Neural Machine Translation

Arul Selvam Periyasamy

July 11, 2017

Rheinische Friedrich-Wilhelms-Universität Bonn
Seminar: Natural Language Processing

Agenda

- Introduction to Machine Translation

Agenda

- Introduction to Machine Translation
- Evaluation Metric

Agenda

- Introduction to Machine Translation
- Evaluation Metric
- Statistical Phrase-Based Translation

Agenda

- Introduction to Machine Translation
- Evaluation Metric
- Statistical Phrase-Based Translation
- Introduction to Deep Learning

Agenda

- Introduction to Machine Translation
- Evaluation Metric
- Statistical Phrase-Based Translation
- Introduction to Deep Learning
- Neural Machine Translation

Machine Translation

In a probabilistic perspective, machine translation can be formulated the problem of finding a target sentence y that maximizes the conditional probability of y from a given source sentence x.

$$arg\, max_y\ p(x|y)$$

Evaluation Metric

Evaluation Metric

- Evaluating the quality of machine translation is a hard task.

Evaluation Metric

- Evaluating the quality of machine translation is a hard task.
- No one best traget sentence possible.

Evaluation Metric

- Evaluating the quality of machine translation is a hard task.
- No one best traget sentence possible.
- Even human translators don't translate to the same target sentence.

Evaluation Metric

- BLEU- Bilingual Evaluation Understudy is the most commonly used error metric.

Evaluation Metric

- BLEU- Bilingual Evaluation Understudy is the most commonly used error metric.
- Depends on modified n-gram precision (or co-occurrence).

Evaluation Metric

- BLEU- Bilingual Evaluation Understudy is the most commonly used error metric.
- Depends on modified n-gram precision (or co-occurrence).
- Needs lots of target sentences for better results.

BLEU

- Candidate 1: It is a guide to action which ensures that the military always obey the commands the party.
- Candidate 2: It is to insure the troops forever hearing the activity guidebook that party direct.
- Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.

BLEU

- Candidate 1: It is a guide to action which ensures that the military always obey the commands the party.
- Candidate 2: It is to insure the troops forever hearing the activity guidebook that party direct.
- Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.
- Reference 2: It is the guiding principle which guarantees the military forces always being under the command of the Party.

BLEU

- Candidate 1: It is a guide to action which ensures that the military always obey the commands the party.
- Candidate 2: It is to insure the troops forever hearing the activity guidebook that party direct.
- Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.
- Reference 2: It is the guiding principle which guarantees the military forces always being under the command of the Party.
- Reference 3: It is the practical guide for the army always to heed directions of the party.

BLEU

- Candidate 1: It is a guide to action which ensures that the military always obey the commands the party.
- Candidate 2: It is to insure the troops forever hearing the activity guidebook that party direct.
- Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.
- Reference 2: It is the guiding principle which guarantees the military forces always being under the command of the Party.
- Reference 3: It is the practical guide for the army always to heed directions of the party.
- **Clearly candidate 1 is better.**

BLEU

To rank candidate 1 better than 2,

- Just count the number of N-gram matches.

BLEU

To rank candidate 1 better than 2,

- Just count the number of N-gram matches.
- The match could be position-independent.

BLEU

To rank candidate 1 better than 2,

- Just count the number of N-gram matches.
- The match could be position-independent.
- Reference could be matched multiple times.

To rank candidate 1 better than 2,

- Just count the number of N-gram matches.
- The match could be position-independent.
- Reference could be matched multiple times.
- No need to be linguistically-motivated.

BLEU

Candidate 1: It is a guide to action which ensures that the military always obey the commands of the party.
Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.
Reference 2: It is the guiding principle which guarantees the military forces always being under the command of the Party.
Reference 3: It is the practical guide for the army always to heed directions of the party.
N-gram Precision : 17

BLEU

Candidate 2: It is to insure the troops forever hearing the activity guidebook that party direct.
Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.
Reference 2: It is the guiding principle which guarantees the military forces always being under the command of the Party.
Reference 3: It is the practical guide for the army always to heed directions of the party.
N-gram Precision : 8

BLEU

Issues with N-gram precision Candidate: the the the the the the the.

Reference 1: The cat is on the mat.

Reference 2: There is a cat on the mat.

N-gram Precision : 7 and BLEU : 1

BLEU

Algorithm	Example
Count the max number of times a word occurs in any single reference	Ref 1: The cat is on the mat. Ref 2: There is a cat on the mat. “the” has max count 2
Clip the total count of each candidate word	Unigram count = 7 Clipped unigram count = 2 Total no. of counts = 7
Modified N-gram Precision equal to Clipped count/ Total no. of candidate word	Modified-ngram precision: Clipped count = 2 Total no. of counts =7 Modified-ngram precision = 2/7

N-grams with different Ns are used but 4 is most common metric.

Phrase Based Machine Translation

Phrase Based Machine Translation

- Follows the paradigm of Statistical Machine Translation.

Phrase Based Machine Translation

- Follows the paradigm of Statistical Machine Translation.
- Works at the level of phrases instead of words.

Phrase Based Machine Translation

- Follows the paradigm of Statistical Machine Translation.
- Works at the level of phrases instead of words.
- Lots of individual components but the core idea is learning statistical patterns in training data.

