Implication of the Temporal Alignment on the Probabilistic Attribute

Martin Leimer

University of Zurich

09-728-569

I. Introduction

Temporal Probabilistic Databases consist of temporal attributes, e.g. timestamps or time intervals alongside with non-temporal ones. One of the non-temporal attributes includes a probability value corresponding to the tuple. We will perceive time using Sequenced Semantics, which allows us to apply state of the art methods like Temporal Alignment and Temporal Normalization. These will allow us to apply temporal operations by splitting up the time intervals. As time intervals get adjusted, the probabilistic attribute gets invalid. Therefore, we propose scaling of the confidence of each tuple to resolve it. We will start with a short reflection on Temporal Alignment in Section 2, before we go into the details of scaling using examples in Section 3. Finally we conclude this report with an evaluation in Section 4.

II. Related Work

Dignös et al. [1] described that time is represented as an interval consisting of a starting and ending timestamp in Temporal Databases, as shown with the tuples r, g_i and T_i in Figures **??** and **??**.

In the following, we will perceive time using Sequenced Semantics. Therefore, to be able to use temporal operators, they propose a Temporal Splitter for group based operators, $\{\pi, \vartheta, \cap, \cup, -\}$ and a Temporal Aligner for tuple based operators, $\{\sigma, \times, \bowtie, ⟕, ⟖, ⟗, \rhd\}$. The appliance of the Temporal Splitter and Temporal Aligner, also called temporal normalization and temporal alignment respectively, splits up each tuple into a set of tuples, each having equal non-temporal attributes, but adjusted time intervals. This process will allow us to find for each temporal operator, its non-temporal counterpart, as shown in the reduction rules in Figure **??**.

Operator			Reduction
Selection	$\sigma^T_\theta(\mathbf{r})$	=	$\sigma_\theta(\mathbf{r})$
Projection	$\pi^T_{\mathbf{B}}(\mathbf{r})$	=	$\pi_{\mathbf{B},T}(\mathcal{N}_{\mathbf{B}}(\mathbf{r};\mathbf{r}))$
Aggregation	${}_{\mathbf{B}}\vartheta^T_F(\mathbf{r})$	=	${}_{\mathbf{B},T}\vartheta_F(\mathcal{N}_{\mathbf{B}}(\mathbf{r};\mathbf{r}))$
Difference	$\mathbf{r} -^T \mathbf{s}$	=	$\mathcal{N}_{\mathbf{A}}(\mathbf{r};\mathbf{s}) - \mathcal{N}_{\mathbf{A}}(\mathbf{s};\mathbf{r})$
Union	$\mathbf{r} \cup^T \mathbf{s}$	=	$\mathcal{N}_{\mathbf{A}}(\mathbf{r};\mathbf{s}) \cup \mathcal{N}_{\mathbf{A}}(\mathbf{s};\mathbf{r})$
Intersection	$\mathbf{r} \cap^T \mathbf{s}$	=	$\mathcal{N}_{\mathbf{A}}(\mathbf{r};\mathbf{s}) \cap \mathcal{N}_{\mathbf{A}}(\mathbf{s};\mathbf{r})$
Cart. Prod.	$\mathbf{r} \times^T \mathbf{s}$	=	$\alpha((\mathbf{r}\Phi_{true}\mathbf{s}) \bowtie_{\mathbf{r}.T=\mathbf{s}.T} (\mathbf{s}\Phi_{true}\mathbf{r}))$
Inner Join	$\mathbf{r} \bowtie^T_\theta \mathbf{s}$	=	$\alpha((\mathbf{r}\Phi_\theta\mathbf{s}) \bowtie_{\theta\wedge\mathbf{r}.T=\mathbf{s}.T} (\mathbf{s}\Phi_\theta\mathbf{r}))$
Left O. Join	$\mathbf{r} ⟕^T_\theta \mathbf{s}$	=	$\alpha((\mathbf{r}\Phi_\theta\mathbf{s}) ⟕_{\theta\wedge\mathbf{r}.T=\mathbf{s}.T} (\mathbf{s}\Phi_\theta\mathbf{r}))$
Right O. Join	$\mathbf{r} ⟖^T_\theta \mathbf{s}$	=	$\alpha((\mathbf{r}\Phi_\theta\mathbf{s}) ⟖_{\theta\wedge\mathbf{r}.T=\mathbf{s}.T} (\mathbf{r}\Phi_\theta\mathbf{r}))$
Full O. Join	$\mathbf{r} ⟗^T_\theta \mathbf{s}$	=	$\alpha((\mathbf{r}\Phi_\theta\mathbf{s}) ⟗_{\theta\wedge\mathbf{r}.T=\mathbf{s}.T} (\mathbf{s}\Phi_\theta\mathbf{r}))$
Anti Join	$\mathbf{r} \rhd^T_\theta \mathbf{s}$	=	$(\mathbf{r}\Phi_\theta\mathbf{s}) \rhd_{\theta\wedge\mathbf{r}.T=\mathbf{s}.T} (\mathbf{s}\Phi_\theta\mathbf{r})$

