PH.D. RESEARCH PROPOSAL

SEQUENCE LEARNING USING DEEP NEURAL NETWORKS WITH FLEXIBILITY & INTERPRETABILITY

Chang LI

Supervisor
Prof. Dacheng TAO

1 Aims & Objectives

1. Flexibility in modeling complex patterns with long-range dependencies
   (a) Capturing complex non-linear correlations without prior knowledge and assumptions
   (b) Encoding high dimensional input variables adaptively
   (c) Discovering long term dependencies of encoded inputs
2. Network architectures with better computational properties
3. Decoding/Encoding human intepretable representations from/into models
   (a) Approximating deep neural networks' input-output relationships using human intepretable models
   (b) Learning deep neural networks coupled with structured (potentially predefined) latent variable sequence

2 Synopsis

3 Background And Literature Review

One interesting task in machine learning is modeling complex sequences having long-term dependencies. Many applications such as machine translation, complex dynamical system analysis, activity recognition and behavioral phenotyping tools for neuro-science involved with capturing non-linear patterns in sequences. Sequence learning mainly have three difficulties: approximating non-linear relationship among sequences, feature selection and capturing long-term dependencies. Our primary aim of this proposal is mainly focused on demonstrating those three difficulties.

Despite substantial effort has been made for modeling sequences, many of those models are neither unable to approximate non-linear relationships nor have rigid assumptions due to the dependency on predefined form of prior function. For example, autoregressive moving average (ARMA) model [10] and many of its variants [3] have raised interests because of their effectiveness in many real world applications. However, those models cannot capturing non-linear relationships. Probabilistic Graphical Models [13, 14] (PGM) are very well studied for the past decades. With predefined prior distributions from domain knowledge, PGMs are capable to encode many sequence relationships such as Gaussian Processes [5], Hidden Markov Models [4] (HMM) and Linear Dynamic Systems [2] (LDS). However, capabilities of approximating non-linear patterns of PGMs are also severely suffered from rigid assumptions over priors.

Deep Neural Networks [9] (DNNs) are powerful and flexible models that have outperforming performance on various difficult learning tasks such as image classification, visual object recognition, machine translation and speech recognition. It can take high dimensional data with rich structure as input and scales over large data-set. DNNs are usually composed of hierarchical layers which contains large amount of latent variables. Non-linearity in the data is usually captured by non-linear interactions through those latent variables. Each of these latent variables normally connected to many other variables in adjacent layers. Results of those unique distributed representations are that DNNs are extremely flexible in fixing difficult problems in high dimensional space and have very generic learning algorithm (Stochastic Gradient Descent, SGD) for various problems. With sufficient training data, DNNs usually can get nice approximation of that information in a reasonable amount of time [9].

Despite the powerful capability of approximation, one significant limitation of DNNs is that they require fixed dimension of inputs and targets. Thus DNNs cannot be applied directly on areas without prior knowledge about variables' length or have various input length, such as speech

