# End-to-end Convolutional Neural Networks for Intent Detection

**Sevinj Yolchuyeva**[1], **Géza Németh, Bálint Gyires-Tóth**
{syolchuyeva, nemeth, toth.b}@tmit.bme.hu
Department of Telecommunications and Media Informatics
Budapest University of Technology and Economics
Magyar Tudósok krt. 2., Budapest, 1111 Hungary

## Abstract

Convolutional Neural Networks (CNNs) have been applied to various machine learning tasks, such as computer vision, speech technologies and machine translation. One of the main advantages of CNNs is the representation learning capability from high-dimensional data. End-to-end CNN models have been massively explored in computer vision domain, and this approach has also been attempted in other domains as well. In this paper, a novel end-to-end CNN architecture with residual connections is presented for intent detection, which is one of the main goals for building a spoken language understanding (SLU) system. Experiments on two datasets (ATIS and Snips) were carried out. The results demonstrate that the proposed model outperforms previous solutions.

**Keywords**: Spoken Language Understanding (SLU), intent detection, Convolutional Neural Networks, residual connections, deep learning, neural networks.

## 1 Introduction

Spoken dialogue systems are agents that are intended to help users to access information efficiently by speech interactions. Creating such a system has been a challenge for both academic investigations and commercial applications for decades. Spoken language understanding (SLU) is one of the essential components in spoken dialogue systems [1]. SLU is aiming to form a semantic frame that captures the semantics of user utterances or queries. The three major tasks in an SLU system are domain classification, intent detection, and slot filling. Intent detection can be treated as a semantic utterance classification problem [5,10]. Intent detection solutions classify speakers' intent and extract semantic concepts as constraints for natural language. Take a weath-

[1] Corresponding author:
syolchuyeva@tmit.bme.hu

er-related utterance as an example, "*Weather next year in Canada*", as shown in Figure 1. There are different slot labels for each word in the utterance and a specific intent for the whole utterance.


**Figure 1**. Snips corpus sample with the utterance and slot annotation.

Slot filling can be formulated as a sequence labelling task [2,3]. Joint training of intent detection and slot filling models has been investigated [5,6]. The slot-gated SLU model, which incorporates attention and gating mechanism into the language understanding (LU) network was proposed by [5]. Moreover, conditional random field (CRF), introduced in [4], provides a framework for building probabilistic models to segment and label sequences and applies on different natural language processing (NLP) tasks (e.g., part of speech tagging, sentence classification, grapheme-to-phoneme conversion). Jointly modelling intent labels and slot sequences, thus, exploiting their dependencies by the combination of convolutional neural networks (CNN) and the triangular CRF model (TriCRF) can be beneficial [6]. With this approach, the intent error on Airline Travel Information System (ATIS) dataset was 5.91% for intent detection, and the F1-score was 95.42% for slot filling. Bidirectional Gated Recurrent Units (GRUs) could also be used to learn sequence representations shared by intent detection and slot filling tasks [9]. This approach employs max-pooling layer for capturing global features of a sentence for intent detection.

Recently, encoder-decoder neural networks (also referred to as sequence-to-sequence, or seq2seq models) have achieved remarkable success in various tasks, such as speech recognition, text-to-speech synthesis and machine translation [14,15,16]. In this structure, the encoder computes a latent representation of each input sequence, and the decoder generates an output sequence based on the latent representation. This type of network has been extended with attention mechanism [12,13] and applied to grapheme-to-phoneme conversion (G2P) [17]. Applying such models, intent detection and slot filling were also investigated [21,24]. The combination of the attention-based encoder-decoder architecture and alignment-based methods for joint intent detection and slot filling achieved 5.60% intent error on ATIS dataset [21].

In this work, we investigated CNN based residual networks for intent detection. Experiments were carried out on the ATIS and Snips dataset, which is widely used in SLU research. We show the effectiveness of the proposed models in different experimental settings. Using pre-trained Word2vec [25] and Glove [27] embedding also help to get comparable results. The remaining part of the paper is organized as follows. In Section 2, we introduce word embedding methods. In Section 3, the used datasets are described. In Section 4 the proposed method is introduced. Section 5 discusses the experiment setup and results on ATIS and Snips datasets. Section 6 concludes the work.

