Implication of the Temporal Alignment on the Probabilistic Attribute

Andrin Betschart

University of Zurich
09-714-882

I. Introduction

In this work we are dealing with Temporal Probabilistic Databases. Relations in those databases consist of a temporal attribute, which defines in what interval the tuples are valid, a probabilistic attribute, which is a non-temporal attribute that has the ability to define how certainly the tuples are valid, and other non-temporal attributes, which define what the tuples are about. We perceive time according to Sequenced Semantics and we choose to do Temporal Alignment, a state of the art technique introduced by Dignös et al. [1], to query data in this Temporal Probabilistic Databases. We adjust the temporal attribute as suggested and adapt this procedure to deal with the probabilistic attributes as well. To be precise we investigate in this report in detail how good it is when the probabilistic attribute is treated as a common non-temporal attribute, whose semantics make no difference during Temporal Alignment.

The rest of the report is organized as follows. In Section 2 we discuss related work that deals with Temporal Alignment. Section 3 describes in detail how we deal with the probabilistic attribute and our approach is illustrated with an example. Finally we conclude with an evaluation in Section 4.

II. Related Work

In Temporal Probabilistic Databases every tuple contains a temporal attribute, which defines the starting and ending points of the time interval in which the tuple is valid. By choosing to interprete time using Sequenced Semantics we look at the database as a sequence of snapshots of all points in time. A snapshot can be seen as the data that are valid at a specified point in time. In order to support Sequenced Semantics the three properties snapshot reducibility, extended snapshot reducibility and change preservation must be satisfied.

Two primitives are proposed by Dignös et al. [1] to transform each tuple into a set of new ones with adjusted time intervals that allow to further query the data with standard non-temporal operators. The goal is to align the time intervals of the tuples in such a way that all intervals that have to be compared are either identical or disjoint. Using this primitives they introduced Reduction Rules that allow us to reduce the temporal operators into non-temporal ones while satisfying all three properties of the Sequenced Semantics.

Depending on how the operation that a query consists of produces its result tuples, either the Temporal Splitter or the Temporal Aligner must be applied. For group based operators $\{\pi, \vartheta, \cup, -, \cap\}$ multiple tuples of the argument relations contribute to a single result tuple. In this case the Temporal Splitter is used to adjust the time intervals. For tuple based operators $\{\sigma, \times, \bowtie, ⟗, ⟕, ⟖, \triangleright\}$ only one tuple of each argument relation contributes to a single result tuple. In this case the Temporal Aligner is used to adjust the time intervals.

Temporal Splitter

For group based operators Dignös et al. [1] propose a Temporal Splitter to adjust the time intervals. It is necessary that the time intervals of the tuples of the argument relations are split at every start and end point of all other tuples that satisfy the same conditions. This means that a tuple is split into tuples with identical non-temporal attributes but with disjoint adjusted time intervals, where the union of all adjusted time intervals equals the initial one. According to Dignös et al. [1], the definition of a Temporal Splitter requires two conditions to hold. Let r be the tuple that we are splitting and g the set of tuples according to which r will be split. The first condition requires that each adjusted time interval is a subinterval of the initial time interval of r and that it is either contained in or disjoint from all time intervals of g. Secondly, each new interval must be maximal, meaning that it cannot be enlarged without violating the first condition. The temporal normalization function is a function that satisfies the properties of a Temporal Splitter.

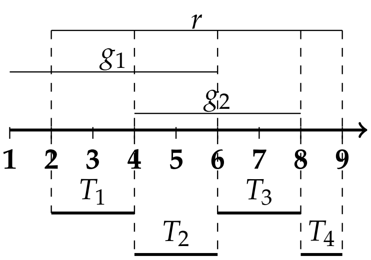

Figure 1: *Temporal Splitter [1].*

In Figure 1 the tuple r is split into four by time disjoint tuples T_1 to T_4. The time intervals of the tuples T_1 and T_3 are both completely contained in one tuple of g and completely disjoint to the other tuple of g. T_2 is completely contained in both tuples of g, whereas T_4 is completely disjoint to all tuples of g. In addition, the time intervals of all result tuples are a subinterval of the time interval of tuple r and cannot be enlarged without violation the first condition.

Temporal Aligner

For tuple based operators Dignös et al. [1] propose a Temporal Aligner to adjust the time intervals. It is necessary that the time intervals of the tuples of the argument relations are aligned to all other tuples that satisfy the same conditions. This means that a tuple r is split into tuples with identical non-temporal attributes but with adjusted, not necessarily disjoint time intervals, where the union of all new time intervals equals the initial one. Let r be the tuple whose time interval we are adjusting and g the set of tuples we are adjusting on. The definition of a Temporal Aligner by Dignös et al. [1] states that the adjusted time interval is either the intersection of the time interval of the tuple r with the time interval of a tuple of the set g or it is a subinterval of the time interval of r that is maximal and disjoint to all time intervals of tuples in g. The temporal alignment function satisfies the properties of a Temporal Aligner.

