Capturing Dialogue State Variable Dependencies with an Energy-based Neural Dialogue State Tracker

Anh Duong Trinh †, **Robert J. Ross** †, **John D. Kelleher** ‡
† School of Computer Science
‡ Information, Communications & Entertainment Institute
Technological University Dublin, Ireland
ADAPT Centre, Science Foundation Ireland
anhduong.trinh@mydit.ie, {robert.ross, john.d.kelleher}@dit.ie

Abstract

Dialogue state tracking requires the population and maintenance of a multi-slot frame representation of the dialogue state. Frequently, dialogue state tracking systems assume independence between slot values within a frame. In this paper we argue that treating the prediction of each slot value as an independent prediction task may ignore important associations between the slot values, and, consequently, we argue that treating dialogue state tracking as a structured prediction problem can help to improve dialogue state tracking performance. To support this argument, the research presented in this paper is structured into three stages: (i) analyzing variable dependencies in dialogue data; (ii) applying an energy-based methodology to model dialogue state tracking as a structured prediction task; and (iii) evaluating the impact of inter-slot relationships on model performance. Overall, we demonstrate that modelling the associations between target slots with an energy-based formalism improves dialogue state tracking performance in a number of ways.

1 Introduction

Dialogue management for spoken dialogue systems is a challenging research domain due in part to difficulties arising from limited resources, the imperfection of technologies on which dialogue management is dependent, and of course the complexities of natural human conversation (Glass, 1999; Ward and DeVault, 2015). Within a conventional dialogue manager, an explicit dialogue state tracker is a key component that attempts to track both interlocutors' contributions to the exchange. The dialogue state tracker in particular suffers due to errors introduced by other components such as an automatic speech recognizer, and, where used, natural language understanding components (Ross and Bateman, 2009). The difficulties also lie within the uncertainties of spoken interactions, and the complexity of conversation context (Paek and Horvitz, 2000; DeVault, 2008).

To reduce the complexity of designing and parameterising a dialogue state tracker, it is typically necessary to limit application of a dialogue state tracker to a specific domain, and to cast the dialogue state as sets of slot-value pairs that are arranged into frames. This structure in its base case is best exemplified by the well-known dialogue state tracking datasets such as Let's Go (Raux et al., 2005), though the structure can also be made more complex as is the case in the tracking of multiple frames of dialogue states throughout the conversation history (El Asri et al., 2017). By casting the dialogue representation as a set of slots to be tracked, the dialogue state tracking process itself is most frequently tackled as a multi-task classification problem.

In recent years, various deep learning approaches that track dialogue states as a combination of individual classification tasks have been proposed (Ren et al., 2018; Perez and Liu, 2017; Vodolan et al., 2017; Mrksic et al., 2017; Rastogi et al., 2017). However, while these systems achieve state-of-the-art results, there remains notable room for improvement (Liu et al., 2018). Our work begins with the hypothesis that by treating dialogue state tracking as a simple multi-label classification task, we are not taking into account the relationships between dialogue state slot variables. This hypothesis is based in part on experience from other applications of machine learning that have demonstrated that taking target variable dependencies into account is useful, but is also based on the intuition that a human interlocutor would of course take multiple target variables into account while interpreting language (Landragin, 2013).

Given the above argument, in this paper we

Proceedings of the SIGDial 2019 Conference, pages 75–84
Stockholm, Sweden, 11-13 September 2019. ©2019 Association for Computational Linguistics

present an end-to-end investigation into the impact of domain variable dependencies on the dialogue state tracking process. For practical purposes, we focus our work on the Dialogue State Tracking Challenge (DSTC) series that were introduced to help the research community focus on the specific task and subsequently improve the quality of spoken dialogue systems (Williams et al., 2016). Specifically, our investigation is conducted with respect to the second and the third dialogue state tracking challenges (Henderson et al., 2014a,b), and is presented in three stages:

- **Data analysis -** We perform statistical tests on the dialogue data to determine whether there are indeed dependencies between slot variables and to what extent are these dependencies present.
- **Model development -** Tracking dialogue states while considering the relationships between target variables casts the problem into a structured prediction task. We develop a deep learning based tracker that incorporates an energy-based modelling approach that is notably efficient for structured predictions.
- **Result analysis -** Our model performance is evaluated and analyzed using a number of metrics to provide insights into the impact of variable dependencies on the dialogue state tracking process. We benchmark our energy-based approach against results for a number of state-of-the-art systems (Vodolan et al., 2017; Mrksic et al., 2015; Henderson et al., 2014c,d).