## 2 Word Embedding

Word embeddings are used for representing words as vectors. Word embedding models generated with tool, such as Word2vec (skip-gram and continuous bag-of-words (CBOW)) [25], and GloVe [27], generate word vectors based on the distributional hypothesis, which assumes that the meaning of each word can be represented by the context of the word. Continuous Bag-of-Words (CBOW) and Continuous Skip-gram models are still powerful techniques for learning word vectors [25]. CBOW computes the conditional probability of a target word given the context words surrounding it across a window with a predefined size. Skip-gram predicts the surrounding context words based on the central target word [25, 28]. The context words are assumed to be located symmetrically to the target words within a distance equal to the window size in both directions. GloVe word embedding is a global log-bilinear regression model and is based on co-occurrence and factorization of the matrix in order to produce the word vectors.

Pre-trained word embeddings have proven to be highly useful in neural network models for NLP, e.g., in machine translation and text classification [11,19, 26]. In this work, we used 300-dimension Word2vec[2] embeddings trained on Google News and 100-dimension GloVe[3] word embeddings trained on Wikipedia.

## 3 Dataset

We used the Airline Travel Information System (ATIS)[4] dataset, which has been frequently chosen by various researchers [5,11,38]. The dataset contains audio recordings from people making flight reservations. The training set contains 4,478 utterances, the test set contains 893 utterances, and 500 utterances are used for validation (referred to as development set in the paper). Besides ATIS, Natural Language Understanding[5] benchmark dataset was also used. This balanced dataset is collected from the Snips personal voice assistant; the number of samples for each intent is approximately the same. The training set contains 13,084 utterances, the test set contains 700 utterances, and 700 utterances as validation data (development set). All words are labelled with a semantic label in a BIO format, which 'B' means to begin, 'I' means inside, 'O' is outside. Words which don't have semantic labels are tagged with 'O'. For example, 'Weather next year in Canada' contains five words, and these words are labelled according to Figure 1. The sequence 'next year' is labelled as B-timeRange and I-timeRange and 'Canada' is tagged as B-country. The rest of the words in the utterance are labelled as 'O'.

[2] https://code.google.com/archive/p/word2vec/, Accessed: 14th November, 2018

[3] https://github.com/stanfordnlp/GloVe, Accessed: 14th November, 2018

[4]https://github.com/MiuLab/SlotGated-SLU/tree/master/data/atis, Accessed: 14th November, 2018

[5]https://github.com/snipsco/nlu-benchmark/tree/master/2017-06-custom-intent-engines, Accessed: 14th November, 2018

There are 120 slot labels and 21 intent types in ATIS; there are 72 slot labels and 7 intent types in Snips dataset. Vocabulary size of these datasets is 722 and 11,241 in ATIS and Snips, respectively. Compared to single-domain ATIS dataset, Snips is more complicated, mainly due to the intent diversity and large vocabulary. The intent diversity of ATIS and Snips dataset are shown in Table 1 and Table 2.

**Table 1**. The number of intents in the training data of Snips.

| Type of intent | Number |
|---|---|
| PlayMusic | 1914 |
| GetWeather | 1896 |
| BookRestaurant | 1881 |
| RateBook | 1876 |
| SearchScreeningEvent | 1851 |
| SearchCreativeWork | 1847 |
| AddToPlaylist | 1818 |

**Table 2**. The number of intents in the training data of ATIS.

| Type | Number |
|---|---|
| atis_flight | 3309 |
| atis_airfare | 385 |
| atis_ground_service | 230 |
| atis_airline | 139 |
| atis_abbreviation | 130 |
| atis_aircraft | 70 |
| atis_flight_time | 45 |
| atis_quantity | 41 |
| atis_flight#atis_airfare | 19 |
| atis_city | 18 |
| atis_distance | 17 |
| atis_airport | 17 |
| atis_ground_fare | 15 |
| atis_capacity | 15 |
| atis_flight_no | 12 |
| atis_meal | 6 |
| atis_restriction | 5 |
| atis_airline#atis_flight_no | 2 |
| atis_aircraft#atis_flight#atis_flight_no | 1 |
| atis_cheapest | 1 |
| atis_ground_service#atis_ground_fare | 1 |

The intents in Snips are diverse and balanced. The maximal number of utterances are in the PlayMusic domain, the least number of utterances are in AddToPlaylist. The intent types in ATIS are unbalanced. For example, the intent atis_flight equals about 73.8% of training data, while there are intents with one utterance only. Intents with

small number of occurrences were excluded from training and evaluation (e.g. atis_day_name, atis_airfare#atis_flight, atis_flight#atis_airline, atis_flight_no#atis_airline).

