Persuasive Corpus-Driven Computer-Assisted Language Learning: On the application of Emdros

Johannes Gottschalk[1]

[1] Aalborg University, Fredrik Bajers Vaj 5, 9100, Aalborg Denmark
gottschalk@aau.dk

Abstract. The goal of this paper is to do a short preliminary analysis of the Emdros [4] annotated text database system against one of eight key factors for good and persuasive software. Emdros has been developed by Ulrik Sandborg-Petersen around 2007 and is a continuation of Crist-Jan Doedens'[2]. The analysis will show that Emdros is not a suitable system with regards to corpus-driven computer-assisted e-learning of corpora e.g. containing the Hebrew Bible or the Greek New Testament. This has mainly to do with the fact that Emdros due being very difficult if not impossible to install on a Raspberry Pi mini-computer with the Raspbian standard operating system cannot function in e-learning systems in hard-to-reach areas in countries of the Global South like Madagascar or Ethiopia. Also, community support is not provided. I therefore follow the conclusion that open source systems like e.g. MongoDB are much better suited for this kind of approach.

Keywords: Emdros, database-driven e-learning, MongoDB

1   Introduction

"Let the corpus be your teacher!" This is the slogan this PhD project can best be summarized with. It deals with persuasive database-driven computer-assisted language learning. My PhD work is rooted within computational linguistics, but it is likewise cross-disciplinary. The ideas for a research plan this dissertation deals with are:

1. In what kind of representation does annotated text need to be stored in a database management system? The question is: What is the best database system to save a corpus of the Hebrew Bible, the Greek New Testament or texts of Kaj Munk in to be able to query?
2. Software development as applied to: My own e-learning system the Bible Online Learner Box which runs within a Docker container on several devices from starting with Raspberry Pis over Laptops to large scale server systems based on a RESTful and microservice-based architecture to work in countries of the Global South.
3. A solution for vocabulary learning in e-learning context using word-sense disambiguation and an approach within the theory of RRG [5].

4. An approach for learning to read modern Danish based on the Kaj Munk corpus to support researchers in a persuasive way who do research on the works of the Danish pastor and author Kaj Munk.

In what follows I will discuss the first of these leading questions and my results with respect to the state of the art in this interdisciplinary area of computational linguistics and database design with regards to persuasive design on the macro-level in a database-driven e-learning system.

2 Position with respect to state of the arts results

First it is important to understand based on Date [1] and Sandborg-Petersen [4] that a database is not at all a piece of software, like e.g. a word processor or a web application or any kind of on premise software. Rather a database is a collection of data ([1], [4]). Such a database consists of some kind of collection of data, which is persistent, and which is used by the application systems ([1], [4]). And persistent data on is data, which is not ephemeral like, e.g. input data, output data, control statements, work queues, software control blocks, intermediate results, but instead it is data that is transient in nature ([1], [4]). A database is being used by application systems. Such systems use layers of software including a database management system (DBMS) as well as applications, which run on top of the DBMS. This is what makes up the software, that is only one kind of component in a full database system [4].

If we assume that a database is a collection of persistent data, a text database, in turn, is a database whose primary content are texts [4]. Annotated in the context of text databases means, that the text itself contains information about the texts [4]. Hence an annotated text database management system (ATDBMSS) is a piece of software performing DBMS function on one or more annotated text databases [4]. Such a system is used by annotated text database applications in order to provide text database services to human users [4]. Sandborg-Petersen bases these notions of an ATDBMS on the definitions of Doedens (1994). Compared to Sandborg-Petersen Doedens does not distinguish between a text database management system and a text database system. Instead Sandborg-Petersen defines a text database system as the whole system with the user input, annotated text databases, hardware and software as a text database system and bases this definition on [1], [4].Based on this definition a text database management system is then part of the software encompassed by the whole text database system. Following [4]. Emdros is an annotated text database management system. Its theoretical foundations stem in a large part from Crist-Jan Doedens' PhD dissertation [2]. [4] Has more or less been following Doedens' theoretical approach and has implemented his ideas in practice while he also been repairing and clearing the theoretical work of Doedens [4]. In his PhD dissertation Doedens has defined numerous theoretical constructs related to the field of annotated text database theory [4]. Preliminary findings with regards to research question 1

In my PhD dissertation I will evaluate Emdros as annotated text database system against the following key factors for good software in software development.

1. Flexibility and Extensibility
2. Maintainability and readability
3. Performance and Efficiency
4. Scalability
5. Usability and Accessibility
6. Platform Compatibility and Manageability
7. Security
8. Functionality and Correctness

For this paper however I will specifically focus on the following two key factors 2) Maintainability and readability.

As has been pointed out in [3] one must distinguish between the persuasive macro-level of an e-learning software, which is on the one hand the infrastructural level of such a piece of software and on the other hand the governance dimension of the implementation of such an e-learning system. In this concrete case study, I focus on the infrastructural level of this software with regards to the usability of Emdros within an e-learning system.

My findings with respect to the use of Emdros with regards to the maintainability of the system and its readability have shown the following:

1. Emdros can only be installed with difficulty on various operating systems such as the Raspbian system which is used by Raspberry Pi mini-computers. In several test environments it was almost impossible to install Emdros either on a Raspberry Pi or on a local Windows Server.
2. If one follows the installation process of Emdros it is almost impossible to install the system on a computer as the description of the installation process is not understandable and the compilation of the C++ source code takes several hours without any indication of success.
3. There is no community support for Emdros now a well working and persuasively understandable documentation of the system.
4. With regards to the readability of the Emdros code the situation is likewise not good because a lack of proper documentation causes the user of the source code to read several thousand lines of code in order to be able to understand what the system does in detail.

These four findings show that the Emdros system is not maintainable nor readable in a good way for users working with it.

3   Conclusion

It will be the task for further research within my PhD project to analyze all eight key factors with regards to the persuasiveness of Emdros and other database systems with regards to their application in persuasive database-driven computer-assisted language learning. Nevertheless, a preliminary analysis has shown that Emdros as it is at the moment is not a persuasive tool to support database-driven e-learning. My suggestion when it comes to the question of what database system could be of better and more persuasive use is that MongoDB with its document based NoSQL approach could be a better choice here. If this is really the case will be shown by future research.

References

1. Date, C. J. An Introduction to Database Systems. Addison-Wesley, sixth edition, (1995).
2. Doedens, C-J. Text Databases: One Database Model and Several Retrieval Languages. Number 14 in Language and Computers. Editions Rodopi Amsterdam, Amserdam and Atlanta, GA, (1994).
3. Gottschalk J., Winther-Nielsen N. Remote but Connected: Ownership-Inspired Behavior-Driven Development and What an E-Learning Governance System for Africa Could Look like. In U. M. Azeiteiro et al. (eds.), Lifelong Learning and Education in Healthy and Sustainable Cities, World Sustainability Series, Verlag, S. 249-26. Springer, Heidelberg (2018).
4. Sandborg-Petersen, Ulrik. Annotated text databases in the context of the Kaj Munk archive: One database model, one query language, and several applications. PhD dissertation, Department of Communication and Psychology, Aalborg University, Denmark (2007).
5. Van Valin, R. D, Jr. Exploring the Syntax-Semantics Interface. Cambridge: Cambridge University Press (2005).