To our knowledge there has been no detailed analysis previously on the role of variable dependencies in dialogue states. The contributions of this paper are, thus, that systematic analysis, and our energy-based structured prediction model for dialogue state tracking.

2 Categorical Data Analysis

The investigation presented in this paper is predicated on the existence of associations between target variables in a dialogue state. Therefore, in this section we provide a concrete analysis of variable dependencies between domain slots in DSTC data.

2.1 DSTC 2 & 3 Datasets

The Dialogue State Tracking Challenge 2 & 3 datasets contain phone calls in the restaurant and tourism information domains (Henderson et al., 2014a,b). Within the datasets, the main task is referred to as *Joint Goals* and requires systems to estimate the value of each slot in the set of informable slots at every turn of the call. The value constraint is retrieved from the set of possible values predefined in a specified domain ontology.

The DSTC2 dataset is split into three subsets: 1612 dialogues for training; 506 for validation; and 1117 for a test set. The DSTC2 ontology predefines four informable slots.

The DSTC3 dataset contains 2275 dialogues that are not split into subsets; the dataset defines nine informable slots in the ontology. Four of the nine slot types also appear in the DSTC2 data, but the value sets are different.

A preliminary analysis shows that these slots are not equally distributed in both datasets (see Table 1). The informable slots are divided into two groups with one group including highly frequent slots ($f > 50\%$) and the other one containing very low frequencies ($f < 10\%$). Therefore, we follow the precedent of other researchers and design our models to track only highly frequent slots. Following this reduction, the DSTC2 *Joint Goals* consist of three slot-value pairs (*food*, *price range*, *area*), and DSTC3 *Joint Goals* consist of four slot-value pairs (*food*, *price range*, *area*, and *type*).

Slot	DSTC2		DSTC3	
	call	turn	call	turn
food	87.9	79.3	63.5	55.4
price range	73.5	62.6	68.3	60.8
area	81.8	72.3	59.5	50.6
type	-	-	98.5	91.0
name	0.8	0.5	1.5	0.6
near	-	-	8.5	6.8
has tv	-	-	7.3	5.8
has internet	-	-	7.6	5.9
children allowed	-	-	4.9	3.6

Table 1: The analysis of informable slot proportions (%) in DSTC 2 & 3 summarised over the number of calls and turns in the whole dataset.

2.2 Variable Dependencies

To test the independence of the slot variables, we apply Pearson's chi-square tests on labels of the informable slots in a pairwise fashion to generate bivariate statistics. The dependencies between slots are confirmed if and only if the significance

DSTC2		food - price	food - area	price - area
Chi-square	$\mathcal{X}^2$	9430.5	12739.0	3937.9
	$\mathcal{V}$	176	180	24
	p	< 2.2e-16	< 2.2e-16	< 2.2e-16
Coefficients	ϕ	0.6081	0.7068	0.3930
	C	0.5196	0.5772	0.3657
	V	0.2720	0.2671	0.1757

Table 2: Statistical tests on DSTC2 dataset.

DSTC3		food - price	food - area	food - type	price - area	price - type	area - type
Chi-square	$\mathcal{X}^2$	5792.6	7985.6	6762.5	5070.7	2873.0	3626.5
	$\mathcal{V}$	145	464	116	80	20	64
	p	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Coefficients	ϕ	0.5547	0.6513	0.5994	0.5190	0.3907	0.4389
	C	0.4851	0.5458	0.5141	0.4607	0.3639	0.4019
	V	0.2265	0.1580	0.2680	0.2119	0.1747	0.1963

Table 3: Statistical tests on DSTC3 dataset.

value $p < 0.05$. The chi-square test results are reported with the $\mathcal{X}^2$ statistic, degree of freedom $\mathcal{V}$, and statistical significance p. The statistic is calculated with the formula:

$$\mathcal{X}^2_{\mathcal{V}} = \sum_i \sum_j \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \qquad (1)$$

where O_{ij} and E_{ij} are observed and expected frequencies of categories i and j being activated for the observed variables at the same time in the whole dataset.