## 4 Proposed Work

This section first explains CNN and then introduces the proposed end-to-end CNN approach with residual connections for intent classification.

### 4.1 Convolutional Neural Networks for Intent detection

The architecture of an ordinary CNN is composed of different types of layers (such as the convolutional layers, pooling layers, fully connecting layers, etc.) [34] where each layer realizes a specific function. The convolutional layers are for representation learning, while the fully connected layers on the top of the network are for modelling a classification or regression problem. Convolutional neural networks are jointly performing representation learning and modelling, which makes these models superior to other methods in many cases. Weight sharing in the convolutional layers is essential for the model to become spatially tolerant: similar representations are learned in different regions of the input, and the total number of parameters can also be reduced drastically.

Increasing the number of layers in deep CNNs does not implicitly results in better accuracy, and some issues, such as vanishing gradient and degradation problems may arise as well. Introducing residual connection can improve the performance significantly [29]. These kinds of connections allow the information and gradients to flow more into the deeper layers, increases the convergence speed and decreases the vanishing gradient problem.

Convolutional neural networks were already successfully applied to various NLP tasks [33, 38, 39]. These results suggest investigating CNN based sequence models for intent classification. We expected that convolutional neural networks enhances the performance of intent detection task.

### 4.2 Model architecture

All utterances and their slots sequences are splatted as Input 1 and Input 2. We use <BOS> and <EOS> tokens as beginning-of-utterances and end-of-utterances tokens in Input 1 and beginning-of-slots and end-of-slots tokens in Input 2, as shown in Table 3.

**Table 3**. The structure of input and output.

| **Input 1** | **Input 2** | **Output** |
|---|---|---|
| <BOS> weather next year in Canada <EOS> | <BOS> O B-timeRange I-timeRange O B-country <EOS> | GetWeather |

Regarding Input 1, an embedding layer with pretrained word vectors, such as Word2vec or GloVe, was applied. Regarding Input 2, the slots were tokenized, and embedding was applied, which is intended to map positive integer values in an array to float values. The proposed model was applied on both inputs separately, and then the output of these models (referred to as Model 1 and Model 2) are combined (see Figure 2). Model 1 and Model 2 contains convolutional layers with residual connections. After embedding, a 1D convolutional layer with 16 filters is applied, which isfollowed by a stack of residual blocks. Through hyperoptimization, the best result was achieved by 3 residual blocks, and the number of filters in each residual block was 32, 64, 128. Each residual block consists of 2 convolutional layers followed by batch normalization layer [32] and ReLU activation. The filter size of all convolutional layers is 5. These blocks are followed by one more batch normalization layer and a ReLU activation. The architecture ends with a fully connected layer coupled with softmax activation function. The model architecture is shown in Figure 3.

