Temporal sequence mining for improved situation understanding in robotics applications

Tove Gustavi
Swedish Defence Research Agency (FOI)

May 21, 2014

Abstract

For both humans and robots, a key to understanding the world is to understand causal relationships. This position paper describes how causal relationships, implicitly given in the form of frequently occurring action sequences, can be retrieved from data sets containing series of actions performed as part of everyday household tasks.

1 Background and motivation

Unlike humans, robots are depending on detailed instructions even when performing tasks of comparatively low complexity. The robotics community is working to improve the robots' cognitive skills and abilities to perform intelligent information processing, with the aim of creating a new generation of robots that will be able to interact with the surrounding world with a higher degree of independence.

To interact smoothly with the world, both humans and robots need a certain degree of situation understanding. Situation understanding implies, among other things, an awareness and understanding of causal relationships. Understanding of causal relations helps to predict the actions of others and to understand what is an apropriate way to act in a given situation.

Adult humans normally have a very good understanding of causal relationships, aquired through logical reasoning and long experience. For robots, on the other hand, causal relations poses a problem. It seems reasonable, however, that a robot with access to sufficient amounts of relevant background data, such as observations of people acting in situations similar to the one considered, should be able to speed up the "learning-from-experience" process. Based on training data it should be possible to draw conclusions of the type "action B follows action A with a probability of 90%".

Systematically going through all possible action combinations in order to detect strong dependencies between actions may not be feasible, due to the computational complexity. Especially if action sequences that involves more than two actions needs to be considered, one needs to turn to more sophisticated approaches.

The work presented in this paper shows how frequently occurring action sequences can be retrieved, using data mining techniques, from data sets containing series of actions performed as part of everyday household tasks. Once the frequent sequences - i.e. sequences that show up "sufficiently often" in the reference data set - have been found, the amount of data that is still interesting to examine has hopefully been significantly reduced. The frequent sequences can then be further processed and used to identify strong causal dependencies that involve series of consecutive actions.

2 Related work

In this work it is assumed that input data is on text format. Other recent work on semantically based reasoning with applications in robotics include [1], [2] and [3]. In [4], Tenorth *et al.* uses Bayesian Logic Networks to describe relations such as partial ordering between common household tasks. The approach taken in [4] is similar to the approach taken in this paper in that it does not assume that the Markov property hold in everyday tasks (see also [5]). A common way to model relations between actions is otherwise using HMM:s, which assume that the probability of an action depends only on the action that immediately preceedes it. In fact, the results presented in Section 5 clearly shows that dependencies stretches over sequences of actions. Note that the CMU test data set used in Section 5 is also one of the data sets used in [4].

3 Sequence mining

Going through a data set in order to find frequently occurring sequences is hard because of the computational complexity. Algorithms for sequence mining can in general be seen as variations of the GSP algorithm, introduced by Srikant and Agrawa in 1996 [6]. These algorithms are based on the observation that for a sequence to be frequently occurring (i.e. its frequency exceeds some use defined threshold), all its subsequences have to be frequently occurring as well. In other words, if the sequence $\{\texttt{a,b}\}$ is *n*ot frequent then the sequence $\{\texttt{a,b,c}\}$ can not be frequent either. This fact allows for efficient pruning of the set of frequences that have to be evaluated.

The GSP algorithm in its basic form assumes a sequence to be a set of items that follow directly after each other in a data set. The algorithm only considers the order of occurrence, while in many practical applications the items are associated with time stamps that need to be taken into account. There may be constraints saying, for instance, that two items must take place within a certain time frame to belong to the same sequence. In reality one may also need to allow for some "misplaced" items in between two items in a sequence. If time dependencies have to be considered, the problem complexity increases. Work that consider aspects of temporal sequence mining include [7] and [8].

4 Implementation

In the CMU data set that is used for evaluation, each observed action is represented as an item associated with the following attributes:

- Verb
- Object_1 (object associated with the verb)
- Preposition
- Object_2 (object associated with the preposition)
- Start time
- End time

In addition, a seventh attribute, *Action type*, was added in the implementation. The *Action type* decides wich of the first four attributes that need to match for two observed actions to be considered equal by the implemented algorithm. The *Verb* attribute is always considered, but the others are optional. The time related attributes are treated separately and are only used by the algorithm to check if the time constraints are satisfied.

With the chosen action representation, a sequence is here considered to be a series of observed actions $\{a_1, a_2, ..., a_K\}$ such that the start time of action a_{k+1} is smaller than the end time of action a_k plus a constant time threshold α. It is not required that a_{k+1} is observed directly following the observation of a_k. Thus, it is possible for several sequences to overlap in time.

