Literature Survey of Deep Learning for Natural Language Processing

Xiaoqiu Huang
School of Computer Science
Carnegie Mellon University
Email: xiaoqiuh@andrew.cmu.edu

I. Introduction

Usually, many shallow-strutured achitectures, such as kernel perceptron, logistic regression, SVM and so on, are utilized in natural language processing. Due to the high efficiency and good performance, they are still very popular in NLP field. However, these algorithms with simple models and limited represention power cannot achieve impressing results in dealing with more complicated tasks. In order to extract more complex features and better represent human language, multiple layers models or deep structure models are designed to handle more complicated human language problems.

The purpose of deep learning is to extract more features from human languages through adopting neural networks with deep structure. So the idea of deep learning originates from artificial neural network(ANN).The combination of feed-forward neural network and back propagation(BP) algorithm[1] make the neural network a good model for learning the representation of natural language. (works concerning NN and NLP). However, BP has many shortcomings which limit its ability in learning features. The most important is that as the number of layers becomes large, it is hard for BP to propagate the error from hidden layers to input layers. In another words, BP algorithm is a optimization method based on optimal local search and easy to achieve local optimum as the structure becomes deep.

However, the algorithm of deep learning can not only adopt deep struture but also achieve better optimization results. In 2006, a fast learning algorithm for deep belief nets(DBN) was proposed by Hinton et al.[2]. Instead of using BP algorithm to optimize all of weights in each layers, a greedy learning algorithm was adopted to train the model layer by layer. But since learning one layer at a time is not optimal, a wake-sleep algorithm was utilized to fine-tune the weights of neural network. In the 2007, another paper proposed a unsupervised deep neural network model[3]. In this model, each layer is trained an auto-encoder. As the idea become popular, many different deep structures are adopted in deep neural networks, such as recursive neural network, recurrent neural network and so on.

In this literature survey, we will first introduce the neural language model which is the basis of most NLP applications. Then the applications which adopt deep learning to handle the problem of NLP will be presented. There are mainly three deep structures in NLP field: recursive neural network, recurrent neural network and autoencoder. In our survey, we will first enumerate the applications of different structures and then focus on applications concerning neural language model. At last, we will compare it with traditional neural network models. The organization of the literature survey is shown as follows: In section 2, the neural language model will be detailedly discussed. In section 3, we will introduce three deep learning structures and the NLP problems that these models can handle. In section 4, different deep learning models for neural language will be compared in the experimental results to prove the efficiency of deep structure. In section 5, we will conclude the effectiveness of deep learning model and propose some issues which have not been handled by this model.

II. Neural language model

Previously, the most popular model utilized in NLP is n-gram model. However, this model suffers from the curse of dimensionality. A word sequence during testing may be different from any sequences during training. Therefore, Bengio et al.[4] proposed a new neural language model which generates distributed presentations for each word in the corpus. In that model, the next term in the text is generated by previous terms with certain probability. Therefore, they want to model the joint probability $\hat{P}(Z_1 = z_1, \cdots, Z_n = z_n) = \prod_i \hat{P}(Z_i = z_i | g_i(Z_{i-1} = z_{i-1}, Z_{i-2}, z_{i-2}, \cdots, Z_1 = z_1))$.

The nerual network model for neural language is shown as follows:

In order to model the probability, probability is decomposed into two parts:

- A mapping C from term i to a real vector $C(i) \in R^m$ is used to transform a single word into a distributed representation, which is a $|V| \times m$ matrix.

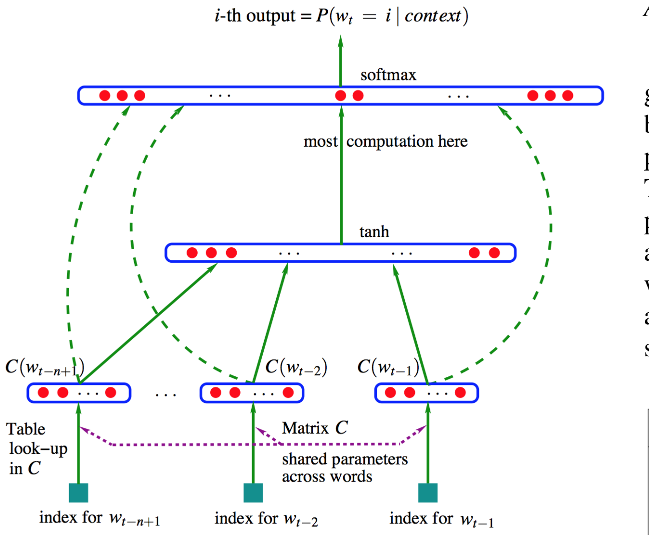

Fig. 1. Neural architecture

Here, $|V|$ denotes the size of vocabulary and m means the number of features.