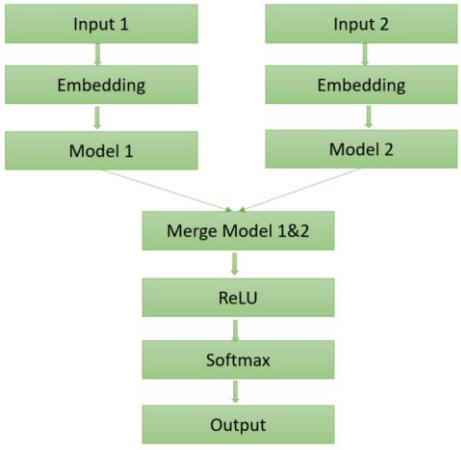


**Figure 2.** Proposed model architecture.

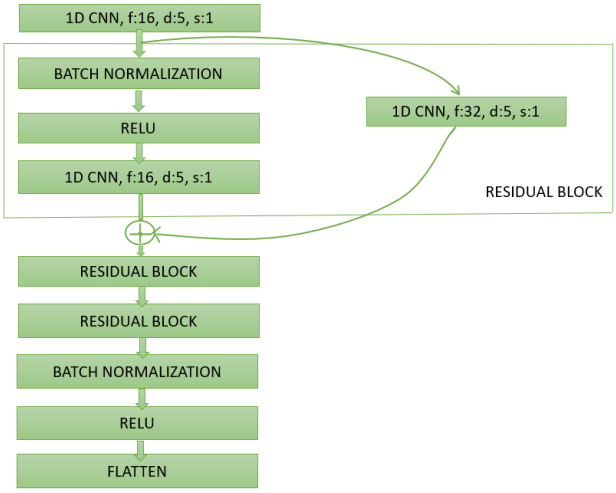


**Figure 3.** End-to-end CNN structure for intent detection task. f, d, and s are the number of the filters, length of the filters and stride, respectively.

In general, using CNN for intent detection is similar to a standard classification problem, ATIS dataset is under the flight reservation domain with 17 intents, Snips with 7 intents.

## 5 Evaluation and Results

We used NVidia Titan Xp (12 GB) and NVidia Titan X (12 GB) GPU cards hosted in two i7 workstations with 32GB RAM. For training and inferrrence the Keras deep learning framework with Theano [30] backend was used.
We trained the models both with Word2vec and Glove vector representations.
After training the models predictions were performed on the test dataset, and the results were evaluated with confusion matrices and accuracy.
The results of the experiments are shown in Table 4. We compared our solution with state-of-the-art intent detection models, such as Slot-Gated (Intent Attention) [5], Attention-based BiRNN [22], and Recursive Neural Network [36] models. Better results by using different approaches are also published, but in those cases different variations or parts of the ATIS dataset were used [17, 23]. In Table 4, the first column shows the applied architecture models; the second and third columns show overall accuracy for each model on ATIS and Snips dataset. For Snips, we are able to get 100% accuracy using pretrained Glove vectors on the test set.
The confusion matrix is an effective method to visualize and to examine the performance of binary and multi-class classifiers [34]. Generally, the confusion matrix shows the detailed number of correctly classified and misclassified intents. The diagonal represents the correct predictions. Each entry outside the diagonal shows how

many tokens from each intent (y-axis) were incorrectly assigned to other intents (x-axis).

Figure 4 shows the confusion matrix of the proposed model using GloVe pretrained vectors on ATIS dataset. The intent atis_flight is 73.8 % of training dataset and it is the most part of test dataset too. 629 utterances were classified correctly out of 630. The number of utterances in atis_restriction is zero in test data. Figure 5 and Figure 6 show the confusion matrix of the proposed model using GloVe and Word2vec pretrained vectors on Snips dataset, respectively. In Figure 5, the proposed model correctly classified 629 utterances out of 661 for the atis_flight and 46 out of 51 for the atis_airfare intent. The accuracy of these intents is 95.2 and 90.2%, respectively. More than half of the test utterances of atis_distance and atis_meal were misclassified. In Figure 5, all intents are correctly classified for Snips test dataset by using GloVe pretrained vectors.

**Table 4.** The accuracy (%) of previous works and the proposed models on ATIS and Snips test datasets

| **Model** | **ATIS** | **Snips** |
|---|---|---|
| Slot-Gated (Intent Attention) [ 5] | 94.1 | 96.8 |
| Attention-based BiRNN [22] | 92.6 | - |
| Recursive Neural Network [36] | 95.40 | - |
| **Word2vec + CNN with residual connections (proposed work)** | 95.46 | 99.7 |
| **Glove + CNN with res. Con. (proposed work)** | 94.40 | 100 |

