Current Development and Technology in the Information Extraction for Clinical Narrative Text

Yan Ge*

School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, 266061, China

*geyan@qust.edu.cn

Abstract

Most of the clinical narratives are free-text forms. The information extractions from clinical narrative text are more complicated than those from other biomedical texts, such as books, articles, literature abstracts, and so on. In this paper, we review recent published researches on the implication and technology of information extraction from free-text clinical narratives. We mainly introduce the Hidden Markov Model, Support Vector Machine, Ontology and other combined method used in information extraction for clinical narratives. The objectives, typical systems, applications, technologies and future challenge of information extraction for clinical narrative are introduced in this paper.

Keywords

Information Extraction; Clinical Narrative Text; Electronic Medical Record; Hidden Markov Model; Ontology

Introduction

In biomedical domain, Electronic Medical Record (EMR) is a rich source of clinical narrative information. The good EMR is needed for improving the quality of health care and reducing medical errors. Most of the available clinical narratives are in free-text forms. So, it is difficult for us to conduct searching, summarization, decision-support, or statistical analysis based on free-text forms of clinical narrative data. In order to solve those problems, Information Extraction (IE) is needed for analyzing clinical narrative data. IE is a specialized sub-domain of Natural Language Processing (NLP) [1]. It can extract interested specific types of information from texts or Web[2]. There are two application categories of NLP in biomedical area: biomedical text and clinical narrative text. The biomedical text can be defined as that appears in books, articles, literature abstracts, and so on. But, clinical narrative texts are written by clinicians in the clinical settings. It includes all the narratives about the patient record. These texts may describe the patient's personal information, pathologies, medical histories, symptoms found during interviews or therapy procedures[3].

In this paper, we reviewed the recent development and technology of research about information extraction from free-text clinical narrative data. The previous research on this topic is described in detail in a review by S.M. Meystre et al. at 2008. This interview focuses on recent publications after 2008.

The Objectives of Information Extraction from Clinical Narrative Data

The extraction of information from clinical narrative text is a relatively new field of research in the biomedical domain. Research on information extraction from the biomedical literature has been well described in reviews by Cohen et al. [4] and by Zweigenbaum et al. [5]. The clinical narrative text has unique characteristics that different from scientific biomedical text. Thus, information extraction from clinical narrative text is harder than from biomedical text. There are many reasons making clinical narrative text different from biomedical text[3]. Firstly, some clinical narrative texts are ungrammatical. Secondly, there are many abbreviations, acronyms, and local dialectal shorthand phrases in the clinical narrative texts. Lastly, there are misspellings in clinical narrative texts. All these issues complicate the application of IE on the clinical narrative texts. In spite of the challenges, excellent

2324-7037/15/02 043-08

doi: 10.12783/ijcsa.2015.0402.01

researches have been done in the field of extract information from clinical narrative text. The vast amounts of clinical narrative data are useful only after the information contained in them can be properly extracted and understood. The extraction of information from clinical narrative text can be used to searching, summarization, decision-support, or statistical analysis.

Typical Systems for Information Extraction from Clinical Narrative Text

There are many software systems support information extraction from clinical text. Such as Linguistic String Project-Medical Language Processor(LSP-MLP), Medical Language Extraction and Encoding System (MedLEE), Medication information extraction system for clinical narratives(MedEx), Unified Medical Language System (UMLS) and The Clinical Text Analysis and Knowledge Extraction System (cTAKES).

LSP-MLP system (http://www.cs.nyu.edu/cs/projects/lsp/) is a system for computerized text processing and presentation. It was developed by Dr. Sager and other researchers at the Courant Institute of Mathematical Sciences of New York University. The Linguistic String Project (LSP) was a sustained research effort in the computer processing of language based on the linguistic theory of Zellig Harris: linguistic string theory, transformation analysis, and sublanguage grammar. Medical Language Processor (MLP) is a system that transforms free-text clinical documents into an XML structured representation of the information in the documents[6].

MedLEE system (http://www.medlingmap.org/taxonomy/term/80) is an Open Health Natural Language Processing system. It was created by Carol Friedman and other researchers in collaboration with the Department of Biomedical Informatics at Columbia University, the Radiology Department at Columbia University, and the Department of Computer Science at Queens College of CUNY. The aim of MedLEE is to extract, structure, and encode clinical information in textual patient reports so that the data can be used by subsequent automated processes[7].

MedEx system (http://knowledgemap.mc.vanderbilt.edu/research/content/about-us) was developed by researchers at School of Biomedical Informatics in The University of Texas Health Science Center at Houston (UTHealth). It uses a context-free grammar and regular expression parsing to process free text clinical notes. It process free-text clinical records to recognize medication names and signature information, such as drug dose, frequency, route, and duration[8].

UMLS system (http://www.nlm.nih.gov/research/umls/quickstart.html) was developed by U.S. National Library of Medicine. It is a set of files and software that combine many health, biomedical vocabularies and standards. It can enable interoperability between computer systems and can be used to enhance or develop applications, such as electronic health records, language translators, and classification tools[9]. The UMLS has three tools, which also can be called as the Knowledge Sources:

- Metathesaurus: Terms and codes from many vocabularies, including CPT®, ICD-10-CM, LOINC®, MeSH®, RxNorm, and SNOMED CT®
- Semantic Network: Broad categories (semantic types) and their relationships (semantic relations)
- SPECIALIST Lexicon and Lexical Tools: Natural language processing tools

cTAKES is a clinical Text Analysis and Knowledge Extraction System (http://en.wikipedia.org/wiki/CTAKES) [10]. The development of cTAKES started in 2006 by a team of physicians, computer scientists and software engineers at the Mayo Clinic. The development team was led by Dr. Guergana Savova & Dr. Christopher Chute. It is an open-source natural language processing system for information extraction from electronic medical record clinical free-text. It processes clinical notes, identifying types of clinical named entities — drugs, diseases/disorders, signs/symptoms, anatomical sites and procedures. cTAKES was built using the Unstructured Information Management Architecture framework and OpenNLP natural language processing toolkit. Its components are specifically trained for the clinical domain, and create rich linguistic and semantic annotations that can be utilized by clinical decision support systems and clinical research. The current open source release consists of the following components [11]:

- Sentence boundary detector
- Tokenizer
- Normalizer
- Part-of-speech (POS) tagger
- Shallow parser
- Named entity recognition (NER) annotator, including status and negation annotators.

The typical examples of natural language processing systems introduced above have also required improving performance of information extraction from clinical narrative text. Some researchers do information extraction from clinical narrative text based on some component of above systems. Several authors do research about more simple systems to implement specific information extraction from clinical narrative text, such as clinical case statistical analysis, clinical decision support, and drug testing analysis.

Applications of Information Extraction from Clinical Narrative Text

Information retrieval and information extraction are significant issues in the medical and health care domains. The vast amounts of clinical narrative data are useful only after the information contained in them can be properly extracted and understood. The extraction of information from clinical narrative text can be used to clinical information searching clinical case statistical analysis, drug testing analysis, clinical decision support, and so on. In this section, we will introduce the current applications of information extraction from clinical narrative text.

Clinical Information Searching

The accuracy of the retrieved information and obtaining it in a time critical situation are extremely important. Jon Patrick et al. proposed an intelligent clinical notes system to help doctors extract useful information from free-text clinical narrative. It was developed for a real real-time environment and integrated into the existing clinical information system in use at the bedside in and Intensive Care Unit[12]. Ploy Tangtulyangkul et al. developed an intelligent integrated query system. It provided information from local veterinary clinical records and supplemented with information from external resources[13].