Figure 1: *Reduction Rules [1]*

Group based operators

Whenever the query consists of temporal group based operators, $\{\pi, \vartheta, \cap, \cup, -\}$, a Temporal Splitter is proposed to adjust the time intervals. This means that a tuple r is split into a set of tuples with identical non-temporal attributes, but disjoint adjusted time intervals, where the union of all adjusted time intervals equals the initial one. According to Dignös et al. [1], the first condition is that adjusted time intervals are either contained in or disjoint from all tuples of g. Moreover, the time intervals shall be maximal, meaning that they cannot be enlarged without violating the first condition.

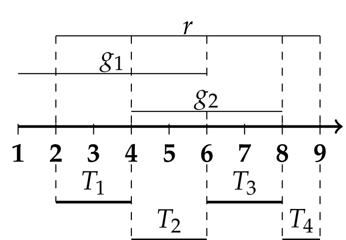

Figure 2: *Temporal Splitter [1]*

In Figure **??**, the tuple r is split into four tuples, where the tuple time intervals T_1, T_2 and T_3 are maximal and disjoint but contained in at least one tuple g_i. Tuple T_4 is not a subinterval of any g_i, but it is a subinterval of r, therefore T_4 is maximal and disjoint from all tuples of g, which is no violation of the first or second condition.

Tuple based operators

Whenever the query consists of tuple based operators, $\{\sigma, \times, \bowtie, ⟖, ⟕, ⟗, \triangleright\}$, they propose a Temporal Aligner to adjust the time intervals. This means that a tuple r is split into a set of tuples with identical non-temporal attributes. The adjusted time intervals are either the intersection of r with a time interval of some tuple g_i, or it is a subinterval of r which is maximal but disjoint with all tuples of g.

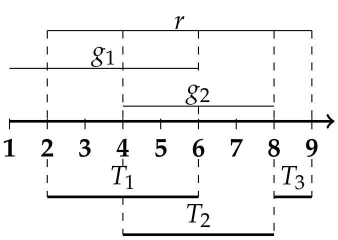

Figure 3: *Temporal Aligner [1]*

In Figure **??**, the tuple r is split into three tuples, where the tuple time intervals of T_1 and T_2 are the intersection between r and g_1, g_2 respectively. T_3 is a subinterval of r which is maximal but disjoint with all tuples of g.

III. Probabilistic attribute

The probabilistic attribute is tuple related and belongs to the non-temporal ones. Tuple relation means, that its probability value holds for the tuple as a whole, over all attributes, including its time interval. Therefore, any adjustment of the time interval, must result in a correction of the probability value. As temporal normalization and alignment enforce adjustments of the time intervals, we introduce scaling to adjust the probabilistic attribute as well.