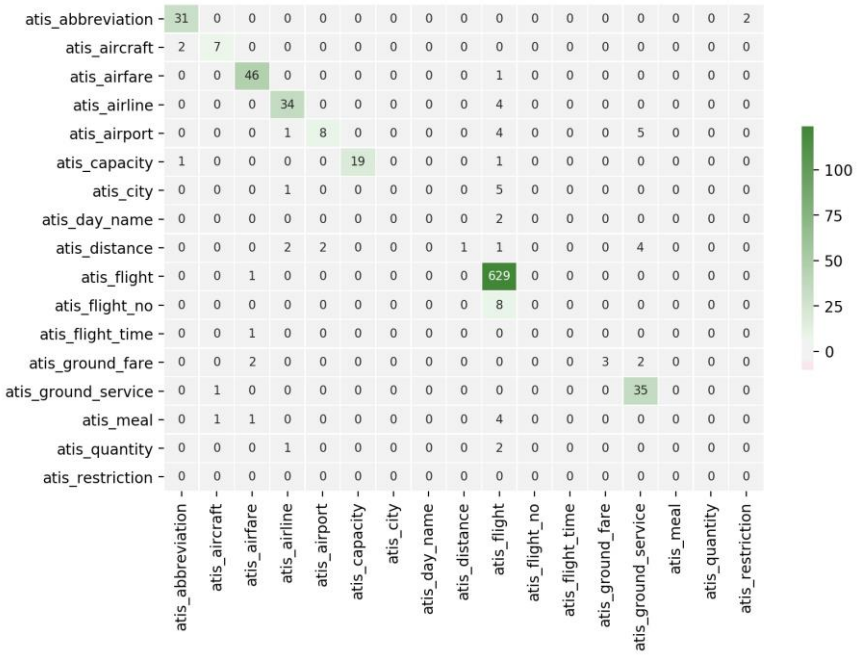


**Figure 4**. Confusion matrix of ATIS test dataset by using GloVe pretrained vectors.

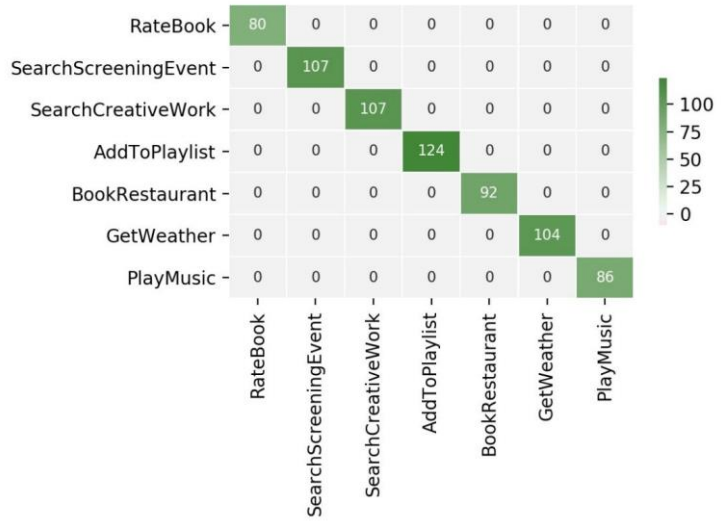


**Figure 5**. Confusion matrix of Snips test dataset by using GloVe pretrained vectors.

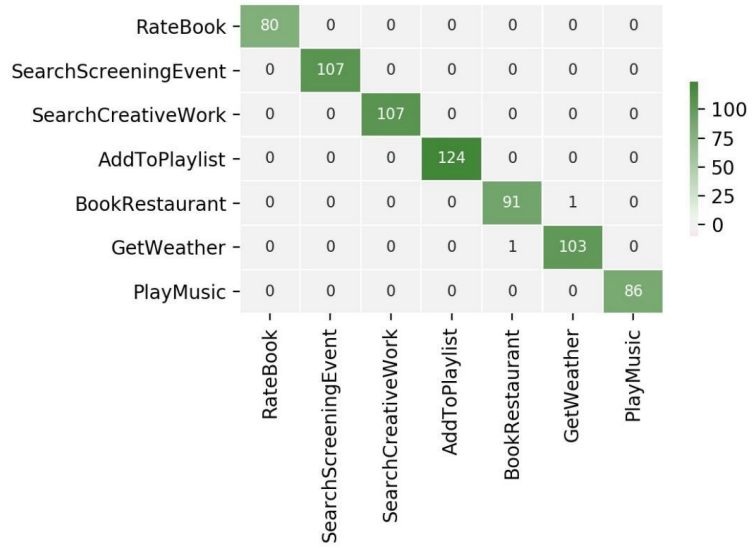


**Figure 6.** Confusion matrix of Snips test dataset by using Word2vec pretrained vectors.

## 6 Conclusions and Future Work

In this paper, an end-to-end CNN model with residual connections for intent detection were proposed. 300-dimensional Word2vec embeddings pretrained on Google News and 100-dimension GloVe word embeddings pretrained on Wikipedia were used for word representations. The results were evaluated with the help of confusion matrix and accuracy. The proposed method outperformed previous solutions in terms of accuracy.

## Acknowledgements

The research presented in this paper has been supported by the BME-Artificial Intelligence FIKP grant of Ministry of Human Resources (BME FIKP-MI/SC), by Doctoral Research Scholarship of Ministry of Human Resources (ÚNKP-18-4-BME-394) in the scope of New National Excellence Program, by János Bolyai Research Scholarship of the Hungarian Academy of Sciences, by the VUK project (AAL 2014-183), and the DANSPLAT project (Eureka 9944). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

## References

[1] Serdyuk, D., Wang, Y., Fuegen, C., Kumar, A., Liu, B., and Bengio, Y. (2018). Towards End-to-end Spoken Language Understanding. 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 5754-5758.

[2] Wang, Y., Shen, Y., and Jin, H. (2018). A Bi-Model Based RNN Semantic Frame Parsing Model for Intent Detection and Slot Filling. Proceedings of Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), 309-314.

[3] Tur, G., and Mori, R.D. (2011). Spoken language understanding: Systems for extracting semantic information from speech. John Wiley & Sons.

[4] Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Machine Learning-International Workshop Then Conference, 282–289.

[5] Goo, C., Gao, G., Hsu, Y., Huo, C., Chen, T., Hsu, K., and Chen, Y. (2018). Slot-Gated Modeling for Joint Slot Filling and Intent Prediction. Proceedings of Annual Conference North American Chapter of the Association for Computational Linguistics, 753-757.

[6] Xu, P., and Sarikaya, R. (2013). Convolutional neural network based triangular CRF for joint intent detection and slot filling. In IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), 78–83.

[7] Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., and Zweig, G. (2015). Using recurrent neural networks for slot filling in spoken language understanding. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23(3):530–539.