Furthermore, it is necessary to measure the strength of these dependencies as the chi-square test can only detect the presence of the dependencies without saying if they are strong or not. Fortunately, there exist several chi-square test-based measurements of association strength between variables. We report three such measures: ϕ coefficient, contingency coefficient C, and Cramer's V coefficient. All the coefficients are calculated through adjustment of the chi-square statistic to account for the dataset size; for instance:

$$V = \sqrt{\frac{\mathcal{X}^2}{N \min(r-1, c-1)}} \qquad (2)$$

where $\mathcal{X}^2$ is the chi-square statistic, N is the number of samples in the dataset, and r and c are the number of rows and columns in the contingency table. These measures are scaled between 0 and 1 indicating that 1 is the perfect relationship and 0 indicates the lack of any relationship between variables.

We report the statistics analysis of DSTC2 data in Table 2 and DSTC3 data in Table 3. In the results, all variables showed significance values $p < 0.05$, that indicate that there are indeed variable dependencies in the dialogue domains of the DSTC series. The association strength measured by the chi-square based coefficients show different level of variable dependencies ranging from a very strong dependency ($\phi \geqslant 0.7$, $V \geqslant 0.25$) to a moderate one ($0.3 \leqslant \phi < 0.39$, $0.11 \leqslant V < 0.15$). For example in DSTC2 data, the dependencies between the slot *food* and the other two, *price range* and *area*, are strong.

While the existence of dependencies across our labels may not be surprising, the consistency of their strong occurrence indicates that tracking systems could achieve more accurate results if judgements on trackable slots were made with reference to the information contained within hypotheses for neighbouring slots.

3 Energy-Based Structured State Tracking

The data analysis performed on the DSTC series data suggests that incorporating label dependencies in the dialogue state prediction process would be beneficial. Formally this indicates that we should cast the dialogue state tracking process as a structured prediction problem (Smith, 2009). This in itself should not be a surprise to the research community, as several researchers have built dialogue state trackers around models that can in principle be thought of as structured classifiers (Zhong et al., 2018; Hori et al., 2016; Jang et al., 2016; Ren et al., 2013).

One of the challenges for previous approaches to structured prediction for dialogue state classification is that they relied on methods that had difficulty integrating a structural component that took inter-slot dependencies into account with a robust underlying classifier that facilitated powerful feature representations from individual contributions to the dialogue. Recently the application of energy-based methods that are implemented through neural architectures have provided one promising avenue for structured prediction. The idea underpinning this approach is that we learn to rate the association between configurations of target variables and our inputs via a so-called energy function (LeCun et al., 2006) rather than attempt to learn to predict the structured output directly.

Below we first introduce the key principles behind energy-based structured prediction, then detail the energy-based dialogue state tracker that we have constructed.

3.1 Energy-Based Structured Prediction

Let us denote the input and structured output variables as X and Y respectively. For us, X can be thought of as the representation of a turn, while Y is a complete dialogue state representation – not the representation of an individual slot. Given X and Y, a function $E(X, Y)$ must be trained to assign some scalar value called energy to any configuration of variables X and Y. This function is called the energy function, and is traditionally designed to assign low energy to correct variable configurations, and higher energy to incorrect configurations. In other words we have low energy when a hypothesis for Y comes close to the ground truth given an input X. At run-time some interpretation process moves through the space of target configurations to find the most appropriate output configuration for a given input.

While the energy function can be thought of as some arbitrary scalar that is to be low for acceptable configurations, the form of the function and training of the function are important. Specifically the energy function takes the following form:

$$E(x, y, \theta) = E_{global}(y, \theta) + E_{local}(x, y, \theta) \quad (3)$$

where θ are trainable parameters of the energy network, $E_{global}(y, \theta)$ is the global energy term for labels y, and $E_{local}(x, y, \theta)$ is the local energy adjustment of both input and output variables. Thus the global energy function specifically considers the acceptability of configurations of the structured target, while the local energy estimates the appropriateness of the input with respect to individualised elements of the prediction.

During training the parameters θ for the energy function are estimated. This is most efficiently done by coupling the energy function to an oracle loss that estimates the loss between a hypothesised output Y and the ground truth label Y^* for a given input X.

Finding the parameters of a good energy function between X and Y directly is generally however not feasible, and historically was one of the key limitations for energy-based structured prediction. Instead it is generally more appropriate to first generate some feature function $F(X)$ that transforms the input to an appropriate representation form that better supports the inference process. Thus more commonly we denote the energy function as $E(F(X), Y)$. Both the feature representation and the energy function itself can be trained through a deep neural network model either dependently or independently.