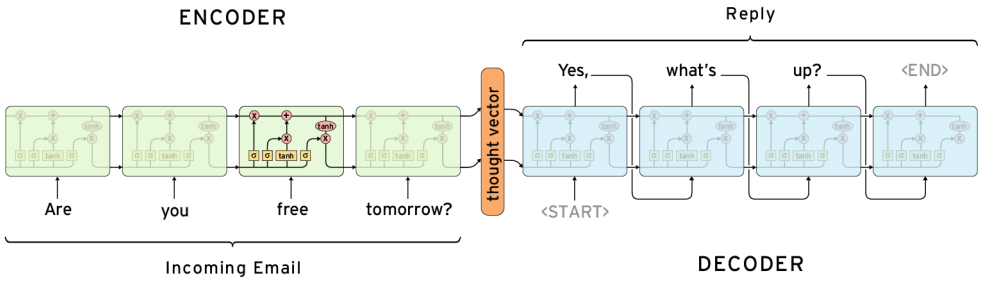

Figure 1: Encoder-Decoder Network

recognition and machine translation. Sutskever et al. [17] in 2014 demonstrated this difficulty as sequence to sequence problems. They introduced an encoder-decoder framework based on two Recurrent Neural Networks (RNNs) and achieved a very successful result in machine translation. The key ideas (as shown in Figure 1) behind encoder-decoder networks and its variants are that they first encode the whole sentence of words into a single, fixed length variable using encoder network. Then a decoder network is used to decode that variable and generate translation. They enabled the network to take various length of inputs by encoding each of them into a single and fixed length variable. One major drawback is that "indeed the performance of a basic encoder–decoder deteriorates rapidly as the length of an input sentence increases [1]". Since for many sequence learning applications there exists many long-term dependencies relationships, the question of solving this drawback remains open.

Other than long-term dependencies difficulty, DNNs contain a very large number of latent variables which make the training results very hard for a human to interpret [9]. Since DNNs are capable of approximating a very large number of insignificant correlations between inputs and outputs of the training data, there is no way to distinguish the true relationships exist in data from suspicious correlations which are caused by sampling bias in data set [9]. Being able to explain the reasons behind such models plays a crucial role in tasks such as model comparison and deployment. It could also provide insights into the data set and model itself. Thus improving the interpretability of DNNs could also be a very interesting topic to investigate.

4 Proposed Methodology

We divide this project into three phases.

To commence we will investigate the mechanism of long-term dependencies in time-series and how to model them using deep neural networks. We will also investigate optimization algorithms which can jointly optimize neural networks together with feature selection.

With optimization algorithms in hand, at the second stage we will try to extend those algorithms into various network topologies such as CNNs and simple feed-forward neural networks. The goal of the second stage is to investigate novel network topologies which can preserve good approximation performance while having better computational properties, such as parallelization.

At the final stage, we will explore a large group of conventional machine learning methods (lasso regression, random forest, graphical model, etc.) aiming at explaining the learning results of neural networks. Outcomes at this stage not only improves the interpretability of neural networks but also give insights on how to encode expert prior knowledge into neural networks.

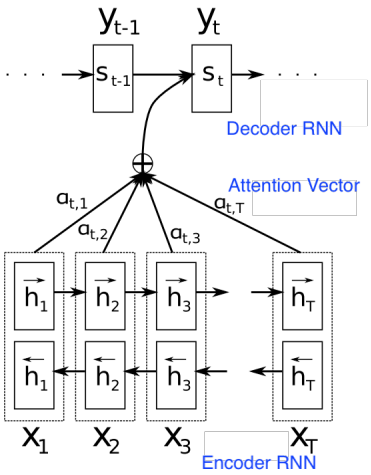

Figure 2: Attention Mechanism

4.1 Modeling Long-term Dependencies And Feature Selection

As discussed in section 3, encoder-decoder networks based on RNNs outperform many other statistical machine learning models while suffering from short-term memories due to encoding global information into a single fixed length variable. One of the most popular ways is to introduce "attention mechanism" into encoder-decoder networks [1].

Bahdanau et al. [1] demonstrated long-term dependencies issue by introducing context vectors in the middle of encoder network and decoder network (as shown in Figure 2). Instead of encoding the global information into a single variable, attention mechanism enables encoder network to provide a sequence of encoded vectors on which decoder network can do soft-searches. As a result, decoder network is able to concentrate on the most relevant information represented by a subset of encoded vectors of arbitrary length. Thus encoder-decoder networks can have much better performance when dealing with long-term dependencies.