Patient Data are critical in healthcare domain. So they should be secure, consistent and coded for the secure transfer from one potential user to another. SNOMED CT (Systematized Nomenclature of Medicine -- Clinical Terms) is a standardized reference terminology that consists of millions of SNOMED CT concepts with SNOMED CT codes[14] (http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html). Saman Hina, et al. described the extraction of natural language concepts from free-text discharge summary reports and mapping with SNOMED CT codes[15].

Clinical information is often coded using different terminologies. So it is not interoperable. Li Zhou, et al. developed a general natural language processing (NLP) system, called Medical Text Extraction, Reasoning and Mapping System (MTERMS). It encodes clinical text using different terminologies and simultaneously establishes dynamic mappings between them[16].

Clinical Case Statistical Analysis

It is a tedious and time consuming task for scenario specific information retrieval from an extensive electronic health record (EHR). Anis Yousefi, et al. developed a scenario-oriented information extraction model from electronic health record. It models the relationship between diseases, symptoms, signs and other clinical information as a graph and applies concept lattice analysis to extract all possible diagnostic hypotheses related to a specific scenario[17]. In traditional Chinese medicine (TCM), clinical cases are viewed as semi-structured text, which is between free text and structured text. Their characteristics are lack for grammar, having no strict format, and even uncompleted sentences. But, clinical cases are an important knowledge sources, the knowledge acquisitions from which are going urgently for inheriting TCM. Zhang Huan-sheng, et al. proposed a new machine learning method for the information extraction TCM clinical cases[18].

Drug Testing Analysis

With the rapidly growing use of electronic health records, the possibility of large-scale clinical information extraction has drawn much attention. Eiji Aramakia, et al. extracted adverse drug events and effects from clinical records[19]. Hua Xu, et al. developed a medication information extraction system for clinical narratives[20]. Sunghwan Sohn, et al. extracted physician-asserted drug side effects from electronic medical record clinical narratives[21]. Sunghwan Sohn, et al.developed a Medication Extraction and Normalization system to extract comprehensive medication information from clinical notes and to normalize it to the most appropriate RxNorm concept unique identifier (RxCUI) [22].

Clinical Decision Support

Clinical decision support systems are information technology-based systems designed to improve clinical decision-making. It is crucial for making correct clinical decisions that provides a comprehensive set of relevant information at the point and time it is needed. The computerized clinical decision support can aid decision making of health care providers. Demner-Fushman, et al. examined the role of natural language processing (NLP) in point of care decision support. They discussed the evolution of clinical NLP from early innovation to stable research at major clinical centers[23]. Mi-Young Kim, et al. explored methods for effectively extracting information from clinical narratives, which are interviews transcribed by nurses during phone conversations. The clinical data include variety of noise. They proposed biomedical term detection/normalization method and dependency path-based filtering method to filter the noise[24]. Buzhou Tang, et al. developed a comprehensive temporal information extraction system that can identify events, temporal expressions, and their temporal relations in clinical text. They developed a rule-based system in Python to extract temporal expressions, normalize their values, and identify the modifier attributes simultaneously based on predefined regular expressions[25]. Guilan Konga, et al. provided a literature review in clinical decision support systems (CDSSs) with a focus on the way knowledge bases are constructed, and how inference mechanisms and group decision making methods are used in CDSSs[26]. Computerized clinical decision support systems are information technology-based systems designed to improve clinical decision-making. Decision support systems should be rigorously evaluated before widespread dissemination into clinical practice. R Brian Haynes, et al. examined the effects of computerized clinical decision support systems on practitioner performance[27].

Technologies Used in Information Extraction for Clinical Narrative Text

There are many artificial intelligent algorithm used to implement information extraction from clinical narrative text. The common used technologies are Hidden Markov Models (HMM), Support Vector Machine (SVM), Ontology, Semi-Markov Modeling, Bayesian networks and Bayes Classifier. In this section, we mainly introduce the implication of HMM, SVM, Ontology and other combined method used in information extraction from clinical narrative text.

Hidden Markov Models (HMM)

Hidden Markov Models (HMM) is a good statistical machine learning method that has been used in information extraction and natural language domains, especially in information extraction for clinical narrative text.

Electronic patient information systems include numerous functionalities to support clinical judgment and decision-making, but their capabilities to analyze free-text narratives are limited. H. Suominen, et al. applied HMM to divide Finnish intensive care nursing notes into topically coherent segments and assigned a topic label to each segment[28]. However, the methods proposed by H. Suominen, et al. required annotated data to adapt to different information needs and have limited applicability to texts with short segment length. Filip Ginter, et al. introduced an unsupervised method based on a combination of HMM and latent semantic analysis[29]. It allows the topics of interest to be defined freely, without the need for data annotation, and can identify short segments. The method is evaluated on intensive care nursing narratives and motivated by information needs in this domain.

Berry de Bruijn, et al. proposed machine-learned solutions for three stages of clinical information extraction[30]. In the concepts task, concept tagging is carried out using a discriminative semi-Markov HMM, and trained using passive-aggressive (PA) online updates. Semi-Markov models are Hidden Markov Models that tag multi-token spans of text, as opposed to single tokens.

Support Vector Machine (SVM)

Extraction information from clinical text can be approached through a classification problem where text is split into tokens and grouped into several classes. HMM is a popular method for this task, but it cannot handle multiple tokens with attribute[31]. Support Vector Machine (SVM) is a new machine learning method which is developed on the basic of statistical learning theory and structural risk minimization [32]. It is one of successful machine learning methods in the extraction of information. SVM has achieved various classification tasks in information extraction, especially in information extraction for clinical narrative text.

Zhou X, et al. developed a clinical reference information model and physical data model to manage the various information entities and their relationships in traditional Chinese medicine clinical data[33]. They used support vector machine, decision tree and Bayesian network, to discover the knowledge of syndrome differentiation. Bryan Rink, et al. proposed a SVW approach to discover relations between medical problems, treatments, and tests mentioned in electronic medical records[34]. A single SVM classifier was used to identify relations between concepts and to assign their semantic type. Angus Roberts, et al. have designed and implemented a SVM based system for relation extraction[35]. They trained and tested it on a corpus of oncology narratives hand annotated with clinically important relationships.

Ontology

Ontology-based information extraction (OBIE) has recently emerged as a subfield of Information Extraction (IE). It is based on the use of ontology to guide the information extraction process. Ontology is defined as a formal and explicit specification of a shared conceptualization[36]. SVM has achieved various classification tasks in information extraction, especially in information extraction for clinical narrative text.

Jessica D. Tenenbaum, et al. described the development and use of the Biomedical Resource Ontology (BRO) to enable resource discovery in clinical and translational research[37]. Bouamrane, et al. introduced the development of ontology for a preoperative risk assessment clinical decision system[38]. The ontology is combined within a preoperative risk assessment software system in order to provide a number of clinical decision support functionalities, including risk assessment, recommended tests and recommended clinical precaution protocols. Using Semantic-Web specifications to represent temporal information in clinical narratives is an important step for temporal reasoning and answering time-oriented queries. Cui Tao, et al. developed Clinical Narrative Temporal Relation ontology[39]. Using this ontology, temporal information in clinical narratives can be represented as Resource Description Framework triples. More temporal information and relations can then be inferred by Semantic-Web based reasoning tools. Zhang Huan-sheng, et al. proposed a new machine learning method for the information extraction TCM clinical cases based on domain ontology. This method is an interactive process with a domain expert[18]. David Riañoa, et al. introduced an ontology-based personalization of health-care knowledge to support clinical decisions for chronically ill patients. It implemented two personalization processes and a decision support tool[40]. The ontology is also used as the knowledge base of a decision support tool that helps health-care professionals to detect anomalous circumstances such as wrong diagnoses, missing information, or unobserved related diseases. Song D, et al. proposed a semi-automatic tool for document annotation with Semantic Web ontology[41]. It supports the creation or deletion of ontology instances for any document fragment, linking or disconnecting instances with the properties in the ontology, and also enables automatic annotation by connecting to the National Center for Biomedical Ontology (NCBO) annotator and cTAKES.