Phrase Based Machine Translation

Some of the individual components:

- Sentence alignment: Gale and Church Algorithm based on Dynamic programming.

Phrase Based Machine Translation

Some of the individual components:

- Sentence alignment: Gale and Church Algorithm based on Dynamic programming.
- Word alignment: Expectation Maximization.

Phrase Based Machine Translation

Some of the individual components:

- Sentence alignment: Gale and Church Algorithm based on Dynamic programming.
- Word alignment: Expectation Maximization.
- Phrases generation: Heuristic based complex algorithms.

Phrase Based Machine Translation

Some of the individual components:

- Sentence alignment: Gale and Church Algorithm based on Dynamic programming.
- Word alignment: Expectation Maximization.
- Phrases generation: Heuristic based complex algorithms.
- Phrase lookup: Statistical matching.

Phrase Based Machine Translation

Some of the individual components:

- Sentence alignment: Gale and Church Algorithm based on Dynamic programming.
- Word alignment: Expectation Maximization.
- Phrases generation: Heuristic based complex algorithms.
- Phrase lookup: Statistical matching.
- Beam search: For generating target sentence.

Phrase Based Machine Translation

Some of the individual components:

- Sentence alignment: Gale and Church Algorithm based on Dynamic programming.
- Word alignment: Expectation Maximization.
- Phrases generation: Heuristic based complex algorithms.
- Phrase lookup: Statistical matching.
- Beam search: For generating target sentence.
- Beam search is a generic algorithm that is used even in the latest NMT systems.

Phrase Based Machine Translation

Some properties:

- Achieved a BLEU score of 28.0 on WMT'14 English-to-French dataset.

Phrase Based Machine Translation

Some properties:

- Achieved a BLEU score of 28.0 on WMT'14 English-to-French dataset.
- Needs one model for each language pair.

Phrase Based Machine Translation

Some properties:

- Achieved a BLEU score of 28.0 on WMT'14 English-to-French dataset.
- Needs one model for each language pair.
- Google avoided the need for combinatorial number of models by always translating to English as intermediate language.

Phrase Based Machine Translation

Some properties:

- Achieved a BLEU score of 28.0 on WMT'14 English-to-French dataset.
- Needs one model for each language pair.
- Google avoided the need for combinatorial number of models by always translating to English as intermediate language.
- Thus in practice, the accuracy dropped further.

Introduction to Deep Learning

Machine Learning (Supervised)

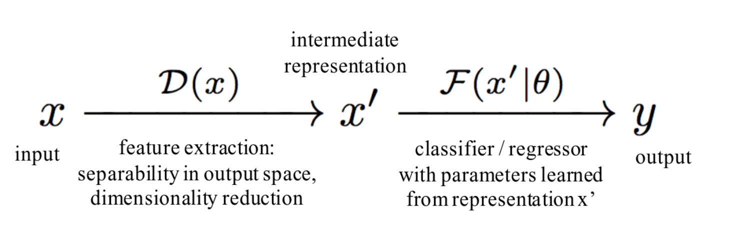


Figure: Traditional Supervised learning

Deep Learning (Supervised)

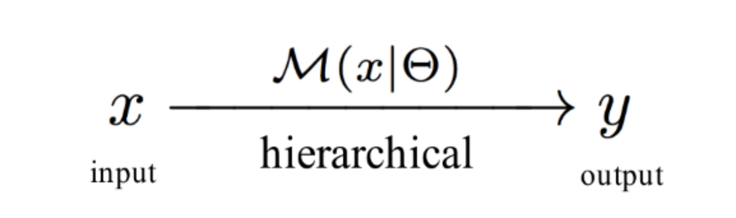


Figure: Deep learning

- Hierarchical representations of features.

Deep Learning (Supervised)

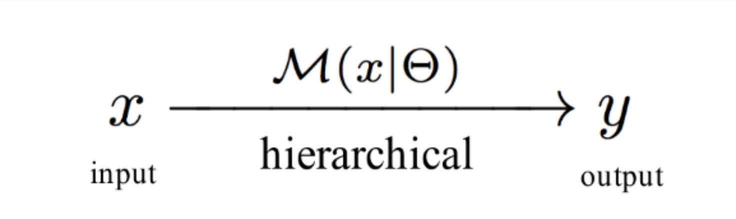


Figure: Deep learning

- Hierarchical representations of features.
- Joint learning of representation.

Deep Learning (Supervised)

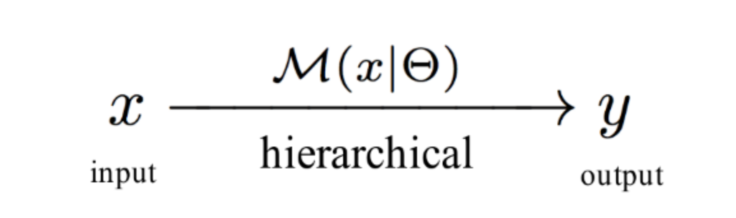


Figure: Deep learning

- Hierarchical representations of features.
- Joint learning of representation.
- Increased levels of abstraction.