3.2 Dialogue State Tracker

Based on the principles of energy-based structured prediction, we have designed an energy-aware dialogue state tracker. The framework for training and applying the energy-based method is based specifically on the Deep Value Network architecture proposed by Gygli et al. (2017). The architecture of our tracking model is illustrated in Figure 1.

The energy-based dialogue state tracker can be thought of as consisting of four key elements with associated training and inference processes; we detail these below.

3.2.1 Feature Function Network

The Feature Function Network $F(X)$ is a deep learning network to process raw DSTC dialogue data into a representation that is suitable for feeding into the energy network. As DSTC dialogues contain different input channels we implement different techniques to accommodate the variety of input variables.

In detail, each input of a dialogue turn consists

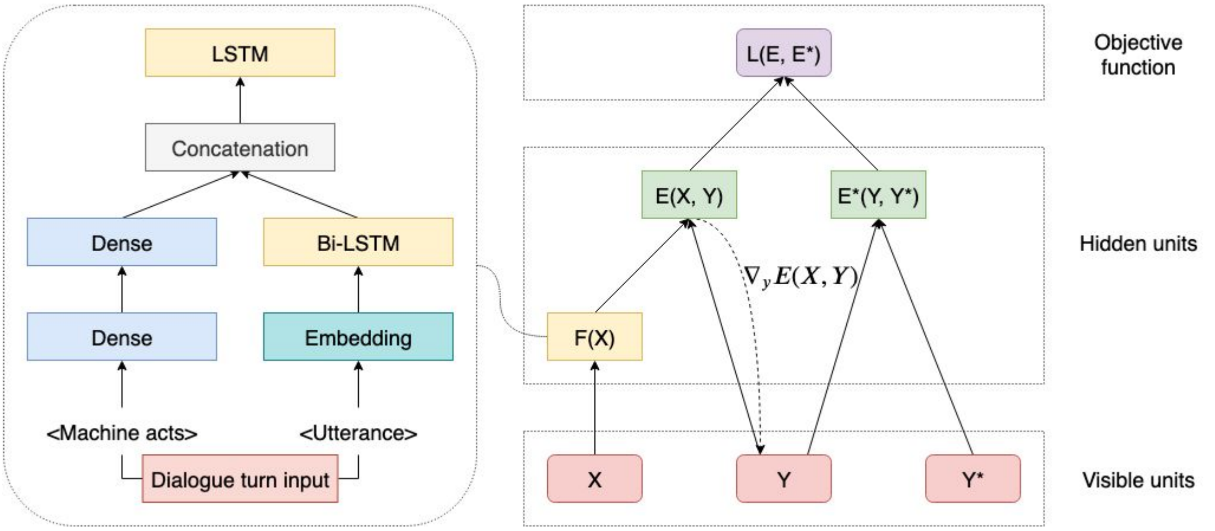

Figure 1: Deep Value Network-based Dialogue State Tracking Model.

of machine acts in a semantic format and user utterance transcribed by an automatic speech recognizer. We parse the machine dialogue acts with the parsing technique by Henderson et al. (2014d) before reducing the dimensionality of the machine act vectors with two dense neural layers. Meanwhile, all the words in user utterances are embedded with an online trained embedding layer, then passed into a bidirectional LSTM layer (Hochreiter and Schmidhuber, 1997). The output vectors of this bidirectional LSTM layer represent user utterances as real-valued tensors. Following that, the machine act and utterance vectors are concatenated, and fed into a unidirectional LSTM layer that processes dialogue by turn and returns fixed-size dialogue vector representations.

We pre-train this feature network as a multi-task classification model following the method proposed by Trinh et al. (2018). The dialogue representations retrieved from this network are treated as input features for the energy function.

3.2.2 Energy Function Network

The energy function network $E(F(X), Y)$ is implemented as a feed-forward network (Belanger and McCallum, 2016) where the general function form as illustrated in the previous section is hard coded and the parameters are acquired during training. Based on the energy function proposed by Belanger and McCallum (2016), the general forms of the global and local energy functions are:

$$E_{global}(Y) = W_2^\top f(W_1^\top Y) \tag{4}$$

$$E_{local}(X, Y) = \sum_{i=1}^{L} y_i W_i^\top F(X) \tag{5}$$

where $\theta = \{W, W_1, W_2\}$ are the energy network's trainable parameters, $f(\cdot)$ is a non-linearity function, $F(\cdot)$ is the feature function described in the previous section, and L is the number of classes in the target.