Bayes Classifier and other Combing Technology

Bayesian decision is a probability theory as the framework for making decisions under uncertainty. In classification, Bayes' rule is used to calculate the probabilities of the classes[42]. Bayes Classifier is a statistical approach based on the tradeoffs between the classification decisions using probability and the costs of those decisions. It assumes that the problem is given in probabilistic terms and the necessary values are already given. Then it decides on the best class that gives the minimum error with the given example. Bayes Classifier has achieved various classification tasks in information extraction, especially in information extraction for clinical narrative text.

Guergana K. Savova, et al. describes the Mayo Clinic information extraction system for the clinical domain which was developed using IBM's Unstructured Information Management Architecture (UIMA) [43]. The system is being used to process and extract information from free-text clinical notes. In this system, the Machine Learning named entities annotator is based on a Naive Bayes classifier trained on a combination of the UMLS entry terms and the Mayo Synonym Clusters. Each diagnostic statement is represented as a bag-of-words and used as a training sample for generating a Naive Bayes classifier which assigns Mayo Synonym Clusters identifiers to noun phrases identified in the text of clinical notes. Clinical Practice Guidelines (CPGs) play an important role in improving quality of care and patient outcomes. Hai-Tao Zheng, et al. explored a Bayesian Network (BN) approach for representing the uncertainty in CPGs based on ontology. Using this representation, the effect of an activity on the whole clinical process can be evaluated, which can help doctors judge the risk of uncertainty for other activities when making a decision[44].

Conclusions and Future Challenges

In this paper, we reviewed the current development and technology in the information extraction for clinical narrative text. For all of the review of current development, application and technology of information extraction from clinical narrative text that we represented in this special issue, there are still some areas and publications that are missing. The potential uses of information extraction from clinical narrative text are numerous.

The current NLP systems have implemented information extraction from electronic medical record clinical free-text based on some open-source natural language processing system. The further development for clinical NLP systems should be large-scale intelligent information extraction from clinical narrative. We should use expert reasoning to implement the cleverest diagnosticians. The decision-support based on information extraction from clinical narrative text is the most popular implication in the future. It is more difficult for computer based information extraction systems to offer acceptable recommendations for patient therapy than to give a case for diagnosis.

ACKNOWLEDGMENT

The author would like to thank Pro. Arvind K. Bansal for his valuable discussions and support. This work was supported by the National Natural Science Foundation of China (grant no. 61273180), the Natural Science Foundation of Shandong Province, China (grant nos. ZR2009GQ013，ZR2011FQ005，ZR2011FL010), the Project of Shandong Province Higher Educational Science and Technology Program (grant no. J14LN74), and the International Cooperation Program for Excellent Lecturers of 2012 by Shandong Provincial Education Department, China.

REFERENCES

[1] D.Manning, C. P. Raghavan. An Introduction to Information Retrieval. 2009, Cambridge: Cambridge University Press.

[2] Naizhou Zhang, Shijun Li, Zhuo Zhang, Wei Cao. WPBL: A Webpage Block Labeling Based Approach for Web Information Extraction. Journal of Information & Computational Science 7: 1 (2010), pp. 49-55.

[3] S. M. Meystre,G. K. Savova, K. C. Kipper-Schuler, J. F. Hurdle. Extracting Information from Textual Documents in the Electronic Health Record: A Review of Recent Research. IMIA and Schattauer GmbH 2008, pp. 129-144.

[4] Cohen AM, Hersh WR. A survey of current work in biomedical text mining. Brief Bioinform. 2005 6(1), pp. 57-71.

[5] Zweigenbaum P, Demner-Fushman D, Yu H, Cohen KB. Frontiers of biomedical text mining: current progress. Brief Bioinform. 2007, pp. 358-75.

[6] http://www.cs.nyu.edu/cs/projects/lsp/

[7] http://www.medlingmap.org/taxonomy/term/80

[8] http://knowledgemap.mc.vanderbilt.edu/research/content/about-us

[9] http://www.nlm.nih.gov/research/umls/quickstart.html

[10] http://en.wikipedia.org/wiki/CTAKES

[11] Filip Ginter, Hanna Suominen, Sampo Pyysalo, Tapio Salakoski. Mayo clinical text analysis and knowledge extraction system (CTAKES) architecture, component evaluation and applications. J Am Med Inform Assoc.2010,17(5) pp.507–513.

[12] Jon Patrick, Min Li. Intelligent Clinical Notes System An information retrieval and information extraction system for Clinical Notes. Healthcom 2009, pp.107-115.

[13] Ploy Tangtulyangkul, Chun Che Fung . An intelligent integrated querying system for free-form information extraction from veterinary clinical records. Proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, 12-15 July 2009, pp. 3430-3435.

[14] http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html

[15] Saman Hina, Eric Atwell, Owen Johnson. Secure Information Extraction from Clinical Documents Using SNOMED CT Gazetteer and Natural Language Processing. Internet Technology and Secured Transactions (ICITST), 2010 International Conference, pp. 1-5.

[16] Li Zhou, Joseph M Plasek, Lisa Li Zhou, Joseph M Plasek, Lisa M Mahoney,etc.Using Medical Text Extraction, Reasoning and Mapping System (MTERMS) to Process Medication Information in Outpatient Clinical Notes.AMIA Annu Symp Proc. 2011,pp.1639–1648.

[17] Anis Yousefi, Negin Mastouri, Kamran Sartipi. Scenario-Oriented Information Extraction from Electronic Health Record. Computer-Based Medical Systems, 2009. CBMS 2009, pp.1-5

[18] Zhang Huan-sheng, Zhang De-zheng,Chen Xi-xuan. A New knowledge acquisition Method from TCM Clinical Cases based on Information Extraction Proceedings of the 7thWorld Congress on Intelligent Control and Automation, 2008, ,pp.7956-7959.

[19] Eiji Aramakia, Yasuhide Miurab, Masatsugu Tonoikeb, etc. Extraction of Adverse Drug Effects from Clinical Records. MEDINFO 2010, pp.739-743.

[20] Xu H, Stenner SP, Doan S, Johnson KB, Waitman LR, Denny JC. MedEx: a medication information extraction system for clinical narratives. J Am Med Inform Assoc 2010, 17(1), pp.19-24.

[21] Sunghwan Sohn, Jean-Pierre A Kocher, Christopher G Chute, Guergana K Savova. Drug side effect extraction from clinical narratives of psychiatry and psychology patients. J Am Med Inform Assoc.2011,18, pp.i144-i149.

[22] Sunghwan Sohn,Cheryl Clark, Scott R Halgrim, Sean P Murphy, Christopher G Chute, Hongfang Liu.MedXN: an open source medication extraction and normalization tool for clinical text. J Am Med Inform Assoc 2014 0, pp:1-8.

[23] Demner-Fushman D, Chapman WW, McDonald CJ. What can natural language processing do for clinical decision support? J Biomed Inform. 2009, 42(5):760–72.