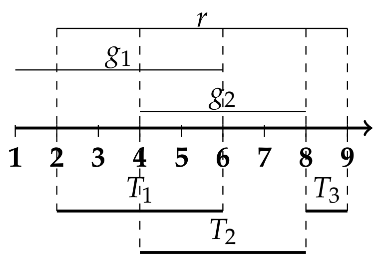

Figure 2: *Temporal Aligner [1].*

In Figure 2 the tuple r is split into three tuples T_1, T_2 and T_3. The time intervals of the tuples T_1 and T_2 are the intersections of the time interval of r and g_1 respectively g_2. And T_3 is a subinterval of the time interval of tuple r, which is maximal and disjoint to all time intervals of tuples from g.

III. Probabilistic Attribute

In non-probabilistic databases, also called deterministic databases, we consider every tuple in the database as an event for the validity of which we are sure. But since we are dealing with Temporal Probabilistic Databases we have to adjust our assumption. In this case every tuple has an probabilistic attribute p. Suciu et al. [3] introduce different approaches about how to perceive this probabilistic attribute. One way to look at it is, that p is the chances of its corresponding tuple to belong to the database. For example if the tuples in a database are parsed from the internet, one can not be 100% sure if the facts are really true. Another way to see probabilistic attribute is, that it is the probability with which the event described by the tuple is going to happen. This is the case when the events lay in the future and are not totally predictable.

For this work we assume the second interpretation and we evaluate an approach that takes the probabilistic attribute as a normal attribute during temporal adjustments. This means we simply copy the probability attribute from the argument relations to the corresponding result tuples, without modifying anything.

Example

Let us illustrate our point with an example. We consider two relations P and W. Both relations have a temporal attribute T, which defines during what time interval the described tuple is true, and a probabilistic attribute p.

P **(People)**

	Name	Dest	T	p
p_1	Ann	Zurich	[3,14)	0.8
p_2	Joe	Zurich	[4,11)	0.5
p_3	Mark	Bozen	[6,12)	0.7
p_4	Jim	Zurich	[5,10)	0.2
p_5	Tina	Bozen	[10,13)	1.0

W **(Weather)**

	Loc	Weather	T	p
w_1	Zurich	Sun	[1,8)	0.1
w_2	Zurich	Rain	[11,17)	0.6
w_3	Bozen	Snow	[5,10)	0.9
w_4	Zurich	Fog	[8,15)	0.3
w_5	Bozen	Sun	[6,9)	0.4

Figure 3: *People and Weather relations.*

The relation P contains information about people, who are visiting certain cities for a specific time. Consider tuple p_1 in relation P. This tuple tells us that Ann is in Zurich from day 3 just until before day 14. The probability with which this event is expected to happen is 80%. In addition, each tuple w_i in the relation W describes a weather forecast, where *Loc* is the Location, where the weather takes place, *Weather* is the kind of weather that is expected, T is the time interval for which the weather is expected and p is the probability with which the described weather occurs. For example tuple w_1 states that it will be sunny in Zurich from day 1 until before day 8 with a probability of 10%.

Figures 4 and 5 graphically illustrate the temporal adjustments of p_1 using W. As we use the condition $P.Dest = W.Loc$ for both, the temporal normalization and the temporal alignment, only the tuples w_1, w_2 and w_4 have en effect on the resulting time intervals, since the other tuples are valid for other locations. As we treat the probabilistic attribute as a normal attribute it remains unmodified during the temporal adjustments.

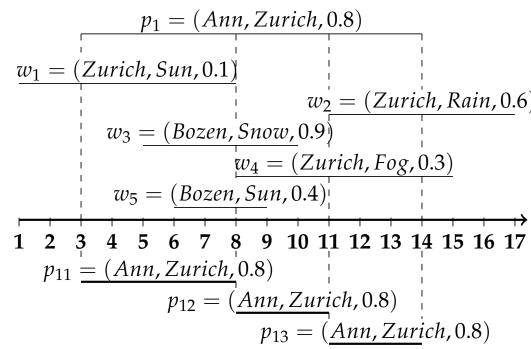

Figure 4: *Temporal normalization of p_1 using W.*

By applying the temporal normalization the time interval of p_1 is split at all points where one of the tuples w_1, w_2 or w_4 starts respectively ends.

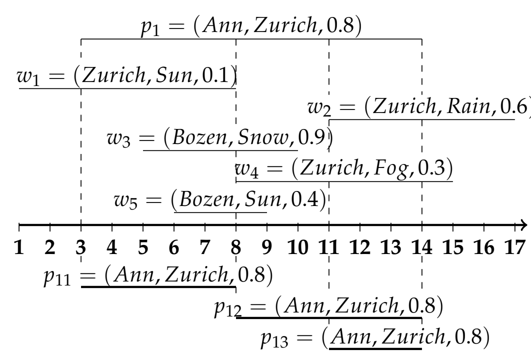

Figure 5: *Temporal alignment of p_1 using W.*

By applying the temporal alignment the tuples p_{11}, p_{12} and p_{13} are derived from the intersection of p_1 with w_1, w_2 or w_4 respectively. Since the whole time interval of p_1 is covered by w_1, w_2 and w_4 no additional result tuple is required.