This energy network produces a scalar energy value that is the sum of global and local energy terms for each input and output configuration.

3.2.3 Oracle Function

To train the energy function we need a signal that estimates the compatibility of an input variable X with an output configuration Y. We achieve this by making use of an oracle function $E^*(Y, Y^*)$ that measures the quality of any output variable configuration Y with respect to the ground truth label Y^*. We implement the oracle value function in our model with the F_1 metric:

$$E^*_{F_1}(y, y^*) = \frac{2(y \cap y^*)}{(y \cap y^*) + (y \cup y^*)} \tag{6}$$

where $y \cap y^* = \sum_i \min(y_i, y_i^*)$; and $y \cup y^* = \sum_i \max(y_i, y_i^*)$, that are extended for continuous output variables.

3.2.4 Objective Function

To train and estimate the energy function, we make use of an objective function $L(E, E^*)$. This function calculates the error between predicted energy $E(X, Y)$ and ground truth energy value that is tied to the oracle value $E^*(Y, Y^*)$. Since the F_1 score falls into the range $[0, 1]$ we design the objective function as a cross entropy loss function:

$$L = -E^* \log E - (1 - E^*) \log(1 - E) \quad (7)$$

3.3 Training Process

The training process for the energy-based dialogue state tracking model is summarized in Algorithm 1. The learning objective is to train the energy function to predict correct quality of output by shaping the energy values to oracle F_1 values. All the trainable parameters of the network are updated via standard backpropagation techniques.

Algorithm 1: Learning process algorithm

```
Function TRAIN_EPOCH (dataset $\mathcal{D}$, initial
  weights $\theta$, learning rate $\lambda$)
    while not end of $\mathcal{D}$ do
        Training sample
        $(x, y^*) \in \mathcal{D}$
        Output generation
        $y \leftarrow GENERATE(x, \theta)$
        Ground truth energy
        $E^* \leftarrow E^*(y, y^*)$
        Predicted energy
        $E \leftarrow E(x, y, \theta)$
        Objective function
        $L \leftarrow L(E, E^*)$
        Backpropagation
        $\theta \leftarrow \theta - \lambda \nabla_\theta L$
    end
end
```

In detail, for each iteration in a training epoch we generate a batch of dialogues from the dataset. A structured output of each turn in the dialogue is then generated through an inference process (see Section 3.4). The system predicts energy terms for these variable configurations, and calculated oracle values as the ground truth energies. We compute the loss value of the batch, and backpropagate the model based on this loss.

3.4 Inference Process

In the training process a $GENERATE(\cdot)$ function was used to come up with a candidate value for Y given a network and input X. This generation process is based in part on the inference process that is used at both training time and run-time to determine a candidate Y for a given X.

The inference process predicts structured output starting from a random initial prediction. The inference process is based on the principle that the gradient of energy with respect to Y can be calculated directly and used to direct a process for selecting Y.

In short, this prediction is generated through an inference loop with the gradient ascent technique for a number of steps:

$$y^{(t+1)} = \mathcal{P}_\mathcal{Y}\Big(y^{(t)} + \eta \nabla_y E(x, y^{(t)}, \theta)\Big) \quad (8)$$

where $\mathcal{P}_\mathcal{Y}$ is the projection operation to shape the predicted output to the output variable space $Y = \{y_i\}^L \in \{[0, 1]\}^L$, and η is the learning rate for gradient ascent.

4 Experimental Design

To evaluate the usefulness of the energy-based approach we implemented and trained a tracker based on the model outlined in the previous section against both the DSTC2 and DSTC3 datasets. Training is a two phase process. First, we trained the feature network independently of the energy-based components by casting the feature network as a standard multi-task learning system where each target variable is assumed to be independent of the others. We present the results of this multi-task based model independently, but critically we also then make use of the trained network prior to the output layer as the feature network that is available for training the full energy network. Thus, the second stage of training targets the parameterisation of the energy network once the feature network has already been learned.

As mentioned, the DSTC2 dataset is divided into three subsets for training, validation, and test purposes, while DSTC3 data are provided in a whole set only for the test purpose. Thus we apply DSTC2 directly, but split the DSTC3 dataset into five folds and use cross-validation in the training process. All experiments are run for at least five times to ensure the stability of our results; we report the average performance.

