Decoding what the sender did not want to transmit. Information technology and historical data; or something

(Abstract of the invited talk at COMHUM 2018)

Manfred Thaller
Emeritus
University at Cologne
Germany
manfred.thaller@uni-koeln.de

In 1978 I was hired by the then Max Planck Institute for History at Göttingen, to support a number of research projects in the field of micro-history by the provision of appropriate IT technologies. The projects planned to use an approach, which was based on "extended family reconstitutions", even if that precise term was coined only a few years later. A "family reconstitution" traditionally is employed in historical demography. It starts with the marriage registers of a historical community (a village or small city) over at least two hundred years, identifies all brides and groom of the marriages in the birth and death registers, assigns all children in the birth registers to the marriages of their parents and identifies their death entries. To this network of all demographic relationships within a community an "extended family reconstitution" adds all mentions of every person in taxation registers, testaments, local court protocols – and basically every other surviving source.

It was clear from the very beginning, that such a project would take time – and it was impossible to predict at the time of data entry, what part of the source would be needed for analysis the years later. The decision was therefore, to preserve "all information" contained in the source – even if such information was vague, unclear or contradictory. A short impression of the rough solutions provided to come to grips with these properties of the data will be given.

Handling massive (for the time) data bases, quickly leads to the understanding, that while one may in the long run understand, what information is conveyed by a particular chunk of data in the source, one certainly does not immediately. This raises the question, how far the kind of information processing to be supported actually follows the classical paradigm of Shannon, where receivers are able to decode cognitively a message transmitted to them immediately. It gets worse, when one hopes to apply the usual model of information science, where a common understanding of the context is supposed to allow such a cognitive understanding.

We propose, therefore, to replace the sender-receiver metaphor in information systems dealing with historical data with an observer metaphor, where observers use observed messages to understand the context in which they have been encoded – the understanding of the observed message itself being a welcome side benefit. If one tries to implement this metaphor determinedly and without compromise, one soon discovers, that quite a few technologies of current IT systems become awkward soon – embedded markup, e.g., loses its charms, when a clear-cut separation between the (mainly) static representation of the data and the (always) dynamic interpretation of these data, a.k.a. the information assumed provisionally, is required.

While, as just mentioned, a number of technological assumptions become problematic with this new metaphor, one of the most obvious bundles of problems deals with the inherent vagueness and uncertainty of the information derived from the data.

In order of increasing complexity we will in the second half of our presentation with three example problems. For the sake of generality, we will handle these on the levels of concepts to be supported by programming languages, not on the level of application systems. While many of the approaches discussed owe much to Zadeh's concept of Fuzzy Sets, we use fuzziness in a broader sense, leaving it uncapitalized therefore.

1 Fuzzy numbers

In many historical sources – or descriptions of their assumed content – numerical data are not points in a continuum, but ranges, or sets of ranges. This is particularly obvious in the case of temporal information, where the handling of intervals has a long tradition in IT applications for historical sources, therefore, it is a more general problem however. We will briefly describe, how a datatype would look like, which can integrate the handling of such data smoothly into existing programming paradigms. We will use the examples presented earlier from the work of the late seventies and early eighties, to show how mathematical developments since then can overcome limitations of the earlier approaches and where major barriers still exist.

2 Fuzzy terms and structures

The greatest successes of computational approaches which are based on alternatives to Boolean logic are visible in the fuzzy control structures of industrial applications described as "computing with words". The classical examples in this field, as "the truth value of 'Lausanne is more or less close to Geneva' is *more or less true*", seem at first look to be extremely close to the kind of reasoning historians – or, indeed, humanists – frequently employ. We will briefly examine reasons, why that kind of approach has, nevertheless, only very rarely been applied in historical research.

We will concentrate, however, on two broader problems.

(a) As it stands, computing with words is currently almost always employed as a fuzzy pocket in an otherwise crisp information system, where the uncertainty of the decision is hidden from the main stream of the program. This would require a more general concept of a fuzzy term which could be seemlessly integrated into a program in such a way that it coexists with variables of traditional datatypes.

(b) In the semantic technologies, which are making much headway in the Humanities currently, ontologies organize terms in graphs currently. In graphs, where two nodes are either connected or unconnected by a node. Applying the logic of computing with words, we have to consider graphs, where some nodes are connected by edges which connect them with a truth value other than 'true' or 'false'.

3 Fuzzy control structures

The thorniest problem seems at first look to be the most simple. To support a logic with any kind of truth values other than 'true' or 'false' is of course no problem, as long as it is restricted to situations, where a decision about the combined truth value of a decision problem has to be made. As soon, as we intend to employ such a truth value in the parts of a programming language controlling the flow of the program, we encounter quite serious situations, where we briefly describe to what sort of larger framework a solution would require.

References

Favre-Bull, Bernard (2001). *Information und Zusammenhang. Informationsfluß in Prozessen der Wahrnehmung, des Denkens und der Kommunikation*. Wien/New York: Springer.

Liu, Sifeng and Yi Lin (2011). *Grey Systems. Theory and Practical Applications*. No. 68 in Understanding Complex Systems. Berlin/Heidelberg: Springer.

Shannon, Claude E. and Warren Weaver (1949). *The Mathematical Theory of Communication*. Champaign, IL: University of Illinois Press.

Zadeh, Lotfi A. (2005). Toward a Generalized Theory of Uncertainty (GTU) – an outline. *Information Sciences*, 172(1–2):1–40. doi:10.1016/j.ins.2005.01.017.

Zadeh, Lotfi A. and Janusz Kacprzyk, eds. (1999a). *Computing with Words in Information/Intelligent Systems 1*. No. 33 in Studies in Fuzziness and Soft Computing. Heidelberg: Physica. doi:10.1007/978-3-7908-1873-4.

Zadeh, Lotfi A. and Janusz Kacprzyk, eds. (1999b). *Computing with Words in Information/Intelligent Systems 2*. No. 34 in Studies in Fuzziness and Soft Computing. Heidelberg: Physica. doi:10.1007/978-3-7908-1872-7.