A modified version of the GSP algorithm - adapted to take the time constraints into consideration - was used to detect the frequent sequences.

5 Experiments

As reported, the implementation was tested on the CMU MMAC Data Set [9], which contains the labeled actions of 16 people making brownies. The output from the algorithm consists of the detected frequent sequences of length k ($k = 1, 2, ...$). On the next page, part of the output for a test with *Action type* defined by the full set of descriptive attributes $\{$*Verb*, *Object_1*, *Preposition*, *Object_2*$\}$ is displayed. An action type is represented in the output as

`verb/object_1/preposition/object_2`.

As seen, all frequent sequences of length 6 are overlapping with the only frequent sequence of length 7. Frequent sequences of length <6 are not displayed here, although they could potentially have strong support. A frequent sequence of length <6, which was found in the same test, is the following 4-sequence:

```
take/oil//-twist_off/cap//-pour/oil/...
into/measuring_cup_small-twist_on/cap//.
```

6 Conclusion

Although ideally one would want to try the algorithm on a larger and more varied data set, where people were performing different tasks rather than the same task, the results obtained in this test showed that the algorithm

```
Frequent 6-sequences:
open/fridge//-take/egg//-close/fridge//-walk//to/counter-crack/egg//-crack/egg//
put/baking_pan/into/oven-walk//to/fridge-open/fridge//-close/fridge//-walk//to/counter-crack/egg//
walk//to/fridge-open/fridge//-close/fridge//-walk//to/counter-crack/egg//-crack/egg//
walk//to/fridge-open/fridge//-take/egg//-close/fridge//-crack/egg//-crack/egg//
walk//to/fridge-open/fridge//-take/egg//-close/fridge//-walk//to/counter-crack/egg//
walk//to/fridge-open/fridge//-take/egg//-walk//to/counter-crack/egg//-crack/egg//
walk//to/fridge-take/egg//-close/fridge//-walk//to/counter-crack/egg//-crack/egg//

Frequent 7-sequences:
walk//to/fridge-open/fridge//-take/egg//-close/fridge//-walk//to/counter-crack/egg//-crack/egg//

Frequent 8-sequences:
NONE
```

performed well and gave the expected results. Future work includes testing the algorithm on other data sets and exploring ways of extracting causal relationships between activities from the detected frequent sequences.

Acknowledgements

This work was sponsored by a VINNMER grant from the Swedish Governmental Agency for Innovation Systems, VINNOVA. The data used in this paper was obtained from kitchen.cs.cmu.edu and the data collection was funded in part by the National Science Foundation under Grant No. EEEC-0540865. The author also wishes to thank CAS/CVAP at KTH.

References

[1] D. Nyga and M. Beetz, *Everything Robots Always Wanted to Know about Housework (But were afraid to ask)*, in Proc. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012.

[2] D. Pangercic, B. Pitzer, M. Tenorth and M. Beetz, *Semantic Object Maps for Robotic Housework - Representation, Aquisition and Use*, in Proc. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012.

[3] S.R.K. Branavan, N. Kushman, T. Lei and R. Barzilay, *Learning High-Level Planning from Text*, in Proc. annual meeting of the Association for Computational Linguistics (ACL), 2012.

[4] M. Tenorth, F. De la Torre, and M. Beetz, *Learning Probability Distributions over Partially-Ordered Human Everyday Activities*, Proceedings of 2013 IEEE International Conference on Robotics and Automation (ICRA), 2013.

[5] Y. Shi, Y. Huang, D. Minnen, A. Bobick, I. Essa, *Propagation networks for recognition of partially ordered sequential action*, Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR, 2004.

[6] R. Srikant, and R. Agrawal, *Mining Sequential Patterns: Generalizations and Performance Improvements*, Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology, 1996.

[7] T. Guyet, and R. Quiniou, *Extracting Temporal Patterns from Interval-based Sequences*, Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence IJCAI'11, Barcelona, Spain, 2011.

[8] Y.-C. Chen, J.-C. Jiang, W.-C. Peng, and S.-Y. Lee, *An Efficient Algorithm for Mining Time Interval-based Patterns in Large Databases*, Proceedings of the 19th ACM international conference on Information and knowledge management, Toronto, Canada, 2010.

[9] F. De la Torre, J. Hodgins, J. Montano, S. Valcarcel, and J. Macey, *Guide to the Carnegie Mellon University Multimodal Activity (CMUMMAC) Database*, CMU-RI-TR-08-22, Robotics Institute, Carnegie Mellon University, Tech. Rep., 2009.