5 Result & Error Analysis

We report our multi-task and energy-based models performance on both the DSTC2 and DSTC3 datasets, and benchmark their performance against state-of-the art systems in Table 4. A state-of-the-art system is selected if it produces highest to date

Model	Entry	DSTC2	DSTC3
Hybrid system (Vodolan et al., 2017)		0.796	-
Web-style ranking system (Williams, 2014)	✓	0.784	-
Multi-domain system (Mrksic et al., 2015)		0.774	0.671
Word-based system (Henderson et al., 2014d)	✓	0.768	-
Unsupervised RNN-based system (Henderson et al., 2014c)	✓	-	0.646
Our work			
Multi-task feature system		0.709	0.531
Energy-based system		0.760	0.622
DSTC baseline	✓	0.719	0.575

Table 4: Performances of state-of-the-art and our dialogue state tracking systems on DSTC 2 & 3 data. The results for *Joint Goals* are reported with Accuracy metric featured in the challenge. The column *Entry* marks the systems submitted to blind evaluation during the competition period.

accuracy on the *Joint Goals* task either during the DSTC competition time or after the competition. We also included the model by Henderson et al. (2014d,c) as it includes data processing techniques that we adopted in our work.

Overall, we find that applying an energy-based algorithm on top of the LSTM enabled slot tracker improves the dialogue state tracking results in term of accuracy by a big margin, 5% for DSTC2 and 9% for DSTC3.

Comparing our work with state-of-the-art systems like the hybrid tracker (Vodolan et al., 2017) and a multi-domain system (Mrksic et al., 2015), the energy-based model has not yet reached their level of performance. However, the hybrid tracker also consists of a feature network and an algorithm to refine the prediction. Vodolan et al. (2017) designed this algorithm with a set of manual rules, while we design the refinement with a deep neural structure and let it learn from the data. With respect to the multi-domain system, we believe it outperforms our energy-based model because of the wider range of data processed by the multi-domain system. Mrksic et al. (2015) trained and combined their models on six datasets of different domains, while we train our energy-based system on a single domain at a time only.

It is also important to note that the web-style ranking system of Williams (2014) was the best entry during the DSTC2 competition, and is not neural-based. It is followed by the word-base tracker (Henderson et al., 2014d) that was developed with a special recurrent neural network architecture. Besides, the word-based system is also notable for its feature parsing technique that is reused in a number of later systems (Henderson et al., 2014c; Vodolan et al., 2015, 2017; Trinh et al., 2017, 2018) and our work.

5.1 Slot-based Result Analysis

We argue that the feature Accuracy metric in the DSTC series do not provide a full picture of how well a model performs for each slot. Therefore it is necessary to evaluate our work for the individual slots as well as for the joint dialogue states. We conduct a separate evaluation on the result track file and report it in Table 5. Overall our models achieve high accuracy across all informable slots and the joint goals. Here the joint goals accuracy is higher than evaluated with the DSTC evaluation scripts due to the absence of low frequent slots that we omit in our experiments.

We observe that the energy-based model improves the tracking results of all slots both as individual and a joint set. The improvement margin of joint goals is similar to the results measured by the DSTC feature Accuracy metric. The tracking result of individual slots varies from a very small change of 0.3% to a big jump of 7%. These change differences are related to the relative difficulties of the slot. For example slot *food* has the biggest set of possible values, which in turn makes it the most difficult slot to track; it is for this slot that we see the greatest improvement.

5.2 Proportional Reduction in Error

Proportional reduction in error is a statistical test to measure association between two variables on how one can influence the other in the prediction process. For example given variables A and B, this method attempts to evaluate the prediction of A in two ways: predicting A independently; and predicting A with the knowledge of B. Reduction

Dataset	Model	Slot				Joint goals
		food	price	area	type	
DSTC2	Feature system	0.825	0.929	0.919	-	0.717
	Energy-based	0.872	0.938	0.923	-	0.768
DSTC3	Feature system	0.730	0.844	0.781	0.937	0.587
	Energy-based	0.802	0.860	0.817	0.940	0.666

Table 5: Performances of our energy-based dialogue state tracking system. The results are reported per slot and for Joint slots of those present in the task.

in error can be formulated mathematically.

$$\lambda = \frac{E_A - E_{A|B}}{E_A} \tag{9}$$

where E_A is the number of errors in predicting A, and $E_{A|B}$ is the number of errors in predicting A while taking into account B. All errors are assumed to be absolute numbers.