[24] Mi-Young Kim, Ying Xu, Osmar Zaiane and Randy Goebel Patient Information Extraction in Noisy Tele-health Texts. 2013 IEEE International Conference on Bioinformatics and Biomedicine, 2013, pp.326-329.

[25] Buzhou Tang, Yonghui Wu, Min Jiang,etc. A hybrid system for temporal information extraction from clinical text. J Am Med Inform Assoc 2013;20(5),pp.828-835.

[26] Guilan Konga, Dong-Ling Xua, Jian-Bo Yanga. Clinical Decision Support Systems: A Review on Knowledge Representation and Inference under Uncertainties. International Journal of Computational Intelligence Systems. 2008,1(2), pp. 159-167.

[27] R Brian Haynes, Nancy L Wilczynski. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: methods of a decision-maker-researcher partnership systematic review. Implementation Science 2010, 5, pp. 5-12.

[28] H. Suominen, S. Pyysalo, F. Ginter, T. Salakoski, Automated text segmentation and topic labeling of clinical narratives. Proceedings of Louhi'08, TUCS, 2008, pp. 99-103.

[29] Filip Ginter, Hanna Suominen, Sampo Pyysalo, Tapio Salakoski. Combining hidden Markov models and latent semantic analysis for topic segmentation and labeling: Method and clinical application. international journal of medical informatics. 2009, 78 (12), pp. e1–e6.

[30] de Bruijn B,Cherry C, Kiritchenko S, Martin J,Zhu X. Machine-learned solutions for three stages of clinical information extraction: the state of the art at i2b2 2010. J Am Med Inform Assoc 2011, 18(5), pp.557-562.

[31] Kuspriyanto, Oerip S Santoso, Dwi H Widyantoro, Husni S Sastramihardja. Performance Evaluation of SVM-Based Information Extraction using τ Margin Values. International Journal on Electrical Engineering and Informatics. 2010, 2(4) pp. 256-265.

[32] Shifei Ding, Junzhao Yu, Bingjuan Qi, Huajuan Huang. An overview on twin support vector machines. Artif Intell Rev (2014) 42: 245-252.

[33] Zhou X, Chen S, Liu B, Zhang R, Wang Y, Li P, Guo Y,Zhang H, Gao Z,Yan X. Development of traditional Chinese medicine clinical data warehouse for medical knowledge discovery and decision support. Artificial Intelligence in Medicine. 2010, 48(2–3), pp.139–152.

[34] Bryan Rink, Sanda Harabagiu, Kirk Roberts. Automatic extraction of relations between medical concepts in clinical texts. J Am Med Inform Assoc 2011,18, pp.594-600.

[35] Angus Roberts, Robert Gaizauskas, Mark Hepple, Yikun Guo. Mining clinical relationships from patient narratives. BMC Bioinformatics 2008, 9(Suppl 11):S3.

[36] Daya C. Wimalasuriya, Dejing Dou. Ontology-based information extraction: An introduction and a survey of current approaches. Journal of Information Science. 2010, 36, pp.306-323.

[37] Jessica D. Tenenbaum,Patricia L. Whetzel,Kent Anderson,etc. The Biomedical Resource Ontology (BRO) to enable resource discovery in clinical and translational research. Journal of Biomedical Informatics. 2011, 44(1), pp. 137–145.

[38] Bouamrane, M.-M., Rector, A., and Hurrell, M. Development of an ontology for a preoperative risk assessment clinical decision support system. In: 22nd IEEE International Symposium on Computer-Based Medical Systems (CBMS 2009), 2009, pp. 1-6.

[39] Cui Tao, Wei-Qi Wei, Harold R. Solbrig, Guergana Savova, Christopher G. Chute. CNTRO: A Semantic Web Ontology for Temporal Relation Inferencing in Clinical Narratives. AMIA Annu Symp Proc. 2010; 2010:pp. 787–791.

[40] David Riañoa, Francis Reala, Fabio Campanab, etc. An ontology-based personalization of health-care knowledge to support clinical decisions for chronically ill patients. Journal of Biomedical Informatics. 2012, 45(3), pp. 429–446.

[41] Song D, Chute CG, Tao C.Semantator: annotating clinical narratives with semantic web ontologies. AMIA Jt Summits Transl Sci Proc. 2012:20-9.

[42] Ethem Alpaydin. Introduction to Machine Learning. The MIT Press,Cambridge, Massachusetts London, England. 2004,pp.47-59.

[43] Guergana K. Savova, Karin Kipper-Schuler, James D. Buntrock, and Christopher G. Chute. UIMA-based Clinical Information Extraction System. Towards Enhanced Interoperability for Large HLT Systems: UIMA for NLP.2008. pp. 39-42.

[44] Hai-Tao Zheng, Bo-Yeong Kang, and Hong-Gee Kim. An Ontology-Based Bayesian Network Approach for Representing Uncertainty in Clinical Practice Guidelines. Reasoning for the semantic Web 1 2008-Springer, pp. 161-173.

Yan Ge was born in Taian, China, in 1975. She received the BS degree in information engineering from Beijing Jiaotong University, Beijing, China, 1996. She received the MS degree in computer application from Shandong University of Science and Technology, Qingdao, China, 2002, and the PhD degree in computer science from Ocean University of China, Qingdao, China, 2005. She is currently an associate professor at school of information science and technology, Qingdao University of Science and Technology. She was a visiting scholar at Department of Computer Science, Kent State Univerity, Ohio, USA from December 2013 to December 2014. Her research interests include intelligent information system, combinatorial optimization, and health informatics.

An Evaluation with Web Developers of Capturing User Interaction with Rich Internet Applications for Usability Evaluation

Luciano T. E. Pansanato*1, Adriano Rivolli2, Douglas F. Pereira3

Department of Computing, Federal University of Technology – Paraná, Av. Alberto Carazzai, 1640, 86300 000, Cornélio Procópio, PR, Brazil

*1luciano@utfpr.edu.br; 2rivolli@utfpr.edu.br; 3doug_felpe@hotmail.com

Abstract

The user interaction logging with a Rich Internet Applications can provide valuable information for usability evaluation and user behavior study. However, current research has focused only on issues of implementation and validation of tools for capturing user interaction instead tool usage by web developers. This work presents an evaluation through user testing of a tool called WAUTT for capturing user interaction with Rich Internet Applications. The main goal of this evaluation is to investigate the use of a tool for capturing user interaction with web developers. The results indicate that the participants have good understanding of the tool, found easiness to operate the tool for capturing and analyzing the user interaction, consider its future re-use, and they felt satisfied with the experience. The results also assisted for identifying the user interaction data considered more useful to web developers.

Keywords

Usability Evaluation; User Interaction Logging; Log File Analysis; Rich Internet Application; User Testing; Web Developer Usage

Introduction

Web applications have evolved from static pages or pages with content generated dynamically to applications that provide sophisticated interfaces. The term Rich Internet Application (RIA) refers to a family of heterogeneous solutions with the objective to add new capabilities to the conventional Web [1]. RIA technologies have been increasingly adopted by user-centric web applications that demand advanced presentation and interactivity resources.

The popularity of RIAs comes from their capability to promote presentation and interaction of quality. A traditional web application interface consists of multiple pages, refreshed at each user interaction. In RIAs, the interface can be a single page comprising subpages that manages all the users' interactions, like in a desktop application. This paradigm avoids full-page refreshes at each interaction and allows applications to independently load, display and update individual page elements [1].