The complete result of the temporal normalization respectively alignment using the condition $P.Dest = W.Loc$ with the probabilistic attribute treated as a normal attribute can be seen in the tables of Figure 6.

Temporal normalization of P using W.

	Name	*Dest*	*T*	*p*
p_{11}	Ann	Zurich	[3, 8)	0.8
p_{12}	Ann	Zurich	[8, 11)	0.8
p_{13}	Ann	Zurich	[11, 14)	0.8
p_{21}	Joe	Zurich	[4, 8)	0.5
p_{22}	Joe	Zurich	[8, 11)	0.5
p_{31}	Mark	Bozen	[6, 9)	0.7
p_{32}	Mark	Bozen	[9, 10)	0.7
p_{33}	Mark	Bozen	[10, 12)	0.7
p_{41}	Jim	Zurich	[5, 8)	0.2
p_{42}	Jim	Zurich	[8, 10)	0.2
p_{51}	Tina	Bozen	[10, 13)	1.0

Temporal alignment of P using W.

	Name	*Dest*	*T*	*p*
p_{11}	Ann	Zurich	[3, 8)	0.8
p_{12}	Ann	Zurich	[8, 14)	0.8
p_{13}	Ann	Zurich	[11, 14)	0.8
p_{21}	Joe	Zurich	[4, 8)	0.5
p_{22}	Joe	Zurich	[8, 11)	0.5
p_{31}	Mark	Bozen	[6, 10)	0.7
p_{32}	Mark	Bozen	[6, 9)	0.7
p_{33}	Mark	Bozen	[10, 12)	0.7
p_{41}	Jim	Zurich	[5, 8)	0.2
p_{42}	Jim	Zurich	[8, 10)	0.2
p_{51}	Tina	Bozen	[10, 13)	1.0

Figure 6: *Result relations.*

IV. Evaluation

We think of two major interpretations of the probabilistic attribute for which the approach of not modifying the probabilistic attribute during Temporal Alignment is suitable.

One way to interpret the probabilistic attribute is by assuming that it defines the chances of the described event happening. Further we consider that the whole event, this means from the start until the end of the time interval, is either happening or not. This means that the tuple $(''Ann'', ''Zurich'', [3, 14), 0.8)$ describes that Ann will be in Zurich from day 3 to 14 with a probability of 80%. In this case keeping the probabilistic attribute unmodified during normalization and alignment seems to be the only correct choice. No matter how small the subintervals we consider, the chances of the event taking place during this interval are still the same as in the initial interval.

A second way is to interpret the probabilistic attribute as if it describes the proportion of time in the given time interval in which the described event will happen. This means that the tuple $(''Zurich'', ''Rain'', [3, 7), 0.6)$ describes that from day 3 to 7 it will be raining in Zurich during 60% of the time. When we now assume that the realisation of the event is uniformly distributed over the whole interval, we still would not modify the probabilistic attribute when considering subintervals. Since the selected time proportion is independent of the interval length, it would be correct to say that from day 3 to 5 it will be raining in Zurich during 60% of the time. This is because we expect the event to happen in exactly 60% of the time no matter how small the time interval that we are considering. If we were dealing with a different distribution, for example if we knew that it is more likely that a larger proportion of the event happens at the beginning of the interval, we would have to adjust the probabilistic attribute when considering subintervals. In this case a scaling of the probabilistic attribute during Temporal Alignment might be appropriate.

In contrast, there also exists an interpretation of the probabilistic attribute according to which it makes no sense to maintain the probabilistic attribute unmodified during Temporal Alignment. Consider that the event, described by a tuble, happens to the given probability at only some undefined subinterval of the given time interval. In this case scaling of the probabilistic attribute might be appropriate when considering subintervals. This means that the tuple $(''Zurich'', ''Rain'', [3, 7), 0.6)$ describes that with 60% probability it will be raining in Zurich at some time from day 3 to 7. In this case one could see the chances that it will rain in Zurich at some time from day 3 to 5 as 30% and therefore a linear scaling could be appropriate.

References

[1] Anton Dignös, Michael H. Böhlen, and Johann Gamper. Temporal Alignment. In *SIGMOID Conference*, pages 433-444, 2012.

[2] Anton Dignös, Michael H. Böhlen, and Johann Gamper. Query time scaling of attribute values in interval timestamped databases. In *ICDE*, 2013.

[3] Dan Suciu, Dan Olteanu, Christopher Ré, and Christoph Koch. *Probabilistic Databases*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2011.