From this formula we can see that λ has the value in the range $[0, 1]$ because $E_{A|B} \leq E_A$ in all cases. If $\lambda = 0$, A and B are completely independent, thus knowing B does not help predicting A better. On the other hand, when $\lambda = 1$, the relationship between A and B is absolute, i.e., that the knowledge of B gives us the perfect prediction of A.

To apply this statistical method in our model performance evaluation, we treat the prediction of the multi-task feature system as the independent prediction of variable A, since the output is produced without the variable dependencies. On the other hand, we think that the energy-based model gives prediction similar to prediction of variable $A|B$, where B acts as variable associations. We calculate the reduction in error by counting the absolute number of errors for each slot and the joint slot set of both our systems. The test result is reported in Table 6.

Dataset	Slot				Joint
	food	price	area	type	
DSTC2	0.27	0.13	0.04	-	0.18
DSTC3	0.27	0.10	0.16	0.05	0.19

Table 6: Proportional reduction in error of the energy-based system for each slot and the joint goals.

The analysis shows that for more challenging slots such as *food*, the energy-based model reduces the error rate significantly. In both DSTC 2 & 3 a quarter of errors for *food* are corrected, subsequently the errors in joint goals are reduced by nearly 20%.

6 Conclusion

In this paper our contributions were two-fold. We demonstrated, through a number of statistical tests performed on dialogue data and an empirical analysis on variable associations presented in dialogue domains, that dependencies between variables exist and taking them into account improves system performance. We also demonstrated how variable dependencies can be addressed in dialogue state tracking through a structured prediction methodology, and verified our model with respect to the second and third DSTC datasets. While our results do not directly improve on the state of the art, we showed a significant improvement over a non-trivial baseline. We therefore argue that the methodology is promising, and if applied to what is already a state-of-the-art methodology, may help to improve existing systems beyond the state-of-the-art.

There are a number of elements of this work that we are looking to improve. At a fine level we are looking at refinements of the energy-based deep learning architecture and are considering in particular variations on our selected oracle and objective functions that would be better aligned with the multi-categorical nature of our target variables. Meanwhile, at a higher level we want to generalise and further substantiate our investigation by applying the energy-based tracking methodology to tracking architectures that already show state-of-the-art or very near state-of-the-art performance. Finally, we note that a key benefit of this structured methodology is that it allows a more holistic tracking process for the user to be considered where tracking aspects of personality and preference can be neatly integrated alongside the tracking of fine-grained dialogue state. Our longer term goal is thus to apply the structured learning approach in the context of user intent and preference tracking.

Acknowledgments

This research was supported by the ADAPT Centre for Digital Content Technology which is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

References

David Belanger and Andrew McCallum. 2016. Structured Prediction Energy Networks. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48.

David DeVault. 2008. *Contribution tracking: Participating in task-oriented dialogue under uncertainty*. Phd dissertation, State University of New Jersey.

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: A Corpus for Adding Memory to Goal-Oriented Dialogue Systems. In *Proceedings of the SIGDIAL 2017 Conference*, pages 207–219.

James R. Glass. 1999. Challenges For Spoken Dialogue Systems. Technical report, Massachusetts Institute of Technology.

Michael Gygli, Mohammad Norouzi, and Anelia Angelova. 2017. Deep Value Networks Learn to Evaluate and Iteratively Refine Structured Outputs. In *Proceedings of the 34th International Conference on Machine Learning*.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014a. The Second Dialog State Tracking Challenge. In *Proceedings of the SIGDIAL 2014 Conference*, pages 263–272.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014b. The Third Dialog State Tracking Challenge. In *Proceedings of 2014 IEEE Workshop on Spoken Language Technology*, pages 324–329.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014c. Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation. In *Proceedings of 2014 IEEE Workshop on Spoken Language Technology*, pages 360–365.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014d. Word-Based Dialog State Tracking with Recurrent Neural Networks. In *Proceedings of the SIGDIAL 2014 Conference*, pages 292–299.

Sepp Hochreiter and Jurgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Takaaki Hori, Hai Wang, Chiori Hori, Shinji Watanabe, Bret Harsham, Jonathan Le Roux, John R. Hershey, Yusuke Koji, Yi Jing, Zhaocheng Zhu, and Takeyuki Aikawa. 2016. Dialog State Tracking With Attention-Based Sequence-To-Sequence Learning. In *Proceedings of 2016 IEEE Workshop on Spoken Language Technology*, pages 552–558.

