

Reasoning with positive and negative cases

Thesis subject proposed by Emmanuel Nauer and Jean Lieber (team \mathcal{K} of the Loria),
UL, CNRS, Inria, Loria, F-54000 Nancy (firstname.lastname@loria.fr)

Scientific Context. Case-based reasoning (CBR [1]) is a reasoning model relying on a case base CB where a case is a representation of a problem-solving experience, generally given by a pair $(pb, sol(pb))$ where pb is a problem in a given application domain and $sol(pb)$ is a solution of pb . For example, in the cooking domain, a problem can be $pb_{\text{pineapple}} = \langle \text{I want a recipe with pineapple.} \rangle$ and a solution $sol(pb_{\text{pineapple}})$ could be a pineapple pie recipe. A *source case* is a case $(srce, sol(srce))$ of CB.

A CBR session takes as input a new problem, the *target problem* tgt and is usually decomposed into the following steps. First, the source case $(srce, sol(srce))$ that is the closest one to tgt is selected (retrieval step). Then, $sol(srce)$ is modified into a solution $sol(tgt)$ of tgt (adaptation step). After that, the newly formed case $(tgt, sol(tgt))$ is proposed to the user (validation step) and, if validated, stored in CB (memorization step). For example, if a cooking CBR system is queried with $tgt = pb_{\text{pineapple}}$, a recipe of apple pie may be found (if there are no source case more similar to tgt than that) and then it can be adapted by substituting apples by pineapples (changing the quantities —3 apples would correspond to a portion of a pineapple— and making other adjustments).

For this purpose, a CBR system uses a knowledge base constituted classically of four *knowledge containers* [2]: CB, the domain ontology, the similarity (i.e., retrieval knowledge) and the adaptation knowledge (often represented by adaptation rules). In order to enrich this knowledge base, knowledge acquisition methods and tools (with experts and/or from data) have been studied. For example, we have studied adaptation knowledge acquisition using knowledge discovery techniques [3].

This classical schema for CBR is based on an implicit assumption: the source cases are *positive cases*, i.e., they are assumed to be satisfying (in a sense that is largely domain-dependent: in cooking, a positive case corresponds to a recipe that is appreciated by many people). Now, there exist also *negative cases*, in particular the cases $(tgt, sol(tgt))$ proposed by adaptation but rejected at the validation time. Such cases are almost never considered by CBR systems, which is a waste of potentially useful knowledge units.

Objective of the thesis. The objective of this thesis is to consider these negative cases, once labelled as such, as knowledge units for themselves for future CBR sessions. The main idea is that the case base is partitioned in $CB = CB^+ \cup CB^-$ and that a source case is used in a different way according to the fact that it is positive (member of CB^+) or negative (member of CB^-).

A new schema for CBR has to be built, but here are ideas on how to re-consider the various components of a CBR system to take into account these two types of cases:

Retrieval The idea could be to retrieve the closest positive source case $(srce^+, sol(srce^+))$ and the closest negative source case $(srce^-, sol(srce^-))$. If retrieval is implemented with the help of a similarity measure, should there be two such measures?

Adaptation The principle could be to *reuse* $(srce^+, sol(srce^+))$ and to «avoid» $(srce^-, sol(srce^-))$. This could be based on the analysis of the difference between the solutions $sol(srce^+)$ and $sol(srce^-)$. Another way to see it would be to use belief change operations: belief revision is used for some approaches to adaptation [4] and could be used for adapting $(srce^+, sol(srce^+))$, but another belief change operations, such as *contraction*, could be used for avoiding $(srce^-, sol(srce^-))$.

Validation and memorization The interactive validation process could remain the same: the newly formed case $(tgt, sol(tgt))$ is either tagged as positive or negative. Then, memorization would store $(tgt, sol(tgt))$ either in CB^+ or in CB^- , according to the result of its validation.

Knowledge acquisition methods and tools For each knowledge container, the question raised is on how to handle the positive/negative cases:

Case base Beside the negative cases learned through the (in)validation-memorization process, should there be particular effort to acquire negative cases? If so, should these cases be chosen close to positive cases?

Domain ontology The domain ontology can be seen as a set of necessary conditions for a positive case to be licit. So, the question raised is how can a negative case be used? It could be helpful to suggest a new necessary condition to be added to the domain ontology that avoids the re-occurrence of such a negative case in the future. This can be likened to the classical problem of learning with positive and negative examples in machine learning [5].

Similarity As mentioned above, retrieval of a closest positive case and retrieval of a closest negative case may be of different natures. Therefore, this difference can reflect on the acquisition of retrieval knowledge.

Adaptation knowledge A classical way of adaptation knowledge learning consists in mining the case base for differences between source cases and then in interpreting the result in terms of adaptation rules. The question here is how this principle can be modified to take into account both positive and negative cases? A first study has been carried out that gives very encouraging results [6].

Application Context. The principles proposed during this thesis must be validated. Two kinds of validation are planned: one in a toy application that enables to accurately measure the benefit of using negative cases and one in the framework of a real application developed in our team and with our partners, for the purpose of practical validation. This application can be in cooking [7], medical diagnosis [8] and/or machine translation [9].

References

- [1] C. K. Riesbeck and R. C. Schank. *Inside Case-Based Reasoning*. Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, 1989.
- [2] M. Richter and R. Weber. *Case-based reasoning: a textbook*. Springer Science & Business Media, 2013.
- [3] E. Gaillard, J. Lieber, and E. Nauer. Adaptation knowledge discovery for cooking using closed itemset extraction. In *The 8th Int. Conf. on Concept Lattices and their Applications - CLA 2011*, October 2011.
- [4] J. Cojan and J. Lieber. Applying Belief Revision to Case-Based Reasoning. In H. Prade and G. Richard, editors, *Computational Approaches to Analogical Reasoning: Current Trends*, volume 548 of *Studies in Computational Intelligence*, pages 133 – 161. Springer, 2014.
- [5] R. Michalski, J. Carbonell, and T. Mitchell. *Machine learning: An artificial intelligence approach*. Springer Science & Business Media, 2013.
- [6] T. Gillard, J. Lieber, and E. Nauer. Improving Adaptation Knowledge Discovery by Exploiting Negative Cases: First Experiment in a Boolean Setting. In *ICCBR 2018 - 26th International Conference on Case-Based Reasoning*, Stockholm, Sweden, July 2018.
- [7] A. Cordier, V. Dufour-Lussier, J. Lieber, E. Nauer, F. Badra, J. Cojan, E. Gaillard, L. Infante-Blanco, P. Molli, A. Napoli, and H. Skaf-Molli. Taaable: a Case-Based System for personalized Cooking. In S. Montani and L. C. Jain, editors, *Successful Case-based Reasoning Applications-2*, volume 494 of *Studies in Computational Intelligence*, pages 121–162. Springer, 2014.
- [8] M. B. Chawki, E. Nauer, N. Jay, and J. Lieber. Tetra: A Case-Based Decision Support System for Assisting Nuclear Physicians with Image Interpretation. In *ICCBR 2017 - Int. Conf. on Case-Based Reasoning*, volume 10339 of *Lecture Notes in Computer Science*, pages 108–122. Springer, June 2017.
- [9] Y. Lepage and J. Lieber. Case-Based Translation: First Steps from a Knowledge-Light Approach Based on Analogy to a Knowledge-Intensive One. In *ICCBR 2018 - 26th Int. Conf. on Case-Based Reasoning*, 2018.