Perceptron

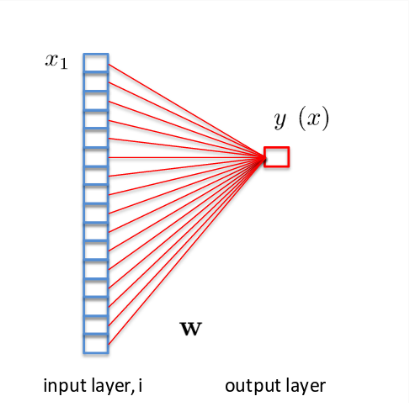


Figure: A Neuron (close to a biological neuron)

$$y(x) = f(W^T x)$$

Logistic Regression

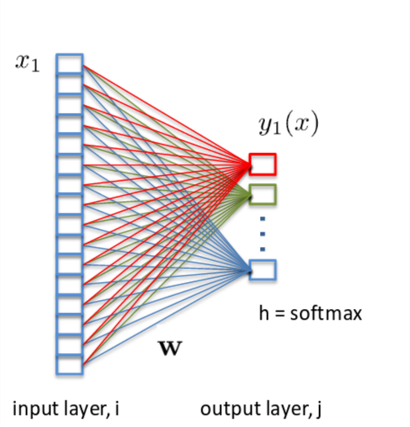


Figure: A perceptron (A collection of perceptrons)

Logistic Regression

Binary classification:

$$P(y=1|x) = h_w(x) = \frac{1}{1+exp(-W^T x)}$$

$$P(y=0|x) = 1 - h_w(x) = 1 - P(y=1|x)$$

Logistic Regression

Cost function:

$$J(w) = -\sum_{i}(y^i log(h_w(x^i)) + (1 - y^i)log(1 - h_w(x^i)))$$

Learning Weights : Gradient Descent

$$\nabla_w J(w) = \frac{\partial J(w)}{\partial w_j} = \sum_{i} x_j^i (h_w(x^i) - y^i)$$

Gradient Descent

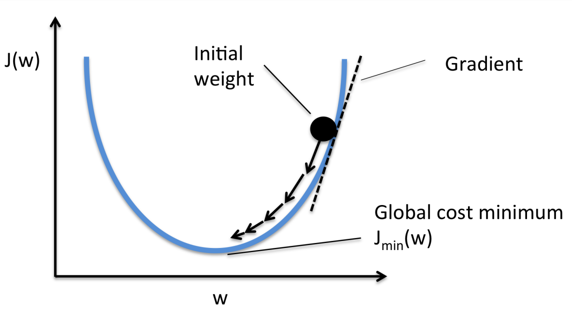


Figure: Update weights in the direction of negative gradient.

Multi Layer Perceptron

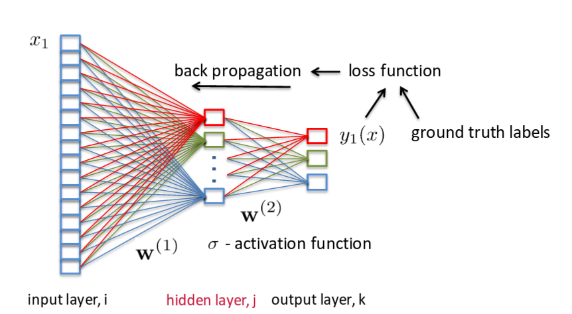


Figure: Multiple layers of perceptron

Learning weights: Same as before but apply chain rule.

$$\frac{\partial x}{\partial y} = \frac{\partial x}{\partial z} * \frac{\partial z}{\partial y}$$

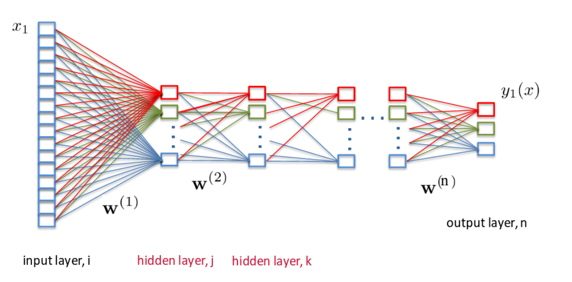


Figure: Deep Neural Networks

- Simply adding layers won't work.

Deep Neural Networks

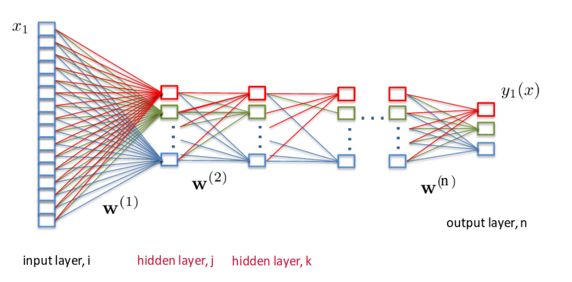


Figure: Deep Neural Networks

- Simply adding layers won't work.
- Too many parameters to train.

Deep Neural Networks

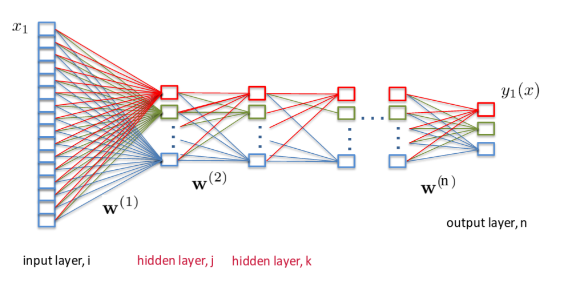


Figure: Deep Neural Networks

- Simply adding layers won't work.
- Too many parameters to train.
- Need smart architectures to capture additional priors.