Youngsoo Jang, Jiyeon Ham, Byung-Jun Lee, Youngjae Chang, and Kee-eung Kim. 2016. Neural Dialog State Tracker for Large Ontologies by Attention Mechanism. In *Proceedings of 2016 IEEE Workshop on Spoken Language Technology*, pages 531–537.

Frédéric Landragin. 2013. *Man-Machine Dialogue: Design and Challenges*. ISTE Ltd and John Wiley & Sons, Inc.

Yann LeCun, Sumit Chopra, Raia Hadsell, Marc' Aurelio Ranzato, and Fu Jie Huang. 2006. A Tutorial on Energy-Based Learning. *Predicting Structured Data*.

Bing Liu, Gokhan Tur, Dilek Hakkani-Tur, Pararth Shah, and Larry Heck. 2018. Dialogue Learning with Human Teaching and Feedback in End-to-End Trainable Task-Oriented Dialogue Systems. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2060–2069.

Nikola Mrksic, Diarmuid O'Seaghdha, Blaise Thomson, Milica Gasic, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2015. Multi-domain Dialog State Tracking using Recurrent Neural Networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 794–799.

Nikola Mrksic, Diarmuid O'Seaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve Young. 2017. Neural Belief Tracker: Data-Driven Dialogue State Tracking. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Tim Paek and Eric J. Horvitz. 2000. Conversation as Action Under Uncertainty. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, pages 455–464.

Julien Perez and Fei Liu. 2017. Dialog state tracking, a machine reading approach using Memory Network. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 1, pages 305–314.

Abhinav Rastogi, Dilek Hakkani-Tur, and Larry Heck. 2017. Scalable Multi-Domain Dialogue State Tracking. In *Proceedings of 2017 IEEE Automatic Speech Recognition and Understanding Workshop, ASRU*, pages 561–568.

Antoine Raux, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi. 2005. Let's Go Public! Taking a Spoken Dialog System to the Real World. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 885–888.

Hang Ren, Weiqun Xu, Yan Zhang, and Yonghong Yan. 2013. Dialog State Tracking using Conditional Random Fields. In *Proceedings of the SIGDIAL 2013 Conference*, pages 457–461.

Liliang Ren, Kaige Xie, Lu Chen, and Kai Yu. 2018. Towards Universal Dialogue State Tracking. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2780–2786.

Robert J. Ross and John Bateman. 2009. Daisie: Information State Dialogues for Situated Systems. In *Proceedings of International Conference on Text, Speech and Dialogue, TSD 2009*, pages 379–386.

Noah A. Smith. 2009. Structured Prediction for Natural Language Processing. In *The 26th International Conference on Machine Learning, ICML. Tutorial*.

Anh Duong Trinh, Robert J. Ross, and John D. Kelleher. 2017. Incremental Joint Modelling for Dialogue State Tracking. In *Proceedings of the 21st Workshop on the Semantics and Pragmatics of Dialogue*, pages 176–177.

Anh Duong Trinh, Robert J. Ross, and John D. Kelleher. 2018. A Multi-Task Approach to Incremental Dialogue State Tracking. In *Proceedings of The 22nd workshop on the Semantics and Pragmatics of Dialogue, SEMDIAL*, pages 132–145.

Miroslav Vodolan, Rudolf Kadlec, and Jan Kleindienst. 2015. Hybrid Dialog State Tracker. In *Proceedings of the Machine Learning for SLU & Interaction NIPS 2015 Workshop*.

Miroslav Vodolan, Rudolf Kadlec, and Jan Kleindienst. 2017. Hybrid Dialog State Tracker with ASR Features. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL*, volume 2, pages 205–210.

Nigel G. Ward and David DeVault. 2015. Ten Challenges in Highly-Interactive Dialog Systems. In *AAAI Spring Symposium on Turn-Taking and Coordination in Human-Machine Interaction*, pages 104–107.

Jason D. Williams. 2014. Web-style ranking and SLU combination for dialog state tracking. In *Proceedings of the SIGDIAL 2014 Conference*, pages 282–291.

Jason D. Williams, Antoine Raux, and Matthew Henderson. 2016. The Dialog State Tracking Challenge Series: A Review. *Dialogue & Discourse*, 7(3):4–33.

Victor Zhong, Caiming Xiong, and Richard Socher. 2018. Global-Locally Self-Attentive Dialogue State Tracker. In *Proceedings of the 56th Annual Meeting ofthe Association for Computational Linguistics*, pages 1458–1467.