Qin et al. [15] extended Bahdanau et al. [1]'s work to adaptively select most relevant hidden states as well as most significant features simultaneously. The developed a dual stage RNNs (see Figure 3). Other than attention vector between encoder and decoder layer (so called "Temporal attention Layer" in Figure 3(b)), they adaptively weights input sequence x_t^k which is the kth feature at t step by attention α_t^k (see Figure 3(a)). The input attention mechanism is a feed-forward network. Thus the extended encoder-decoder network can weight features according to their importance while the whole network can still be trained jointly. However, their work is focused on regression. We will further investigate their methods by extending to classification cases.

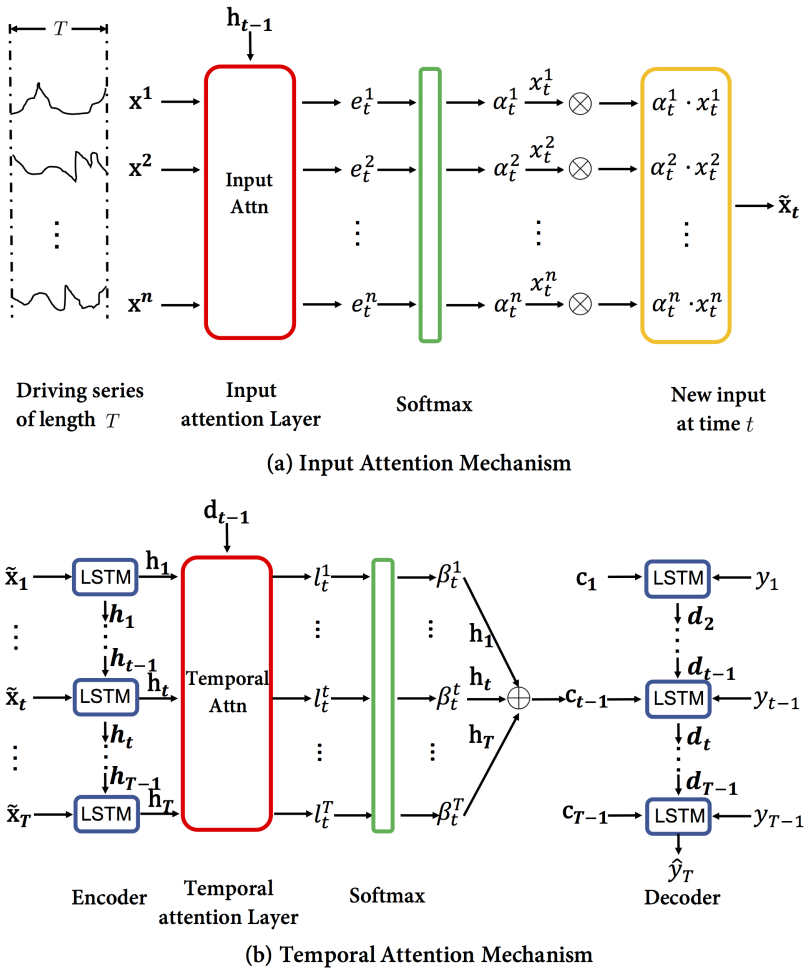

Figure 3: Dual-Stage Attention Based RNN

4.2 Topologies With Better Computational Properties

One major drawback of RNNs formulation is computationally expensive. RNNs usually have a very deep recursive structure. It maintains hidden states of the entire past which prevents parallel implementation of the algorithm. On the contrary, Convolutional Neural Networks (CNNs) have a hierarchical structure and can fully exploit the GPU hardware by using parallel computing techniques. Gehring et al. [7] attempted to replace RNNs with CNNs in sequence modeling task by using a hierarchical structure CNNs. They captured correlations in length of n sequence by applying $\mathcal{O}(\frac{n}{k})$ convolutional operations for kernels of width k and constructed both of encoder and decoder networks using CNNs. Results of their model are slightly better than conventional encoder-decoder networks [1] on multiple data set but with a huge decreasing in training time.