Deep Neural Networks

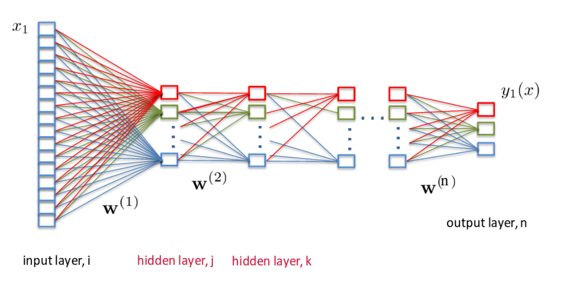


Figure: Deep Neural Networks

- Simply adding layers won't work.
- Too many parameters to train.
- Need smart architectures to capture additional priors.
- Two most commonly used architectures are CNNs and RNNs.

Convolutional Neural Networks

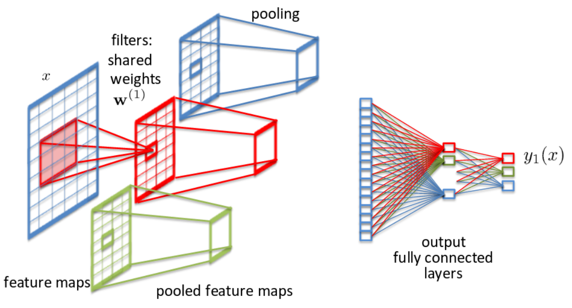


Figure: Convolutional Neural Networks

- Each layers learns a set of convolution kernels.

Convolutional Neural Networks

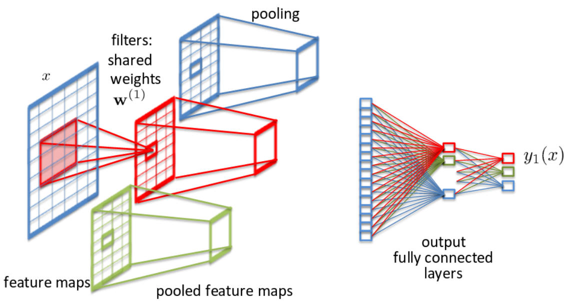


Figure: Convolutional Neural Networks

- Each layers learns a set of convolution kernels.
- Captures a very important prior –smoothness prior– known to computer vision community for a very long time.

Convolutional Neural Networks

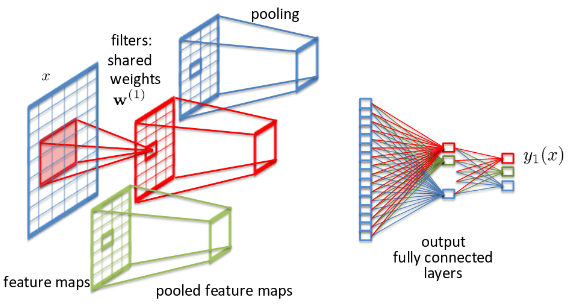


Figure: Convolutional Neural Networks

- Each layers learns a set of convolution kernels.
- Captures a very important prior –smoothness prior– known to computer vision community for a very long time.
- Much less number of parameters.

Recurrent Neural Networks

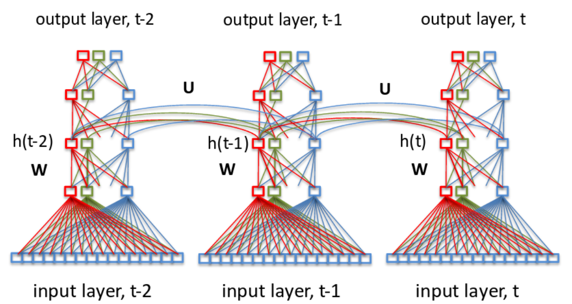


Figure: Recurrent Neural Networks

- Used for predicting sequential data

Recurrent Neural Networks

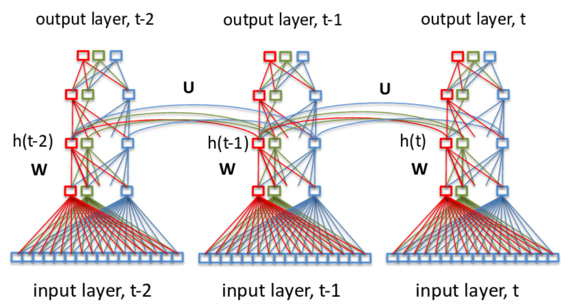


Figure: Recurrent Neural Networks

- Used for predicting sequential data
- Captures dependences across time frames.

Recurrent Neural Networks

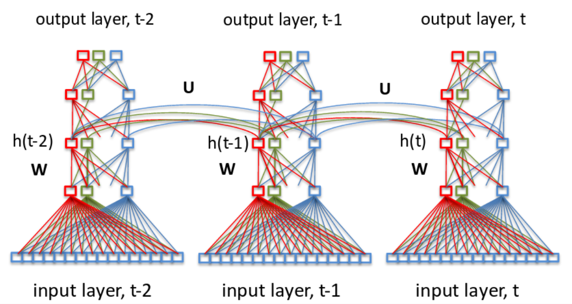


Figure: Recurrent Neural Networks

- Used for predicting sequential data
- Captures dependences across time frames.
- Usually harder to train (Vanishing Gradients).