RIAs pose a problem whenever it is necessary to get information on how users actually use them. Data on user interaction that can be stored in the server are not enough to extract detailed information on the application's real use. Regarding usability evaluation, the installation of additional software in the client's machine may be necessary to capture the entire user interaction. However, a user may not be willing to install software of this nature in his/her equipment for safety or performance reasons.

The Web Application Usage Tracking Tool, WAUTT [2], is a tool used to capture detailed information at the level of the page elements and events associated to user interaction with a RIA. The idea underlying WAUTT is to use a JavaScript code capable of capturing user interaction in the client and registering the obtained data in a server for further analysis and interpretation.

Data from user interaction with a RIA provide valuable information for usability evaluation and user behavior study. However, substantial web developers can have some issues in using a tool to capture the interaction as well

2324-7037/15/02 051-10

doi: 10.12783/ijcsa.2015.0402.02

as the captured information. Current research has focused only on issues of implementation and validation of tools for capturing user interaction with web applications. This work presents an evaluation of WAUTT with web developers to investigate the use of a tool for capturing user interaction for getting knowledge about tool usage. The knowledge obtained will address future work to improve tools for capturing user interaction with web applications and web development practices.

This work is organized as follows. Next section describes related works according to different approaches adopted to capture user interaction and it highlighs at the end the lack of studies with web developers on the use of tools for capturing user interaction. Subsequently, WAUTT is presented focusing on its capture approach and data that are captured. Thereafter, the evaluation proposed is presented which is based on user testing with web developers. The results of this evaluation are presented subsequently. The final section presents the conclusions and future works.

Related Work

The capturing of user interaction is highly used as a way to get data for usability evaluation and to study user behavior on the Web [3] [4] [5] [6]. Data on user interaction can also be used to justify the realization of some maintenance and/or alteration in the web application. For instance, a careful analysis of the data can identify less and more visited areas and most frequently used functionalities.

The capturing of user interaction can be done via web server. Generally, the scope of the interaction captured through the web server is limited since only visited pages are registered instead of all users' interactions with a RIA. For instance, the ErgoMonitor environment [7] carries out an analysis using selective data collected from the server's access log related to the interactions realized by the user and the web interface. However, this analysis of the data stored in the server access log alone is not sufficient for applications which have great parts of the interaction realized in the client.

The AWUSA [8] also uses data from the server but it takes into consideration pages and links categories determined from an analysis of the application's HTML code. These categories are used to increase the level of detail about the user interaction information. In addition, to meet its objective, this tool also used data from the application log (transaction data) and other additional log files. Thus, the need for more information is clear, even more when a RIA is considered.

An alternative to capturing user interaction is to use a proxy server between the client and the web server. This alternative, however, requires that the user makes a configuration in his/her browser to use the proxy and later to undo that configuration. The use of a transparent proxy could avoid this configuration, but this type of proxy is limited to the traffic of one network only. The approach adopted by WebQuilt [9] differs from the traditional HTTP proxy, since it is based on the URL which redirected all the links in a way that the URLs themselves point to the proxy and the intended destination is coded inside the URL string. Thus, the user does not have to do a configuration in his/her browser to send all requisitions to the proxy. However, the WebQuilt does not capture the interaction in the client-side; only information about the visited pages is available to the proxy. Especially, the focus of WebQuilt is information visualization technique applied to the data.

The approach adopted by UsaProxy [10], a HTTP proxy, consists of intercepting the traffic and registering data on any request sent to the web server as well as the answer that the web server sends back to the client. UsaProxy adds a JavaScript code to each requested page before delivering it to the client. This code captures the user interaction with the page such as mouse move and click, page scroll, window resize, focus change, and key press. Data are sent periodically to the proxy to be stored in a log file. As the inclusion of the JavaScript in the page only takes place when the page content is a HTML text (text/html), the UsaProxy approach does not work with the HyperText Transfer Protocol Secure (HTTPS), since the content is encrypted.

WAUTER [11], WebHint [12], and Web Usability Probe [13] are similar to UsaProxy. Actually, the WebHint uses the UsaProxy to capture user interaction. WAUTER also intercepts the HTML code sent to the client and inserts a JavaScript to capture user interaction. The difference is that the proxy is installed in the client's machine to overcome the practical question of web server access. Thus, questions about the need to install software in the user's equipment and the working with HTTPS still remain unanswered. In addition, WALTER and WebHint do not capture data related to the attributes of HTML elements and style sheets (CSS).

The capturing of the user interaction can also be done through the client. For instance, the WebTracker [14] is used to capture user interaction with the browser. This tool captures the selections made in the menus and toolbars (buttons), and the actions realized by the keyboard. These interactions are associated with the open page in the browser plus date and time. The scope of the interaction captured by the WebTracker is limited to the browser's functionalities and visited pages; mouse clicks are only captured when a link is selected on the open page in the browser. Another tool, called Wrapper [15], registers the user interactions with the browser and several other actions such as adding to favorites, copy, print, save and scroll bar use.

The problem with these tools (that capture through the client) lies in the granularity level of the obtained information which is too high for RIAs, considering that the interest is to get detailed information at the level of the page elements and events associated with user interaction. Even when the user interaction is obtained through a program installed in the user's equipment, such as the WebLogger [16], which captures page scroll and mouse move, it is necessary to process the existence of a click in the coordinates associated with the desired element to obtain data on the page elements that were used. Ultimately, the need to install these applications in the client's equipment turns into a highly intrusive process for the user.

WebRemUSINE [17], WELFIT [18] and USABILICS [19] can also capture user interaction through the client. These tools prevent the installation of software in the user's machine since they use a JavaScript code inserted in the pages by the application developer himself. In WebRemUSINE, the script does the capture and sends the data to an applet Java responsible for forwarding to the web server at the end of the session (indicated explicitly by the user). In WELFIT and USABILICS, data are sent by the JavaScript code itself.

Information captured by the WebRemUSINE, WELFIT and USABILICS provides lots of details about the user interaction with the application. In general, the JavaScript code redefines the event handlers to capture the user interaction. An event handler is a part of a code associated with the interaction object. Captured data are related to interaction objects that, when selected or manipulated by the user, triggering events corresponding to the actions. WebRemUSINE, WELFIT and USABILICS are limited to standard event handlers of the JavaScript language and do not capture events defined by the user (application developer). Moreover, the WELFIT does not capture data related to the style sheets attributes (CSS).

This work uses the version of the WAUTT which also captures user interaction through the client. As WebRemUSINE, WELFIT and USABILICS, the capture occurs through a JavaScript code inserted in the RIA's pages. However, besides capturing interaction related to standard events, WAUTT also register other events to be captured and that were created by the developer to deal with specific interactions or are defined by some JavaScript library used in the RIA.

Despite advances of the tools for capturing user interaction with rich internet applications, studies on the use of such tools by web developers are not available. An evaluation with web developers must consider an investigation about tool usage and the information is considered useful for usability and/or behavior analyses.

Capturing User Interaction with RIA Using WAUTT

The use of server log files to support the evaluation of web systems is a convenient and reliable strategy [5]. In general, a server log file entry contains client IP address, request date/time, requested page (URL), HTTP code, requisition protocol, and size of the requested file. The server log files have a limitation because they contain information about the pages (URLs) accessed by the user and they do not offer detailed data about the real user interaction with a RIA. Data on user interaction that can be stored in the web server are not enough to extract detailed information on the RIA's real use.

