 2041-2066, 2004, doi: 10.1162/0899766041732404.
- [14] S. Grossberg, E. Mingolla, and C. Pack, "A neural model of motion processing and visual navigation by cortical area MST," *Cerebral Cortex*, vol. 9, no. 8, pp. 878-895, 1999, doi: 10.1093/cercor/9.8.878.
- [15] P. Koprinkova-Hristova, N. Bocheva, S. Nedelcheva, M. Stefanova, B. Genova, R. Krалева et al., "STDP Plasticity in TRN within Hierarchical Spike Timing Model of visual Information Processing," *IFIP Advances in Information and Communication Technology*, vol. 583, pp. 279-290, 2020, doi: 10.1007/978-3-030-49161-1_24.
- [16] N. Bocheva, D. Angelova, and M. Stefanova, "Age-related changes in fine motion direction discriminations," *Experimental Brain Research*, vol. 228, no. 3, pp. 257-278, 2012, doi: 10.1007/s00221-013-3559-4.
- [17] N. Bocheva, O. Georgieva, and M. Stefanova, "Data analysis of age-related changes in visual motion perception," in *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence ICAART 2011*, Rome, Italy, vol. 1, 2011, pp. 556-561.
- [18] J. Pratt, T. Welsh, and M. Dodd, "Growing older does not always mean moving slower: examining aging and the Saccadic motor system," *Journal of Motor Behavior*, vol. 38, no. 5, pp. 373-382, 2006, doi: 10.3200/JMBR.38.5.373-382.
- [19] M. D. Humphries and K. Gurney, "A means to an end: Validating models by fitting experimental data," *Neurocomputing*, vol. 70, no. 10-12, pp. 1892-1896, 2007, doi: 10.1016/j.neucom.2006.10.061.
- [20] G. Manjunath, M. N. Murty, and D. Sitaram, "Combining heterogeneous classifiers for relational databases," *Pattern Recognition*, vol. 46, no. 1, pp. 317-324, 2013, doi: 10.1016/j.patcog.2012.06.015.
- [21] V. Krалев, R. Krалева, N. Sinyagina, P. Koprinkova-Hristova, and N. Bocheva, "An analysis of a web service based approach for experimental data sharing," *International Journal of Online Engineering*, vol. 14, no. 9, pp. 19-34, 2018, doi: 10.3991/ijoe.v14i09.8740.

- [22] R. H. Kulkarni, P. Padmanabham, M. Harshe, K. K. Baseer, and P. Patil, "Investigating Agile Adaptation for Project Development," *International Journal of Electrical and Computer Engineering*, vol. 7, no. 3, pp. 1278-1285, 2017, doi: 10.11591/ijece.v7i3.pp1278-1285.
- [23] B. B. Semlali, C. El Amrani, and S. Denys, "Development of a Java-based application for environmental remote sensing data processing," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 3, pp. 1978-1986, 2019, doi: 10.11591/ijece.v9i3.pp1978-1986.
- [24] N. Bikakis, C. Tsinaraki, N. Gioldasis, I. Stavrakantonakis, and S. Christodoulakis, "The XML and semantic web worlds: Technologies, interoperability and integration: A survey of the state of the art," *Studies in Computational Intelligence*, vol. 418, pp. 319-360, 2013, doi: 10.1007/978-3-642-28977-4_12.
- [25] D. Atilgan, I. S. Altıngöve, and O. Ulusoy, "On the size of full element-indexes for XML keyword search," *Lecture Notes in Computer Science*, vol. 7224, pp. 556-560, 2012, doi: 10.1007/978-3-642-28997-2_63.
- [26] H. Gamido and M. Gamido, "Comparative review of the features of automated software testing tools," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 5, pp. 4473-4478, 2019, doi: 10.11591/ijece.v9i5.pp4473-4478.
- [27] R. Kraleva, V. Kraleva, and D. Kostadinova, "A methodology for the analysis of block-based programming languages appropriate for children," *Journal of Computing Science and Engineering*, vol. 13, no. 1, pp. 1-10, 2019, doi: 10.5626/JCSE.2019.13.1.1.
- [28] H. Brata, D. Liang, and S. H. Pramono, "Software Development of Automatic Data Collector for Bus Route Planning System," *International Journal of Electrical and Computer Engineering*, vol. 5, no. 1, pp. 150-157, 2015, doi: 10.11591/ijece.v5i1.pp150-157.
- [29] U. Tosun, "Distributed Database Design: A Case Study," *Procedia Computer Science*, vol. 37, pp. 447-450, 2014, doi: 10.1016/j.procs.2014.08.067.
- [30] V. Dimitrieski, M. Celikovic, S. Aleksic, S. Ristic, A. Alargt, and I. Lukovic, "Concepts and evaluation of the extended entity-relationship approach to database design in a multi-paradigm information system modeling tool," *Computer Languages, Systems & Structures*, vol. 44, pp. 299-318, 2015, doi: 10.1016/j.cl.2015.08.011.
- [31] P. Nicolaos and T. Katerina, "Simple-talking database development: Let the end-user design a relational schema by using simple words," *Computers in Human Behavior*, vol. 48, pp. 273-289, 2015, doi: 10.1016/j.chb.2015.02.002.
- [32] N. B. Bocheva, B. Z. Genova, and M. D. Stefanova, "Drift Diffusion Modeling of Response Time in Heading Estimation Based on Motion and Form Cues," *International Journal of Biology and Biomedical Engineering*, vol. 12, pp. 75-83, 2018.