LSTM

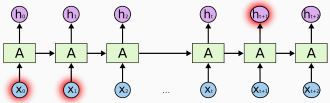


Figure: Recurrent Neural Networks

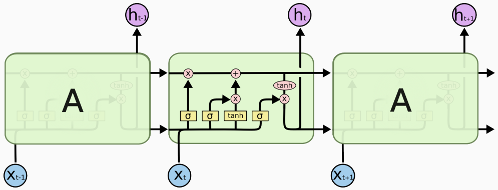
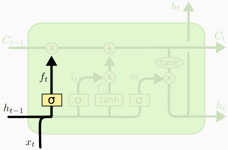


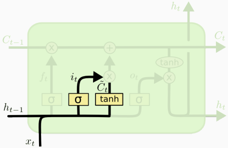
Figure: Long Short Term Memory

LSTM



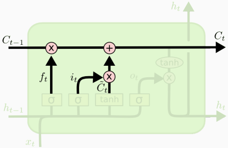
$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

Figure: LSTM Forget gate



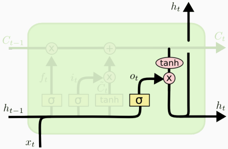
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

Figure: LSTM new content



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Figure: LSTM Add gate



$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Figure: LSTM Output Gate

Neural Machine Translation

Neural Machine Translation

$$arg\, max_y \; p(y|x)$$

In Neural Machine Translation, a parameterized model (a neural network) is trained to maximize the conditional probability of the sentence pairs given parallel training corpus.

NMT - A historic perspective

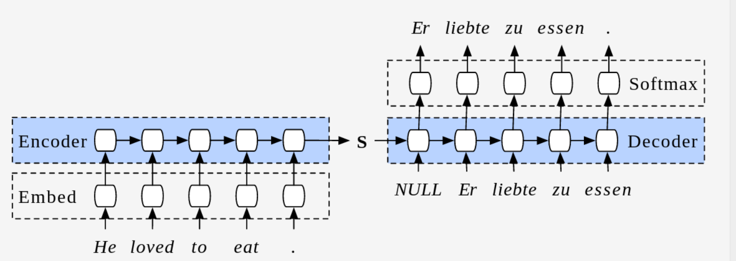


Figure: Encoder-Decoder model for Machine Translation

- Fixed size encodings.

NMT - A historic perspective

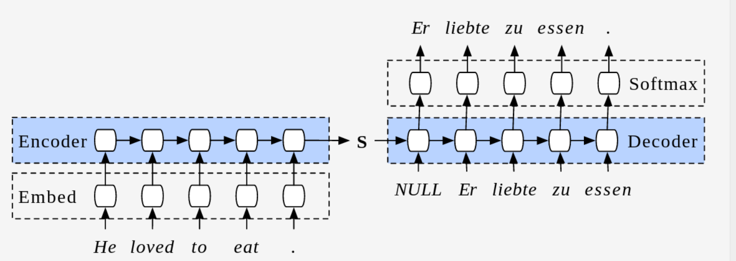


Figure: Encoder-Decoder model for Machine Translation

- Fixed size encodings.
- Each language typically required an Encoder and Decoder.

Jointly learning to Align and translate

Jointly learning to Align and translate

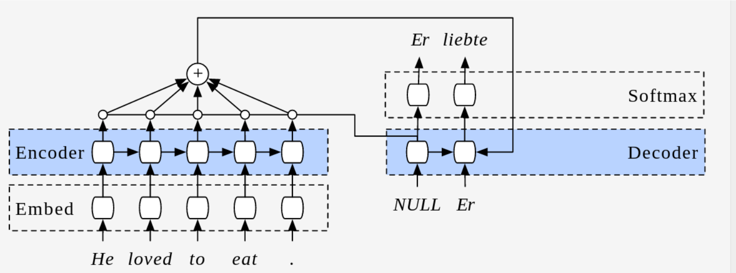


Figure: Encoder-Decoder model with context

Jointly learning to Align and translate

For a input sentence, $X = (x_1, \cdots, x_{T_x})$. The NMT[1] system consists of,

- Encoder and Decoder are multi-layer recurrent neural networks (RNNs).

[1] NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE. Dzmitry Bahdanau, KyungHyun Cho, Yoshua Bengio, ICLR, 2015.

Jointly learning to Align and translate

For a input sentence, $X = (x_1, \cdots, x_{T_x})$. The NMT[1] system consists of,

- Encoder and Decoder are multi-layer recurrent neural networks (RNNs).
- Encoder RNN, at each input step t, generates hidden state, $h_t = f(x_t, h_{t-1})$.

[1] **NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE. Dzmitry Bahdanau, KyungHyun Cho, Yoshua Bengio, ICLR, 2015.**

Jointly learning to Align and translate

For a input sentence, $X = (x_1, \cdots, x_{T_x})$. The NMT[1] system consists of,

- Encoder and Decoder are multi-layer recurrent neural networks (RNNs).
- Encoder RNN, at each input step t, generates hidden state, $h_t = f(x_t, h_{t-1})$.
- Context vector encodes the input sequence as, $c = q(\{h_1, \cdots, h_{T_x}\})$.