The WAUTT [2] provides the "wautt.js" script for capturing user interaction. The technology adopted for the RIA is transparent to WAUTT, since the developer inserts a HTML code in the pages, which includes the remote file "wautt.js" available in the WAUTT server. Therefore, all logic that involves the capture of the interaction and data forwarding to the WAUTT server is centralized in a single place and only referenced in the RIA. This approach is similar to that adopted by Google Analytics (www.google.com/analytics/) and other tools that capture user interaction through the client [17] [18] [19].

The information captured by WAUTT provides lots of details about the user interaction with a RIA. The collected data are stored in a relational database and can be accessed via the administrative interface, as shown in FIGURE 1. Each user interaction registered can be associated with the following data groups:

- Session data. This data group includes the session identifier, browser and operational system (name and version), and the IP address. A session involves a sequence of actions of the same user during the use of the monitored application. A session is created when a user opens the browser and loads the application, and is concluded when the browser is closed.
- Browsing data. This data group includes the page title, URL, parameters, request method (GET or POST), and page size. Browsing involves the sequence of the pages loaded during the use of the application. While in some applications the user interacts with a single page without refreshing it completely, in other applications several pages are loaded.
- Event data. This data group includes specific details of the action performed by the user in the interface and varies according to the event type. These data allow identifying which (event type and details) and when (time) the interaction took place.
- DOM data. This data group includes the name of the page element and its attributes. For instance, the attributes are common to all elements such as id, class, title, and specific attributes (according to element type) such as type, value and name of an element of the type input.
- Task data. This data group refers to the active task, if the resource of distinct tasks is being used. This information allows the realization of the analysis of the captured data according to the task.

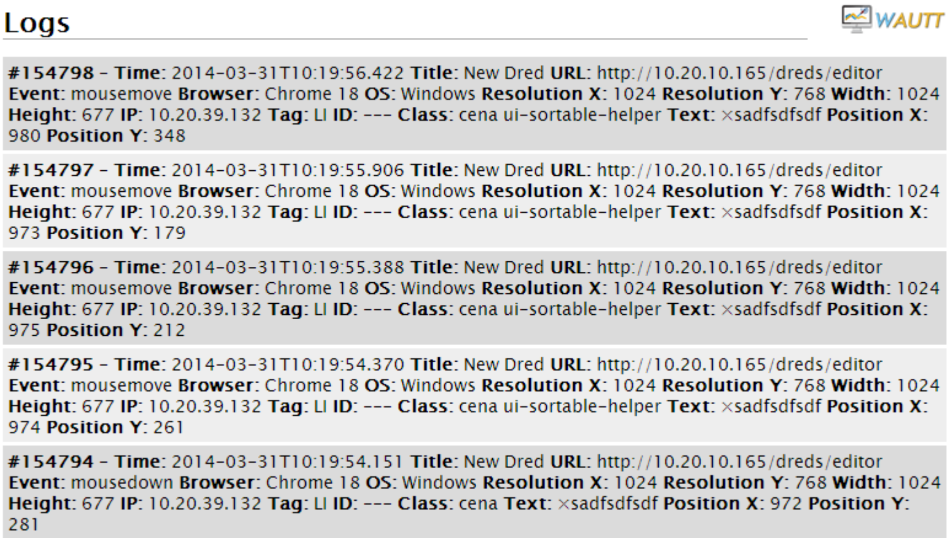

FIGURE 1 EXAMPLE OF DATA CAPTURED BY WAUTT

An advantage of capturing user interaction is the possibility to collect large volumes of data automatically. However, adequate tools are needed to help in the quantitative analysis and interpretation of the collected data. After using WAUTT to capture user interaction, the collected data are made available in the WAUTT server. WAUTT summarizes some basic information related to the captured data: session details, number of open pages by session, graph of data by period of time, total of data by event, page element, task and session. In addition, WAUTT also allows exporting data in three different formats (JSON, XML e CSV) to be submitted to analyses by specific tools.

Evaluation with Web Developers

User testing is a type of applied experimentation in which developers check that the system being developed is usable by the intended user population for their tasks [5]. In the case of WAUTT, its intended users are web developers that work with development of RIAs.

The evaluation with web developers was carried out in agreement with the recommendations of ISO/IEC standards for software product evaluation, which divides the process into four stages: specification, planning, pilot evaluation, and evaluation. Next subsections describe this evaluation with web developers. The results are presented in the next section.

Specification

In specification stage, the scope and objective of the evaluation were defined. The scope of the evaluation was the WAUTT interface and its functionalities which implement the capture and analysis of user interaction with RIA. The goals were to identify preferences in using WAUTT and whether web developers could have difficulties to use the captured information for usability evaluation purposes. In addition, an underlying objective was to identify other user interaction data that tools could collect to support RIA development and evaluation.

Planning

In planning stage, an initial plan was developed to carry out the evaluation. The work to produce and execute the user testing was estimated from this initial plan which includes the following:

- Definition of the target group of users (participants);
- Set of tasks to be completed by participants;
- Description of the material to be produced;
- Guide for conducting the evaluation.

The target group of users considered to participate of the evaluation is composed by developers with experience in development of RIAs and academic background in computing (graduate or undergraduate). In addition, a test monitor was involved to conduct the evaluation. The participants were recruited from software development companies using unintentional sampling. In summary, the process of selecting participants was non-probability sampling for convenience.

The set of tasks to be completed by participants comprises four tasks regarding using the tool for capturing user interaction with a RIA:

- Task 1: make the registration in WAUTT server of the target application (free choice) and download the script for capturing user interaction. The participant was informed that the interface of the target application must be rich in terms of interaction elements and events that can be captured.
- Task 2: include the script downloaded in the pages of the target application.
- Task 3: browse the target application to collect data concerning user interaction.
- Task 4: use the functionalities provided by WAUTT to view and analyze the details of the collected data through data filtering and reporting.

Tasks 1 and 2 refer to the preparation of a RIA to have the user interaction captured by WAUTT. These tasks are important to investigate the understanding of web developers (participants) concerning the tool principle for capturing user interaction. In short, the preparation consisted of the registration of the target application in the WAUTT server and the inclusion of instructions in the HTML code of the pages is similar to the following:

```
<script src="http://lab.utfpr.edu.br/wautt/js/start.php?appid=2a7725e1a20f84b538e6b8a1e238b95c">
</script>
```

Task 3 is useful for web developers to understand what types of interaction can be captured. Participants followed the generation of log file entries concomitantly with its interaction with the RIA (through functionality provided by WAUTT). Task 4 refers to the understanding in using collected data to get some useful information about the RIA usage. Web developers should be able to relate collected data with some evidence of problems or behaviors (desired or not).

The tasks were presented to the participants in agreement with the concept of simulated work task situation [20]. A simulated work task situation is an open description of the context/scenario of a given situation. It then works as the trigger of the participant's goal and the base for judgment. The objective is to ensure the largest possible realism by the involvement of web developers who, based on the simulated work task situation, develop individual and subjective interpretations of the user tasks.

The material prepared to be distributed for participants includes: consent form to participate in the test; description of the WAUTT; survey questionnaire; description of user tasks (scenarios); and post-test questionnaire.

The primary purpose of the consent form is to let the participants know the goals of the test and to explain what will happen to the findings as well as guarantee privacy and confidentiality of information. In addition, this consent form also works as an agreement between evaluator and participants to confirm the professional commitment of serious participation in the evaluation [5].

The descriptive material relating to WAUTT enabled participants familiar with this tool and its purpose. A brief presentation was also offered to the participants for introducing an overview of the proposed user testing.

