A Appendix

A.1 Dataset Characteristics

The BPIC_15 comprises of a set of 5 logs pertaining to building permit process from 5 different Dutch municipalities while the BPIC_20 event logs pertain to the travel expense claims. There are five variants of the travel expenses, viz., domestic (BPIC_20_1) and international declarations (BPIC_20_2), travel permits (BPIC_20_3), requests for payment (BPIC_20_4) and pre-paid travel costs (BPIC_20_5).The BPIC_18 dataset contains event logs from a loan application process in a Dutch financial institute, detailing various stages of application handling, including approvals, rejections, and intermediate steps. The characteristics of each of these logs are presented in Table 1. Contexts of different lengths were generated using an overlapping sliding window. For each event, we considered the `concept:name` attribute and `org:resource` attribute, i.e., the activity name and the resource as input characteristics for modeling the DBNs to predict the next `concept:name`. The training is done using a 80:20 ratio split for train/test data on BPIC_15 and BPIC_20 logs and the same data split has been used to train and evaluate all models. For BPIC_18, the training is done using a 90:10 ratio split for train/test data on all models. To report the results for context lengths 5 and 10 for [4], [3], [5], we have used the code provided by Pauwels et al. [4]. For [2] and [1], we have used the code provided by the respective authors.

Table 1. Characteristics of event logs used for our study.

Characteristic	BPIC 15					BPIC 20					BPIC 18
	15_1	15_2	15_3	15_4	15_5	20_1	20_2	20_3	20_4	20_5	
events	52217	44354	59681	47293	59083	56437	72151	86581	36796	18246	2514266
cases	1199	832	1409	1053	1156	10500	6449	7065	6886	2099	43809
distinct activities	398	410	383	356	389	17	34	51	19	29	41
contexts ($k = 5$)	46236	40202	52648	42041	53303	6384	39926	51285	5799	8264	2295221
contexts ($k = 10$)	40368	36084	45926	38829	47545	267	10270	21151	1880	1285	2076176
contexts ($k = 20$)	-	-	-	-	-	-	-	-	-	-	1638086
unseen contexts (%) ($k = 5$)	36.05	42.15	30.83	32.60	35.35	00.47	00.95	05.23	00.69	04.11	17482
unseen contexts (%) ($k = 10$)	39.33	49.95	40.71	35.24	41.52	01.86	01.07	16.23	00.53	12.84	23908
unseen contexts (%) ($k = 20$)	-	-	-	-	-	-	-	-	-	-	21364

A.2 Training Algorithm for DBN

The psuedocode for the DBN graph building algorithm is shown below:

Algorithm 1 DBN Graph Building

```
Input: all-edges
Output: graph
improvement ← true
graph ← ()
while improvement do
    add_best ← ()
    best_add_delta ← 0
    for edge in all-edges do
        if edge not in graph then
            if score(graph + edge) - score(graph) > best_add_delta then
                best_add_delta ← score(graph + edge) - score(graph)
                add_best ← edge
            end if
        end if
    end for
    del_best ← ()
    best_del_delta ← 0
    for edge in graph do
        if score(graph - edge) - score(graph) > best_del_delta then
            best_del_delta ← score(graph - edge) - score(graph)
            del_best ← edge
        end if
    end for
    if best_add_delta > best_del_delta then
        graph ← graph + add_best
    else if best_del_delta > 0 then
        graph ← graph - del_best
    else
        improvement ← false
    end if
end while
return graph
```

A.3 DBN inferencing with SM algorithm

The psuedocode for the inferencing from DBN using the novel proposed SM algorithm is shown below:

Algorithm 2 DBN inferencing with SM

```
Input: test_context, prediction_fields, cpts, parents
Output: next_event
for field in prediction_fields do
    cpt ← cpts[field]
    parent_combination ← test_context(parents(field))
    if parent_combination in cpt then
        field_val ← max_w(cpt(parent_combination))
    else if at_least_one_seen(parent_combination) then
        probs ← SM(parent_combination, field)
        field_val ← max_w(probs)
    else
        field_val ← UNKNOWN
    end if
    next_event(field) ← field_val
end for
return next_event
```

Some points of note in the testing algorithm are:

- SM refers to the novel score based marginalization, detailed in Section **??**.
- The 'UNKNOWN' prediction for any feature is the value 0 in all experiments for this paper. Thus all string encodings start from 1 for each feature.

References

1. Bukhsh, Z.A., Saeed, A., Dijkman, R.M.: Processtransformer: Predictive business process monitoring with transformer network. arXiv preprint arXiv:2104.00721 (2021)
2. Di Mauro, N., Appice, A., Basile, T.M.: Activity prediction of business process instances with inception cnn models. In: AI* IA 2019–Advances in Artificial Intelligence: XVIIIth International Conference of the Italian Association for Artificial Intelligence, Rende, Italy, November 19–22, 2019, Proceedings 18. pp. 348–361. Springer (2019)
3. Lin, L., Wen, L., Wang, J.: Mm-pred: A deep predictive model for multi-attribute event sequence. In: Proceedings of the 2019 SIAM international conference on data mining. pp. 118–126. SIAM (2019)
4. Pauwels, S., Calders, T.: Bayesian network based predictions of business processes. In: Business Process Management Forum: BPM Forum 2020, Seville, Spain, September 13–18, 2020, Proceedings 18. pp. 159–175. Springer (2020)
5. Tax, N., Verenich, I., La Rosa, M., Dumas, M.: Predictive business process monitoring with lstm neural networks. In: Advanced Information Systems Engineering: 29th International Conference, CAiSE 2017, Essen, Germany, June 12-16, 2017, Proceedings 29. pp. 477–492. Springer (2017)