[1] NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE. Dzmitry Bahdanau, KyungHyun Cho, Yoshua Bengio, ICLR, 2015.

Jointly learning to Align and translate

- The decoder is trained to predict the next work y_t given the context vector c and all previously predicted words $\{y_1, \cdots, y_{t-1}\}$

$$p(y) = \prod_{t=1}^{T} p(y_t | \{y_1, \cdots, y_{t-1}\}, c)$$

- With RNN, each conditional probability is modeled as,

$$p(y_t | \{y_1, \cdots, y_{t-1}\}, c) = g(y_{t-1}, s_t, c)$$

where s_t is the hidden state of the RNN.

Context Vector

The context vector for a input sentence i, is computed as a weighted sum of hidden states of the encoder (also known as **annotations**)

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{T_x} exp(e_{ik})}$$

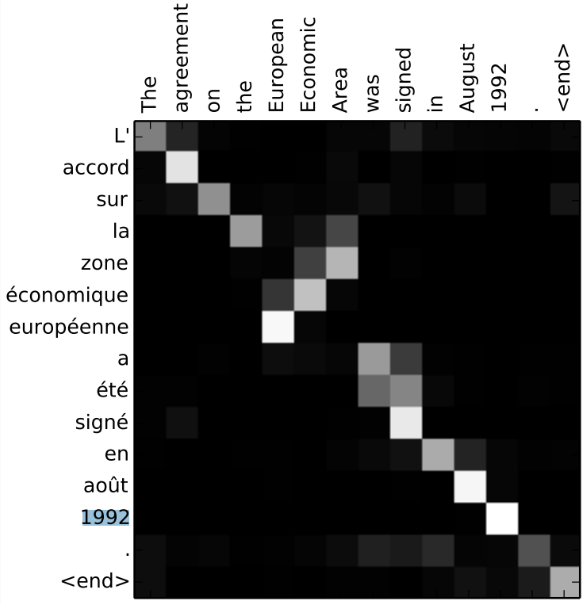
where,

$$e_{ij} = a(s_{i-1}, h_j)$$

is the alignment model that scores how well the inputs around the j and the output at the position i match.

A feedforward neural network is used as the alignment model and is **jointly trained** with all the NMT system as a whole.

Visualization of the context



Bi-directional Encoder

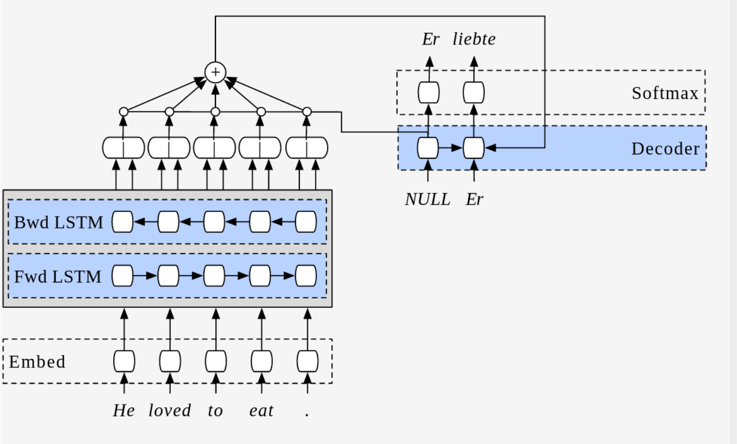


Figure: Bi-directional Encoder

- Recurrent connection in both directions.

Bi-directional Encoder

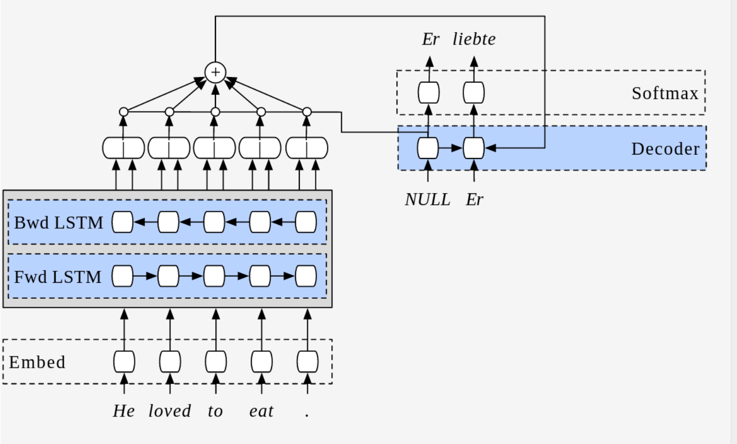


Figure: Bi-directional Encoder

- Recurrent connection in both directions.
- Two independent states, updated independently.

Error Function and training details

- Standard maximum-likelyhood:

$$\mathbb{O}_{ML}(\Theta) = \sum_{i=1}^{N} logP_{\Theta}(Y^{*(i)}|X^{(i)})$$

Error Function and training details

- Standard maximum-likelyhood:

$$\mathbb{O}_{ML}(\Theta) = \sum_{i=1}^{N} log P_{\Theta}(Y^{*(i)} | X^{(i)})$$

- Optimizer-SGD.