The survey questionnaire was designed for characterizing the participants. The questions was used to collect basic demographic information (for instance, gender and age) and details about participant's experience regarding web development, integrated development environments, database management systems, and web analytics techniques.

The post-test questionnaire was designed for collecting subjective judgments on WAUTT usage and it has five questions in Likert scale, one multiple choice question, and two open questions. Likert is a bipolar scaling method, measuring either positive or negative response to a statement. A seven-point scale was used in the post-test questionnaire (1: very strongly disagree; 2: strongly disagree; 3: disagree; 4: neutral; 5: agree; 6: strongly agree; 7: very strongly agree).

A guide with precise instructions for all procedures needed to user testing was also designed. This guide aimed to minimize the influence on task performance of different procedures in conducting several sessions of user testing.

Pilot Evaluation

In pilot evaluation stage, a complete user testing with two users was carried out to identify problems with the planning [5]. During this stage, the following criteria were analyzed: (a) coverage and quality of the user tasks; and (b) sufficiency and quality of the descriptive material and presentation. At the end of pilot evaluation, only minor problems concerning ambiguity in user task descriptions were encountered.

Evaluation

The user testing was carried out with 30 participants randomly selected in agreement with the target group of users defined in the planning stage. The participants made the tasks in laboratory conditions and in a controlled way (with a test monitor). However, there was no supervision in the sense of aiding the participants to complete the tasks. The chosen environment was a usability laboratory and its choice aimed to minimize the influence of the environment on task performance.

Results and Discussion

This section presents initially the characterization of the participants as potential users of a tool for capturing user interaction. Following, the data collected are analyzed and summarized using descriptive statistics. The results concerning the tasks are first presented, and then the results concerning the post-test questionnaire are presented.

Participants

The survey questionnaire applied at the beginning of the user testing shows that the selected participants have an average of 2.25 years of experience in web developing. The participants (86.7%) have advanced knowledge of HTML and CSS as well as programming languages for web development (73.3%), most of them (79.9%) prefers JSP and ASP. The participants also have intermediate knowledge of integrated development environments (IDE) and database management systems (DBMS) (70% and 63.3% respectively). All the participants have used some methods of web analytics, server log file analysis (80%), page tagging (56.7%), other (16.7%). Therefore, the participants can be considered potential users of tools for capturing user interaction.

Tasks

The analysis of the time to completion tasks may indicate if participants have encountered difficulties for completing the tasks. Time to completion is the calculated amount of time required for any particular task to be completed. Table 1 presents the results concerning each task. Analysis of data dispersion revealed a significant standard deviation (SD) and wide total range – minimum (Min) and maximum (Max) values. The wide variability found can be explained, among other factors, by the use of the concept of simulated work task situation in the task descriptions, which allow the participants to take different paths to complete the task. However, this concept was used to ensure realism and commitment through the participant's involvement in the task.

TABLE 1 TIME TO COMPLETION FOR EACH TASK; VALUES IN MINUTES AND SECONDS

Tasks	Mean	SD	Min	Max
1	2min 05s	0min 58s	1min 12s	4min 03s
2	2min 25s	1min 16s	1min 31s	6min 10s
3	5min 49s	2min 27s	1min 54s	9min 42s
4	3min 52s	1min 26s	2min 27s	6min 55s

The results for Task 2, Task 3 and Task 4 present high variability for time to completion (observed by standard deviation in Table 1). However, the lower variability for Task 1 can be due to characteristics of the task: make the registration and download the script. The participants' behavior tends to be more homogeneous because Task 1 is simple. Dispersion in the values was specially expected in Task 3 and Task 4 because some participants tend to explore all options while others have focus on completing the task. Therefore, the variability for Task 3 and Task 4 is mainly influenced by its exploratory nature: browsing a web application; and using the functionalities provided by a tool.

An outlier participant was identified for Task 1 and Task 2 who also presented high time to completion for Task 3 and Task 4. If the outlier is due to a strange or rare event that never will happen again, the data from that participant could be excluded (for example, due to wrong data). However, because the case of a participant takes more time for completing the tasks which is an event that may occur again in user testing, it is not advisable to exclude the data from the analysis.

The description of the tasks could be adjusted to avoid that high variability through increasing the level of detail of the actions was to be accomplished by the participant. A more homogeneous behavior of the participants is expected when they follow detailed instructions on what to do in the tasks. However, the freedom of exploration was important to obtain subjective judgments of the participant regarding his/her experience. The subjective judgments were obtained from the post-test questionnaire whose results are presented in the sequence.

Post-test Questionnaire

At the end of user testing, a set of questions was presented to the participants to rate their experience. This post-test questionnaire was used to obtain different categories of information: understandability, operability, data analysis functionality, future re-use, and satisfaction. Table 2 presents a summary of the post-test questions and results.

TABLE 2 SUMMARY OF THE POST-TEST QUESTIONS AND RESULTS

Questions	Frequency						
	1	2	3	4	5	6	7
Q1. I found the tool for capturing user interaction ease to understand	-	-	-	3.3%	46.7%	40.0%	10.0%
Q2. I found the tool for capturing user interaction ease of use (operate)	-	-	3.3%	3.3%	36.7%	30.0%	26.7%
Q3. I found the various functionalities in the tool were useful for data analysis	-	-	-	10.0%	26.7%	50.0%	13.3%
Q4. I would like to use the tool for capturing user interaction in the future	3.3%	-	3.3%	23.3%	60.0%	6.7%	3.3%
Q5. I felt very satisfied using the tool for capturing user interaction	3.3%	-	-	13.3%	40.0%	36.7%	6.7%

The most of participants expressed agree (5), strongly agree (6), or very strongly agree (7) for the five questions (96.7%, 93.3%, 90%, 70%, and 83.3% of participants, respectively for questions Q1, Q2, Q3, Q4 and Q5). These results indicate that the participants have good understanding of the tool, found easiness to operate the tool for capturing and analyzing the user interaction, consider its future re-use, and they felt satisfied with the experience. The median for the five questions (5.5, 6.0, 6.0, 5.0, and 5.0, respectively for questions Q1, Q2, Q3, Q4 and Q5) and mode (5 for all questions) also provides support for these results. However, at last one participant (3.3%) did not consider the future re-use of the tool (Q4) and he/she was not satisfied with the experience (Q5).

The multiple choice question of the post-test questionnaire was used for investigating what user interaction data are considered useful for identifying usability problems in rich internet applications. The question allowed the participants to choose several among the following options:

- Page load (and URL);
- Window resize;
- Page scroll;
- Mouse move (and x-y coordinates);
- Mouse click (and x-y coordinates);
- Mouse click in page element (and attributes);
- Key press;
- Focus change in page element (and attributes).

Figure 2 presents a bar chart with the frequency of the answers for the multiple choice question. The options were ranked in descending order by frequency for highlighting which were considered more useful in participants' opinion.

The most chosen options by the participants were the focus change, mouse click in page element, and page load. Those preferences are directly related with the knowledge of the path taken by the user in the interaction with a web application to accomplish a task. That knowledge is important, for instance, to verify if the user uses the sequence of interaction considered ideal by the web developer for using determined functionality. Additionally, a discerning analysis of those data can identify which areas are more or less visited or still which functionalities are used with greater or smaller frequency.