Basically there are three options on how scaling can be done:

- *Constant*: No scaling at all
- *Linear*: The probabilistic attribute is scaled linearly to the ratio between the adjusted and initial time interval: $p_{adjusted} = \frac{Te_{adjusted} - Ts_{adjusted}}{Te_{initial} - Ts_{initial}} * p_{initial}$
- *Function*: The probabilistic attributed is scaled according to some function $f(t_{adjusted}, t_{initial})$

The scaling method is chosen according to the attribute or tuple the probabilistic attribute belongs to. However, even with a given a relation, it might be sometimes difficult to determine the appropriate scaling method. Consider the tuple p_1 in relation $P(People)$ where Ann is travelling to Zurich with a probability of 80% from or between day 3 to/and 14. It is indeterminable whether Ann will travel once for some unspecific subinterval, e.g. 1 day to Zurich or whether she will be there for the whole period with a probability of 80%. In the first case, linear scaling might be the appropriate solution, whereas we should not scale if we perceive the data as mentioned in the second case. We therefore need some additional information such that we can apply the correct scaling method.

In the following, we assume that we have to adjust the probabilistic attribute linearly.

Consider two temporal relations $P(People)$ and $W(Weather)$ with a probabilistic attribute p as seen in Figure **??**. We will now do a temporal normalization and alignment of the relation $P(People)$ using $W(Weather)$ on $P.Dest = W.Loc$ with linear scaling.

P (People)

	Name	*Dest*	*T*	*p*
p_1	Ann	Zurich	[3, 14)	0.80
p_2	Joe	Zurich	[4, 11)	0.50
p_3	Mark	Bozen	[6, 12)	0.70
p_4	Jim	Zurich	[5, 10)	0.20
p_5	Tina	Bozen	[10, 13)	1.00

W (Weather)

	Loc	Weather	T	P
w_1	Zurich	Sun	[1, 8)	0.80
w_2	Zurich	Rain	[11, 17)	0.50
w_3	Bozen	Snow	[5, 10)	0.70
w_4	Zurich	Fog	[8, 15)	0.20
w_5	Bozen	Sun	[6, 9)	1.00

Figure 4: *Temporal Probabilistic Databases*

Figures **??** and **??** show an extract of the graphical temporal adjustment of P using W together with linear scaling of the probabilistic attribute p. As we do a temporal adjustment on $P.Dest = W.Loc$, temporal normalization/alignment for tuple p_1 is only affected by the tuples w_1, w_2 and w_4, as the others hold for different locations.

Temporal Normalization

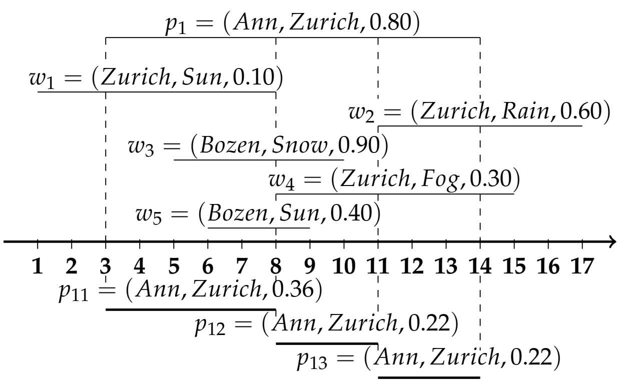

Figure 5: *Temporal Normalization of p_1 using W combined with linear scaling*

For temporal normalization, the tuple p_{11} derives from the intersection of p_1 with w_1, as there is no other tuple w_i which intersects the time interval of w_1 for the same location. p_{12} and p_{13} derive from a temporal normalization on p_1 using w_2 and w_4. The complete result is shown in Figure **??**.

	Name	*Dest*	*T*	*p*
p_{11}	Ann	Zurich	[3, 8)	0.36
p_{12}	Ann	Zurich	[8, 11)	0.22
p_{13}	Ann	Zurich	[11, 14)	0.22
p_{21}	Joe	Zurich	[4, 8)	0.29
p_{22}	Joe	Zurich	[8, 11)	0.21
p_{31}	Mark	Bozen	[6, 9)	0.35
p_{32}	Mark	Bozen	[9, 10)	0.12
p_{33}	Mark	Bozen	[10, 12)	0.23
p_{41}	Jim	Zurich	[5, 8)	0.12
p_{42}	Jim	Zurich	[8, 10)	0.08
p_{51}	Tina	Bozen	[10, 13)	1.00