Error Function and training details

- Standard maximum-likelyhood:

$$\mathbb{O}_{ML}(\Theta) = \sum_{i=1}^{N} logP_{\Theta}(Y^{*(i)}|X^{(i)})$$

- Optimizer-SGD.
- Minibatch of 80 sentences.

Error Function and training details

- Standard maximum-likelyhood:

$$\mathbb{O}_{ML}(\Theta) = \sum_{i=1}^{N} logP_{\Theta}(Y^{*(i)}|X^{(i)})$$

- Optimizer-SGD.
- Minibatch of 80 sentences.
- BLEU comparable to PBMT.

Seq2Seq Learning

Seq2Seq Learning

Machine Translation is can treated as a special case of a more generic sequence to sequence modeling.

1. Idea is simple: Throw more power at the network.

$$p(y_1, \cdots, y_T | x_1, \cdots, x_T) = \prod_{t=1}^{T} p(y_t | v, y_1, \cdots, y_{t-1})$$

Seq2Seq Learning

Machine Translation is can treated as a special case of a more generic sequence to sequence modeling.

1. Idea is simple: Throw more power at the network.
2. Deep LSTM layers.

$$p(y_1, \cdots, y_T | x_1, \cdots, x_T) = \prod_{t=1}^{T} p(y_t | v, y_1, \cdots, y_{t-1})$$

Seq2Seq Learning

Machine Translation is can treated as a special case of a more generic sequence to sequence modeling.

1. Idea is simple: Throw more power at the network.
2. Deep LSTM layers.
3. No special handling for Machine translation.

$$p(y_1, \cdots, y_T | x_1, \cdots, x_T) = \prod_{t=1}^{T} p(y_t | v, y_1, \cdots, y_{t-1})$$

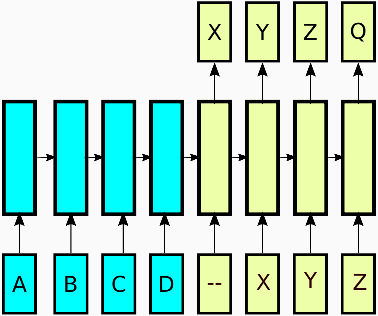
Seq2Seq Learning

Machine Translation is can treated as a special case of a more generic sequence to sequence modeling.

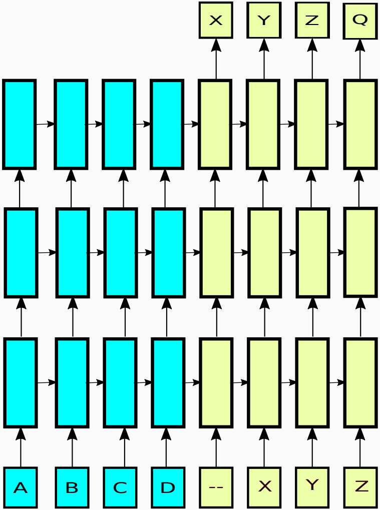
1. Idea is simple: Throw more power at the network.
2. Deep LSTM layers.
3. No special handling for Machine translation.
4. Trained with SGD.

$$p(y_1, \cdots, y_T | x_1, \cdots, x_T) = \prod_{t=1}^{T} p(y_t | v, y_1, \cdots, y_{t-1})$$

Seq2Seq Learning



Seq2Seq Learning



Seq2Seq Learning

- Trained in WMT English to French dataset with 12M sentences consisting of 348M French words and 304M English words.

Seq2Seq Learning

- Trained in WMT English to French dataset with 12M sentences consisting of 348M French words and 304M English words.
- Used 160,000 of the most frequent words for the source language and 80,000 of the most frequent words for the target language

Seq2Seq Learning

- Trained in WMT English to French dataset with 12M sentences consisting of 348M French words and 304M English words.
- Used 160,000 of the most frequent words for the source language and 80,000 of the most frequent words for the target language
- Every out-of-vocabulary word was replaced with a special "UNK" token

Seq2Seq Learning

Architecture details:

- 4 LSTM layers.

Seq2Seq Learning

Architecture details:

- 4 LSTM layers.
- 1000 LSTM cells in each layer.

Architecture details:

- 4 LSTM layers.
- 1000 LSTM cells in each layer.
- 1000 dimensional word embeddings.

Seq2Seq Learning

Architecture details:

- 4 LSTM layers.
- 1000 LSTM cells in each layer.
- 1000 dimensional word embeddings.
- Achieved BLEU score of 33.3 on WMT'14 English-to-French dataset.

Google NMT

Google NMT

- Paper describing all the details about the Google's MT system.

Google NMT

- Paper describing all the details about the Google's MT system.
- Heavily borrows from the previous two papers.

Google NMT

- Paper describing all the details about the Google's MT system.
- Heavily borrows from the previous two papers.
- Also, adds almost all the nice ideas in Deep learning research in the last few years.

Google NMT

- Paper describing all the details about the Google's MT system.
- Heavily borrows from the previous two papers.
- Also, adds almost all the nice ideas in Deep learning research in the last few years.
- Not just a research idea but already serves billions of queries a day.