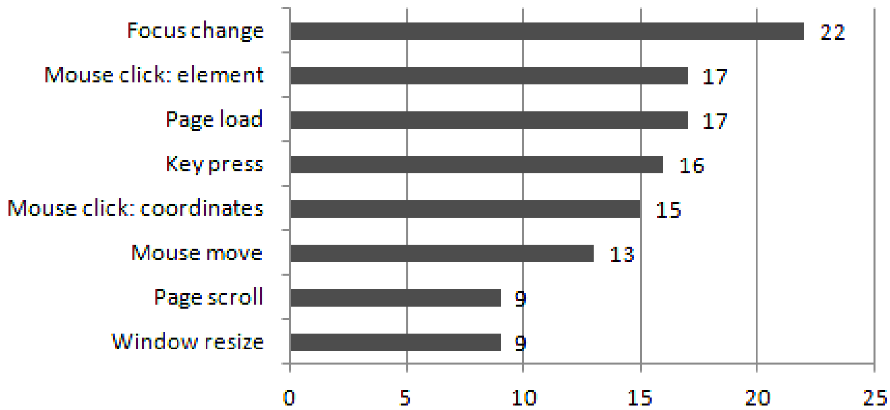

FIGURE 2 FREQUENCY OF ANSWERS FOR THE MULTIPLE CHOICE QUESTION ON USER INTERACTION DATA

An open question was used to identify which other information the participants considers important to be collected through a tool for capturing user interaction. The information more mentioned is related with the time. For instance, page load time (or page speed), time elapsed until the first interaction, permanence time in the page, time between two events, feedback time, response time of the client-server communication, among other measures of time related to rich internet applications.

Participants' concern with time measures is comprehensible, once time is a critical factor for web applications. For example, the users' satisfaction increases according to response time decreases, and small variations are acceptable.

In contrast, great variations can affect the user's behavior.

Another open question approaches which other resources can be used for analysis and interpretation of the data captured from user interaction, besides those provided by the tool used in the evaluation. The main resources mentioned by participants are summarized by the following:

- Identify user interaction sequences which are repeated;
- Provide more graphical resources;
- Search by a user interaction sequence informed;
- Provide customized reports and graphics.

Ultimately, the suggestions presented by participants indicate the application of pattern recognition and information visualization techniques.

Conclusions

The results of this work indicate that the participants have good understanding of the tool, found easiness to operate the tool for capturing and analyzing the user interaction, consider its future re-use, and they felt satisfied with the experience. The results also assisted for identifying the user interaction data considered more useful to web developers. In addition, the knowledge obtained from this evaluation allows addressing future works for the improvement of WAUTT in the sense of supplying time measures related to user interaction with rich internet applications and new resources to aid the analysis of the data. Other future works include the planning and realization of more tests with users to investigate in more depth the user acceptance regarding the use of tools for capturing and analyzing user interaction data in the level of page elements and events.

REFERENCES

[1] Fraternali, P., Rossi, G. and Sánchez-Figueroa, F., "Rich Internet Applications". IEEE Internet Computing, Vol. 14, No. 3, 2010, pp. 9-12.

[2] Rivolli, A., Pereira, D. F. and Pansanato, L. T. E., "Capturing User Interaction with Rich Internet Applications for Usability Evaluation". In Proceedings of the 13th International Conference WWW/Internet. Porto, Portugal, 2014. pp. 3-10.

[3] Jansen, B. J., "Search log analysis: What is it; what's been done; how to do it". Library and Information Science Research, Vol. 28, No. 3, 2006, pp. 407-432.

[4] Kellar, M. and Watters, C., "Studying User Behaviour on the Web: Methods and Challenges". In Proceedings of the ACM Conference on Human Factors in Computing Systems. Portland, USA, 2005.

[5] Sharp, H., Rogers, Y. and Preece, J., Interaction Design: Beyond Human-computer Interaction. John Wiley & Sons, 2nd Edition, 2007.

[6] Turnbull, D., "Methodologies for Understanding Web Use with Logging in Context. Workshop on Logging Traces of Web Activity: The Mechanics of Data Collection". In Proceedings of the International Conference on World Wide Web. Edinburgh, Scotland, 2006.

[7] Schwerz, A. L., Morandini, M. and Silva, S. R., "A Task Model Proposal for Web Sites Usability Evaluation for the ErgoMonitor Environment". In Proceedings of the 12th International Conference on Human-Computer Interaction. Beijing, China, 2007, pp. 1188–1197.

[8] Tiedtke, T., Martin, C. and Gerth, N, "AWUSA a tool for automated website usability analysis". In Proceedings of the 9th International Workshop on the Design, Specification and Verification of Interactive Systems. Rostock, Germany, 2002, pp. 251-266.

[9] Hong, J. I., Heer, J., Waterson, S. and Landay, J. A., "WebQuilt: A proxy-based approach to remote web usability testing". ACM Transactions on Information Systems, Vol. 19, No. 3, 2001, pp. 263-285.

[10] Atterer, R., Wnuk, M. and Schmidt, A., “Knowing the user's every move: user activity tracking for website usability evaluation and implicit interaction”. In Proceedings of the 15th International Conference on World Wide Web. Edinburgh, Scotland, 2006, pp. 203-212.

[11] Balbo, S. Goschnick, S., Tong, D. and Paris, C., “Leading web usability evaluations to WAUTER”. In Proceedings of the 11th Australian World Wide Web Conference. Gold Coast, Australia, 2005.

[12] Vargas, A., Weffers, H. and Rocha, H. V., “A method for remote and semi-automatic usability evaluation of web-based applications through users’ behavior analysis”. In Proceedings of the 7th International Conference on Methods and Techniques in Behavioral Research. Eindhoven, Netherlands, 2010.

[13] Carta, T., Paternò, F., Santana, V. F., “Web Usability Probe: A Tool for Supporting Remote Usability Evaluation of Websites”. In Proceedings of the 13th IFIP TC 13 International Conference on Human-Computer Interaction. Lisbon, Portugal, 2011, pp. 349-357

[14] Choo, C., Detlor, B. and Turnbull, D., “Information Seeking on the Web: An Integrated Model of Browsing and Searching”. First Monday, Vol. 5, No. 2, 2000.

[15] Jansen, B. J., Ramadoss, R., Zhang, M. and Zang, N., “Wrapper: An Application for Evaluating Exploratory Searching Outside of the Lab”. In Proceedings of the 29th Annual International ACM SIGIR Conference on Research & Development on Information Retrieval. Seattle, USA, 2006.

[16] Reeder, R. W., Pirolli, P. and Card, S. K., “WebEyeMapper and WebLogger: tools for analyzing eye tracking data collected in web-use studies”. In CHI '01 Extended Abstracts on Human Factors in Computing Systems. Seattle, Washington, USA, 2001, pp. 19 - 20.

[17] Paganelli, L. and Paternò, F., “Intelligent analysis of user interactions with web applications”. In Proceedings of the 7th International Conference on Intelligent User Interfaces. San Francisco, USA, 2002, pp. 111-118.

[18] Santana, V. F. and Baranauskas, M. C. C., “Summarizing observational client-side data to reveal web usage patterns”. In Proceedings of the 25th ACM Symposium on Applied Computing. Sierre, Switzerland, 2010, pp. 1219-1223.

[19] Vasconcelos, L. and Baldochi Jr, L., “Usabilics: avaliação remota de usabilidade e métricas baseadas na análise de tarefas”. In Proceedings of the 10th Brazilian Symposium on Human Factors in Computing Systems and 5th Latin American Conference on Human-Computer Interaction. Porto de Galinhas, Brazil, 2011, pp. 303-12.

[20] Borlund, P., “The IIR evaluation model: a framework for evaluation of interactive information retrieval systems,” Information Research, Vol. 8, No. 3, 2003.