Recently, numerous pure attention architecture have been proposed and achieved the state-of-the-art results in natural language processing problems. Vaswani et al. [18] dispensed CNNs and RNNs entirely and proposed a self-attention-based model. They extends the conventional attention mechanism to a so-called Multi-Head Attention (MHA) architecture by allowing each head to generate different distributions over encoder output. This novel network architecture has very concise topology representation and outperforms both RNNs and CNNs based sequence

to sequence [18].

In this project we plan to further investigate those network topologies. Besides, the question of creating attention mechanism in input layers (feature selection) in those architectures still remains open.

4.3 Interpretability

Substantial effort has been made for improving DNNs interpretability in recent years. There are mainly three different types of approaches:

- Explain local predictions directly
- Approximating DNNs using human interpretative models
- Joint learning probabilistic graphical models with DNNs

Ribeiro et al. [16] proposed a Local Interpretative Model-agnostic Explanations (LIME) method to interpret arbitrary machine learning models directly by explain individual outputs. They tried to approximate a human interpretative local estimator (Lasso in their case) by sampling around models' outputs and explain the reason behind those decisions by observing which features (input) are taking most responsibility in the local estimator. However, such approaches are limited to providing explanations for individual decision. A more challenging task would be providing a global insights of models.

Instead of explaining local predictions, Frosst and Hinton [6] use the DNNs to train a soft decision tree. One main difficulty of such model is that it requires exponentially large amount of training data with respect to the depth of the tree. However, with approximated DNNs in hand, sufficient amount of training data for soft decision tree can be provided by labeling unlabeled data using DNNs, sampling synthetic unlabeled data using generative approach [8] and distilling method [11]. Such soft decision tree could approximate DNNs to a reasonable extent while maintain human interpretability for further investigation.

More advanced approaches combines complementary strengths of Probabilistic Graphical Models (PGMs) and DNNs. As discussed above, PGMs have very easy to understand structures. Other than interpretability, they are also easy to fit and data efficient. The main drawbacks of PGMs is that the rigid assumptions severely limit PGMs capability of approximating non-linear relationship. On the contrary, DNNs are highly capable of approximation while are data intensive and hard to explain. Johnson et al. [12] proposed a combination of flexible deep learning feature models with structured Bayesian priors. They used deep auto-encoders to extract lower representation of non-linear observed data and used graphical models for latent variables sequence inference. A very efficient variational inference algorithm was developed to solve the joint learning problem. A very interesting view of their work is that other than providing better explanations, they also encoded human expert knowledge into DNNs through PGMs [12]. Investigation in this topic could provide very interesting insights for both DNNs and PGMs researches.

5 Expected Research Contribution

6 Work Plan

References

[1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[2] Y. Bar-Shalom and X.-R. Li. Estimation and tracking- principles, techniques, and software. *Norwood, MA: Artech House, Inc, 1993.*, 1993.
[3] P. J. Brockwell and R. A. Davis. *Time series: theory and methods*. Springer Science & Business Media, 2013.
[4] S. R. Eddy. Hidden markov models. *Current opinion in structural biology*, 6(3):361–365, 1996.
[5] R. Frigola, Y. Chen, and C. E. Rasmussen. Variational gaussian process state-space models. In *Advances in neural information processing systems*, pages 3680–3688, 2014.
[6] N. Frosst and G. Hinton. Distilling a neural network into a soft decision tree. *arXiv preprint arXiv:1711.09784*, 2017.
[7] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122*, 2017.
[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
[9] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.
[10] M. Hibon and S. Makridakis. Arma models and the box–jenkins methodology. 1997.
[11] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
[12] M. Johnson, D. K. Duvenaud, A. Wiltschko, R. P. Adams, and S. R. Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.
[13] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.
[14] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.
[15] Y. Qin, D. Song, H. Cheng, W. Cheng, G. Jiang, and G. Cottrell. A dual-stage attention-based recurrent neural network for time series prediction. *arXiv preprint arXiv:1704.02971*, 2017.
[16] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016.
[17] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.
[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.