[8] Meng, L., and Huang, M. (2018). Dialogue Intent Classification with Long Short-Term Memory Networks. Natural Language Processing and Chinese Computing. Ed. by X. Huang et al. Cham: Springer International Publishing, 42–50.
[9] Zhang, X., and Wang, H. (2016) A Joint Model of Intent Determination and Slot Filling for Spoken Language Understanding. International Joint Conferences on Artificial Intelligence, 2993–2999.
[10] Liu, B., and Lane, I. (2016). Joint Online Spoken Language Understanding and Language Modeling with Recurrent Neural Networks. 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL), 22-30.
[11] Kim, J., Tür, G., Çelikyilmaz, A., Cao, B., and Wang, Y. (2016). Intent detection using semantically enriched word embeddings. 2016 IEEE Spoken Language Technology Workshop (SLT), 414-419.
[12] Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
[13] Luong, M.T., Pham, H., and Manning, C.D. (2015). Effective approaches to attention-based neural machine translation. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 1412–1421.
[14] Kalchbrenner, N., and Phil, B. (2013). Recurrent Continuous Translation Models. Proceedings of Conference on Empirical Methods in Natural Language Processing, 1700–1709.
[15] Cho, K., Bart, M., Caglar, G., Dzmitry, B., Fethi, B., Holger, S., and Yoshua, B. (2014). Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation. Proceedings of Conference on Empirical Methods in Natural Language Processing, 1724-1734.
[16] Lu, L., Zhang, X., and Renals, S (2016). On Training the Recurrent Neural Network Encoder-Decoder for Large Vocabulary End-to-End Speech Recognition. IEEE International Conference on Acoustics, Speech and Signal Processing, 5060–5064.
[17] Toshniwal, S., and Livescu, K. (2016). Jointly learning to align and convert graphemes to phonemes with neural attention models. IEEE Spoken Language Technology Workshop (SLT), 76-82.
[18] Hashemi, H.B. (2016). Query Intent Detection using Convolutional Neural Networks. In International Conference on Web Search and Data Mining, Workshop on Query Understanding.
[19] Wang, P., Qian, Y., Frank K. Soong, He, L., and Zhao, H. (2015). Word embedding for recurrent neural network based TTS synthesis. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 4879–4883.
[20] Ravuri, S., and Stolcke, A. (2015). A Comparative Study of Neural Network Models for Lexical Intent Classification. 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), 368–374.
[21] Liu, B., and Lane, I. (2016). Attention-based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling. Proceedings of the 17th Annual Meeting of the International Speech Communication Association, 685-689.
[22] Hakkani-Tur, D., Tur, G., Celikyilmaz, A., Chen, Y.N., Gao, J., Deng, L., and Wang, Y.Y. (2016). Multi-Domain Joint Semantic Frame Parsing using Bi-directional RNN-LSTM. In Proceedings of the 17th Annual Meeting of the International Speech Communication Association, 715-719.
[23] Zhang, X., and Wang, H. (2016). A Joint Model of Intent Determination and Slot Filling for Spoken Language Understanding. International Joint Conferences on Artificial Intelligence (IJCAI), 2993–2999.
[24] Schumann, R., and Angkititrakul, P. (2018). Incorporating ASR Errors with Attention-based, Jointly Trained RNN for Intent Detection and Slot Filling. 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),685-689.

[25] Mikolov, T., Corrado, G., Chen, K., and Dean J., (2013). Efficient Estimation of Word Representations in Vector Space. Proceedings of the International Conference on Learning Representations (ICLR), 1–12.
[26] Qi, Y., Sachan, D.S., Felix, M., Padmanabhan, S.J., and Neubig, G. (2018). When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation?. Proceedings of Annual Conference of the North American Chapter of the Association for Computational Linguistics 2018 (NAACL-HLT), 529–535.
[27] Pennington, J., Socher, R., and Manning, C.D. (2014). GloVe: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 1532–1543.
[28] Young, T., Devamanyu. H., Soujanya, P., and Cambria, E. (2017). Recent Trends in Deep Learning Based Natural Language Processing. arXiv: preprint arXiv:1708.02709v4.
[29] Kaiming, H., Xiangyu, Z., Shaoqing, R. and Jian, S. (2016). Deep Residual Learning for Image Recognition. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 770-778.
[30] The Theano Development (2016). A Python framework for fast computation of mathematical expressions, arXiv preprint arXiv:1605.02688.
[31] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. (2015). Going Deeper with Convolutions. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1–9.
[32] Ioffe, S., and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning. 448-456.
[33] Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., and Xu, B. (2016). Text Classification Improved by Integrating Bidirectional LSTM with Two-Dimensional Max Pooling. In Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers (COLING 2016), 3485–3495.
[34] Yolchuyeva, S., Németh, G. and Gyires-Tóth, B. (2018) Text normalization with convolutional neural networks. International Journal of Speech Technology, Volume 21, Number 3, 589-600.
[35] Sonmez, C., and Ozgur, A. (2014). A Graph-Based Approach for Contextual Text Normalization. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 313–324.
[36] Guo, D., Tür, G., Yih, W., and Zweig, G. (2014). Joint semantic utterance classification and slot filling with recursive neural networks. 2014 IEEE Spoken Language Technology Workshop (SLT), 554-559.
[37] Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y.N. (2017). Convolutional Sequence to Sequence Learning. arXiv preprint arXiv: 1705.03122.
[38] Gehring, J., Auli, M., Grangier, D., and Dauphin, Y.N. (2016). A Convolutional Encoder Model for Neural Machine Translation. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 123-135.
[39] Xiang, Z., Zhao, J., and Yann, L. (2015). Character-Level Convolutional Networks for Text Classification. Advances in Neural Information Processing Systems 28 (NIPS 2015), 1–9.