- After mapping words into vectors, we can further utilize function g to estimate the probability $\hat{P}w_t = i|w_1^{t-1}$.

As we can see from the figure, the function of the output layer is a softmax function which can compute the probability of each possible term. In sum, the probability of next term can be estimated as follows:

$$f(i, w_{t-1}, \cdots, w_{t-n+1}) = g(i, C(w_{t-1}, \cdots, C(w_{t-n+1}))) \quad (1)$$

As mentioned before, the objective of the model is to maximize the probability $f(w_t, \cdots, w_{t-n+1})$. And the parameter of the model is composed of two parts. One is C in the mapping. Another is the weight of neural network ω, which is also the parameter of function g. The overall parameter $theta$ equals (C, ω). So we need to maximize the penalized log-likelihood as follows:

$$L = \frac{1}{T}\sum_t \log f(w_t), w_{t-1}, \cdots, w_{t-n+1}; \theta + R(\theta) \quad (2)$$

Finally, stochatic gradient ascent can be used to learn the parameter of the model. Through optimizing this model, the word vector of each term can be generated.

III. Three structures of deep learning in NLP

Since human language is a kind of sequence data and has a specific grammar structure. Therefore, many structures of network are designed based on the sequences of terms and the gammar tree of sentences.

A. *Recursive neural networks*

It is well known that human language has a recursive grammar structure. For example, two noun terms can be combined as a noun phrase. And different kinds of phrases can also combined as another kind of phrase. Therefore, a recursive neural network, which is composed of several neural units with the same structure, is adopted in this scenario. In order to explain the structure, we take the application concerning language parsing[5] as an example. We first observe the recursive structure shown as follows:

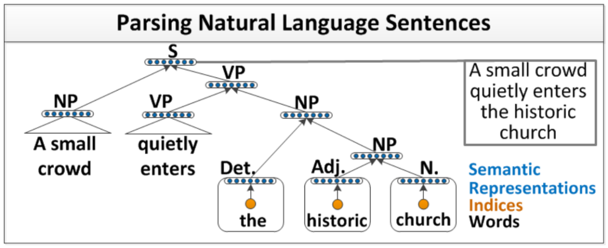

Fig. 2. Parsing natural language sentences

In this model, the input of neural network is the word vector which is mentioned above. And two adjacent vectors are combined together to form a new vector through the same neural network. After all terms are merged, a gammar tree is formed.

I think recursive neural network has a totally different structure from the deep brief net and autoencoder. It doesn't train the model layer by layer. However, this model can also handle the optimization problem of deep structure because the same neural network structure is merged into the whole recursive neural network. And the weight of all basic neural networks is the same. Therefore, it is more easy to optimize the parameter.

B. *Autoencoder*

Autoencoder is a neural network model that learns the function $f(x) = x$ in order to learn the reduced representation of the input text. In this model, the network is trained layer by layer. In order to introduce the autoencoder model, the application about sentiment analysis[6] will be discussed. As we can see from the figure, the input of the model is also the distributed representation. After being processed by the hidden layer, the output layer reconstructs the input vector. Through this process, the reduced representation of the input vectors can be learned. In order to measure the performance of the representation, the reconstruction error is computed as the distance between the input vector and output vector. Also we can assign different terms with different

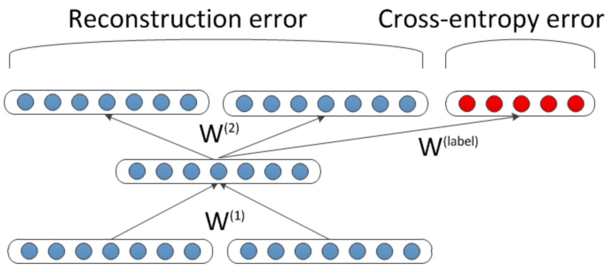

Fig. 3. Autoencoder

weights. So

$$E_{rec}([c_1; c_2]; \theta) \\ = \frac{n_1}{n_1+n_2}||c_1 - c_1'||^2 + \frac{n_2}{n_1+n_2}||c_2 - c_2'||^2 \quad (3)$$

, where n denote the number of words. In addition, c denotes the vector of input and c' present the vector of output.