Residual learning

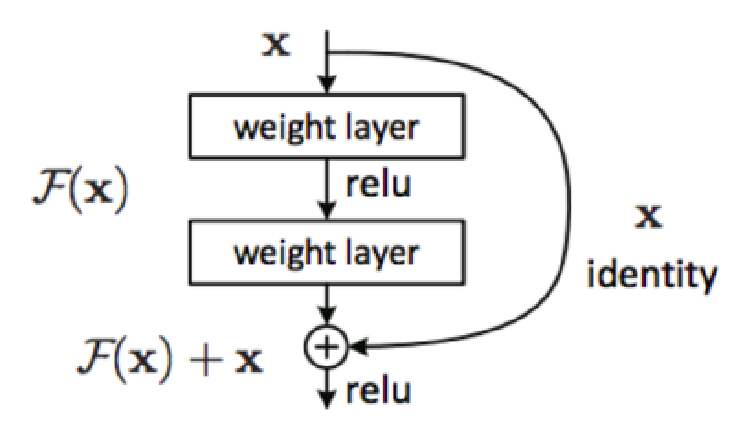


Figure: Residual networks

Residual connections enables training of very deep networks.[2]

[2] Deep Residual Learning for Image Recognition. Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. CVPR, 2016.

Google NMT with Residual connection

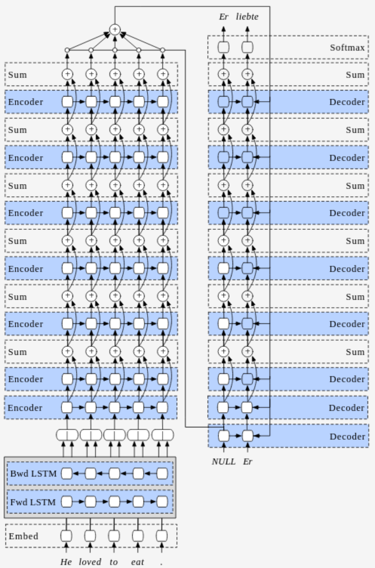


Figure: GNMT with residual connections.

Google NMT

Error function

- Standard maximum-likelyhood:

$$\mathbb{O}_{ML}(\Theta) = \sum_{i=1}^{N} log P_{\Theta}(Y^{*(i)}|X^{(i)})$$

Google NMT

Error function

- Standard maximum-likelyhood:

$$\mathbb{O}_{ML}(\Theta) = \sum_{i=1}^{N} logP_{\Theta}(Y^{*(i)}|X^{(i)})$$

- Refined error function aiming to maximize BLEU:

$$\mathbb{O}_{ML}(\Theta) = \sum_{i=1}^{N} \sum_{Y=\mathbb{Y}}^{N} logP_{\Theta}(Y^{*(i)}|X^{(i)})r(Y, Y^{*(i)})$$

where $r(\Delta)$ is per-sentence score computed as an expectation over all Y upto certain-length.

Google NMT

Google NMT properties

- Achieved a BLEU score of 38.95 BLEU on WMT'14 English-to-French dataset.

Google NMT

Google NMT properties

- Achieved a BLEU score of 38.95 BLEU on WMT'14 English-to-French dataset.
- One Encoder and one Decoder for all the languages.

Google NMT

Google NMT properties

- Achieved a BLEU score of 38.95 BLEU on WMT'14 English-to-French dataset.
- One Encoder and one Decoder for all the languages.
- This joint training of languages improves accuracy for languages for which not much training data exist.

Google NMT

Google NMT properties

- Achieved a BLEU score of 38.95 BLEU on WMT'14 English-to-French dataset.
- One Encoder and one Decoder for all the languages.
- This joint training of languages improves accuracy for languages for which not much training data exist.
- The input language are encoded using word2vec for all languages.

Google NMT

Google NMT properties

- Achieved a BLEU score of 38.95 BLEU on WMT'14 English-to-French dataset.
- One Encoder and one Decoder for all the languages.
- This joint training of languages improves accuracy for languages for which not much training data exist.
- The input language are encoded using word2vec for all languages.
- One additional token $(< \_\_EN\_\_ >, < \_\_FR\_\_ >, < \_\_DE\_\_ >, < \_\_ES\_\_ >)$ indicating the target language to be generated.

Google NMT

Google NMT properties

- Achieved a BLEU score of 38.95 BLEU on WMT'14 English-to-French dataset.
- One Encoder and one Decoder for all the languages.
- This joint training of languages improves accuracy for languages for which not much training data exist.
- The input language are encoded using word2vec for all languages.
- One additional token $(< \_\_EN\_\_ >, < \_\_FR\_\_ >, < \_\_DE\_\_ >, < \_\_ES\_\_ >)$ indicating the target language to be generated.
- One giant model that runs all Google translate queries.

Conclusion

Neural Machine Translation systems,

- Are State-of-the-art in machine translation.
- Greatly benefited from the neural network research by other communities.
- Used in production by companies like Google, Microsoft, Facebook, etc.

Conclusion

- Actually, I lied to you all.

Conclusion

- Actually, I lied to you all.
- Maybe, we don't need such a complex network like GNMT to achieve better results.

Conclusion

- Actually, I lied to you all.
- Maybe, we don't need such a complex network like GNMT to achieve better results.
- "Attention Is All You Need"arxiv preprint from Google threw away all LSTMS, Residual connections, etc., but managed to achieve BLEU score of 41.0 with only feedforward connections and attention.