Figure 6: *Temporal Normalization of P using W*

Temporal Alignment

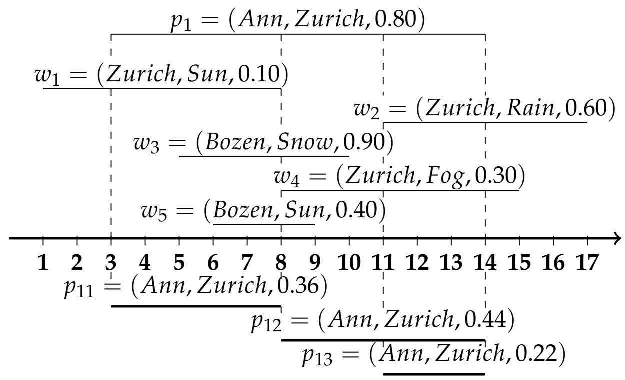

Figure 7: *Temporal Alignment of p_1 using W combined with linear scaling*

For temporal alignment, the tuple p_{11} derives from the intersection of p_1 with w_1, as in the temporal normalization above. However, p_{12} and p_{13} are now derived from the intersection of p_1 with w_4 and w_2 respectively. There is no additional result tuple, as w_1, w_2 and w_4 cover the whole time interval of p_1. The result is shown in Figure **??**.

	Name	*Dest*	*T*	*p*
p_{11}	Ann	Zurich	[3, 8)	0.36
p_{12}	Ann	Zurich	[8, 14)	0.44
p_{13}	Ann	Zurich	[11, 14)	0.22
p_{21}	Joe	Zurich	[4, 8)	0.29
p_{22}	Joe	Zurich	[8, 11)	0.21
p_{31}	Mark	Bozen	[6, 10)	0.47
p_{32}	Mark	Bozen	[6, 9)	0.35
p_{33}	Mark	Bozen	[10, 12)	0.23
p_{41}	Jim	Zurich	[5, 8)	0.12
p_{42}	Jim	Zurich	[8, 10)	0.08
p_{51}	Tina	Bozen	[10, 13)	1.00

Figure 8: *Temporal Alignment of P using W*

IV. Evaluation

The possibility to calculate with temporal attributes using normalization and alignment is a big and useful instrument. However, with the extension of probabilistic attributes, we have to scale the values whenever the time interval changes. As this might be easy in a first instance, we have seen that it is actually dependent on and how we perceive information in the database whether and how we have to scale values.

As linear scaling might seem adequate for most cases, it also has a significant drawback. Assuming that each event will take at least some time amount x, whereas the initial time interval T only specifies in which time range this event might occur, linear scaling is an appropriate heuristic as long as $x \geq duration(T_{adjusted})$. However, if this condition is not satisfied, the adjusted probability is invalid, especially when the adjusted time interval $T_{adjusted}$ goes to zero, as the probability p will go to zero as well.

In order to overcome the above mentioned obstacles, we propose the use of the following function:

$$p_{adjusted} = \begin{cases} \frac{duration(T_{adjusted})}{duration(T_{initial})} * p_{initial} & if x \geq duration(T_{adjusted}) \\ \frac{x}{duration(T_{initial})} * p_{initial}, & else \end{cases}$$

With this function, we do linear scaling on the probabilistic attribute as long as the adjusted time interval $T_{adjusted}$ is greater or equal to the actual duration of the event x . If the adjusted time interval $T_{adjusted}$ is shorter than the actual duration of the event x, the probability p is adjusted to the duration of the event x, instead of the adjusted time interval $T_{adjusted}$.

The advantage of this proposal is that it works even if the event takes place during the whole range described by the timestamps, e.g. Ann travels (for 11 days) to Zurich from day 3 to 14 with a probability of 80%. However, its drawback is that we need somehow to know the actual duration of the event x, which is actually nowhere described in the given relations.

Therefore, we need further descriptions of the probabilistic attributes of the relations, such that we can apply an appropriate scaling method in every case.

Acknowledgements

Many thanks to Aikaterini Papaioannou, who has supported me while learning the concepts and writing the report. Moreover, I also like to appreciate Prof. Dr. Michael Böhlen, who has given me the opportunity to do the master-base module in this interesting topic.

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 Dataabses*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2011.