Also, in order to predict the sentiment of the sentence, an extra output unit is added to generate the sentiment label. And this generated label will be compared with the correct label. A softmax function is utilized in this scenario to generate the probability of each label. And author use the cross-entropy error to measure the correctness of the model.

$$E_{cE}(p, t; \theta) \\ = -\sum_{k=1}^{K} t_k \log d_k(p; \theta) \quad (4)$$

, where t_k denotes the target label and d_k denotes the probability of each label.

After combining the two process, we can compute the reconstruction error and cross-entropy error. And the objective is to minimize the weighted sum of the two errors as follows:

$$E([c_1; c_2]s, ps, t,) = \\ = \alpha E_{rec}([c_1; c_2]_s; \theta) + (1-\alpha)E_{cE}(p_s, t; \theta). \quad (5)$$

IV. Deep learning based neural language model

A. Convolution neural network

The structure of convolution neural network can be utilized to generate neural language model[7][8]. The model of this paper tries to compute the score of the window with n words. If the score $f(w_{t-n+1}, \cdots, w_t)$is higher, then the sentence is more normal. Otherwise, it means that this sentence may be generated randomly. With this assumption, the author use the convolution neural network to train the model. More specifically, the purpose is to minimize the objective function as follows:

$$\sum_{x \in X} \sum_{w \in D} max0, 1 - f(x) + f(x^w) \quad (6)$$

, where X denote the n-gram terms, D means the whole dictionary. $f(x)$ is the score of all correct n-gram terms, while $f(x^w)$ denotes the score of all negative samples.

B. Recurrent neural network

The recurrent neural network can also be utilized in generating neural language mode[9]. As we can see from the figure, the advantage of this model is that the context of each word is considered. Previous models only use the terms in certain size of window to predict the next term.

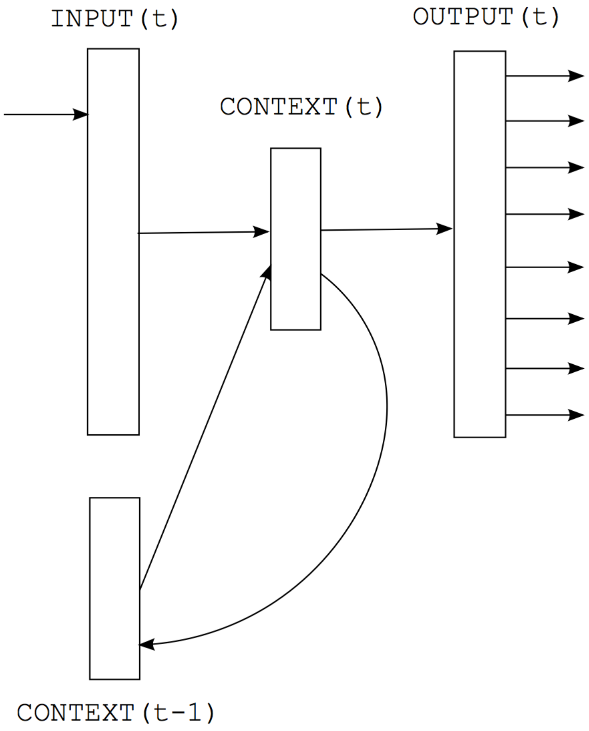

Fig. 4. Parsing natural language sentences

V. Conclusion

The idea of deep learning is to utilize deep structure of neural network to learn more accurate features of the input information. The distributed representation of terms in the neural language model, is a good example. With these features, the task of classification and regression can be more accurate.

In terms of learning algorithm, some structures such as DBN and autoencoder train the model layer by layer. Through this method, the original weights of NN can be more reasonable and the whole model is more likely to approach the global optimum. Another kind of algorithms utilize repetitive neural network structure to capture the semantic information of the language. Since the basic unit of NN is the same, the process of optimizing the parameter is more easy than original multiply layer neural network.

In general, the model of deep learning has been widely in the field of NLP and achieved good performance in many NLP issues.

References

[1] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cognitive modeling*, 1988.

[2] G. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[3] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," *Advances in neural information processing systems*, vol. 19, p. 153, 2007.

[4] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain, "Neural probabilistic language models," in *Innovations in Machine Learning*. Springer, 2006, pp. 137–186.

[5] R. Socher, C. C. Lin, C. Manning, and A. Y. Ng, "Parsing natural scenes and natural language with recursive neural networks," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 129–136.

[6] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011, pp. 151–161.

[7] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.

[8] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[9] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model." in *INTERSPEECH*, 2010, pp. 1045–1048.