Evaluation of Two Different Models in Neural Machine Translation

Peidong ZHANG
School of Computer Science and Engineering
Beihang University
XueYuan Road No.37, HaiDian District, Beijing
China
peidong.zhang@buaa.edu.cn

Abstract: - This paper introduces two different models in Neural Machine Translation task and evaluates their performances. In the evaluation, the research is based on the Chinese-English news and English-Chinese news machine translation evaluation task. For the translation of the Chinese-English orientation, we used traditional hierarchical phrase model. In the English-Chinese translation direction, we used not only the traditional hierarchical phrase model but also the RNN neural network model and the attention model. At the same time, we compared and analyzed the translation results. In the remainder of this article, we introduce the system framework, data process approach and evaluation results in various evaluation tasks.

Key-Words: - Machine Translation; Hierarchical Phrase Model; Neural Network; Attention Model

1 Introduction

This paper completes the evaluation of two machine translations of Chinese-English news and English-Chinese news. In the Chinese-English translation, we used the traditional machine translation based on the hierarchical phrase model. In the English-Chinese translation direction, both the machine translation model based on the hierarchical phrase and the RNN and attention model were used. The neural network model was compared and analyzed for translation results. Finally, the paper gives the understanding and summary of data processing, information transmission in traditional machine translation and neural network machine translation.

2 System Description

2.1 Joshua - A Phrase-based Statistical Machine Translation System

Joshua [1] is an advanced open source SMT system developed by the Johns Hopkins University [Lang et al. Language and Speech Processing, 2009] Language Speech Processing Center. The model used in Joshua is a layered phrase-based model proposed by [Chiang, 2005]. In addition to the basic model, it provides some interesting features such as SCFGs decoding (syntax annotation), multi-method decoding and parallel training and Map-reduce. The Joshua system is implemented in the Java language and has good scalability and portability on multiple platforms. It is one of the statistical machine translation systems with stable performance at present, which can reach the baseline of the hierarchical phrase model.

2.2 Sockeye - A Neural Network Translation System based on MXNet

Sockeye is Amazon's open source neural network machine translation framework in 2017. Sockeye provides an implementation of the current optimal neural machine translation (NMT) model and a platform for conducting NMT research. Sockeye is a fast and extensible deep learning library based on Apache MXNet. The Sockeye codebase has a unique advantage from MXNet. For example, through the symbolic and imperative MXNet APIs, Sockeye combines declarative and imperative programming styles; it can also train models in parallel on multiple GPUs. The Sockeye architecture based on Apache MXNet takes most of the work to build, train, and run the current optimal sequence-to-sequence model.

3 Methods

3.1 Phrase-based Statistical Machine Translation Model

In this paper, for the Chinese-English translation, the research team used the open source tool Joshua to build a statistical machine translation system based on the hierarchical phrase model. The hierarchical phrase model extracts non-contiguous parts of meta-language sentences. The statistical machine translation system based on hierarchical phrases is a formal grammar translation system. The

synchronous context-free grammar (SCFG) is used to establish the translation model. The rule form is as shown in formula (1):

$$x \rightarrow <\gamma, \alpha, \sim> \quad (1)$$

Where x is a non-terminal, γ and α are strings of terminal and non-terminal characters at the source and target languages, and ∼ is a one-to-one correspondence between non-terminals in γ and α.

The hierarchical phrase model uses the SCFGs grammar to obtain SCFGs from the word-aligned parallel corpus through heuristic rules, first extracting the initial phrase pairs, and then obtaining the hierarchical phrase rules. The hierarchical phrase model uses phrase rules. Similar to the phrase-based method, it can translate continuous source language word strings into target language word strings. At the same time, it introduces variables for hierarchical rules and can implement phrase ordering function.

The translation of a hierarchical phrase model is often seen as a process of derivation from a continuous use process. The translation model uses a log-linear model. Characteristic functions including translation probability P(γ|α), lexical weights P_w (γ| α) and P_w (α|γ), n-gram language model, number of rules, and number of target words are used. The translation system ultimately selects the derivation with the largest score to generate translation results.

Joshua uses GIZA++ [Och and Ney, 2005] [2] to train word alignment models and extract phrase pairs using Kenlm to train language models. Based on the Minimum Error Rate Training (MERT) [Och et al, 2003] [3], the logarithmic linear model parameters were adjusted by the development set.

3.2 Neural Network Machine Translation Model Based on LSTM and Attention Model

In this paper, we used the deep learning framework open source neural framework Sockeye for English-Chinese translation.

The neural network machine translation model uses a sequence-to-sequence (seq2seq) and a coding and decoding model of the attention model. The encoding end converts the input sentence into a vector representation of the hidden layer through the multi-layer RNN. The decoding end is also a multi-layer RNN structure. After decoding the state of the hidden layer, the one-hot vector is obtained by softmax and finally the sentence of the target language is obtained. . Google's GNMT [4] released in 2016 has implemented industrial-grade production applications in the translation of multiple language pairs and proved its effectiveness.

This evaluation uses the Long Short Term (LSTM), a long-term and short-term memory model. This is a special type of RNN that can learn long-term dependency information. The structure of LSTM is shown in Figure 1.

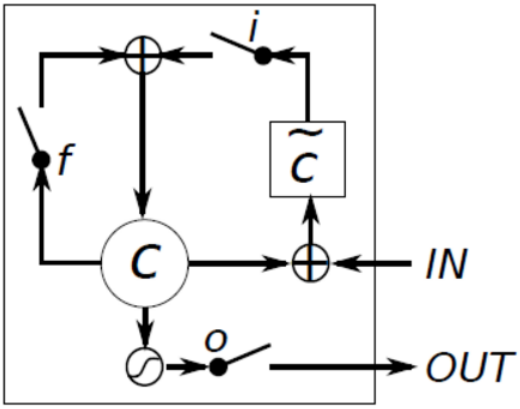

Figure 1. LSTM Architecture

LSTM was proposed by Hochreiter & Schmidhuber (1997) and has many improvements. In many problems, LSTM has achieved considerable success and has been widely used. LSTM avoids long-term dependencies through deliberate design. Remember that long-term information is the default behavior of LSTM in practice, not the ability to get it at a great price.

All RNNs have a chained form of a repetitive neural network module. In a standard RNN, this repetitive module has only a very simple structure, such as a tanh layer, such as Equation 2.

$$S_t = f(U \cdot x_t + W \cdot S_{t-1}) \quad (2)$$

S_t represents the state of the RNN, U and W are the parameter matrices of the network, and x_t is the input at time t.

LSTM also has this chain structure, but its repeating module structure is different. There are four neural network layers in the LSTM repeating module, and the interaction between them is very special, including three sigmoid and one tanh layer. The processing of memory by LSTM has changed from the tanh operation of the original RNN to the addition operation, effectively alleviating the problem of gradient disappearance.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (3)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (4)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (5)$$

C_t is the memory of LSTM. The new memory is the sum of the old memory and the newly generated memory after the forgetting gate, so that f_t can be preserved in the wrong back propagation without approaching zero.

The early neural network translation model is a simple sequence-to-sequence structure. After the coding end integrates the information, the decoder outputs one output, as shown in Figure 2:

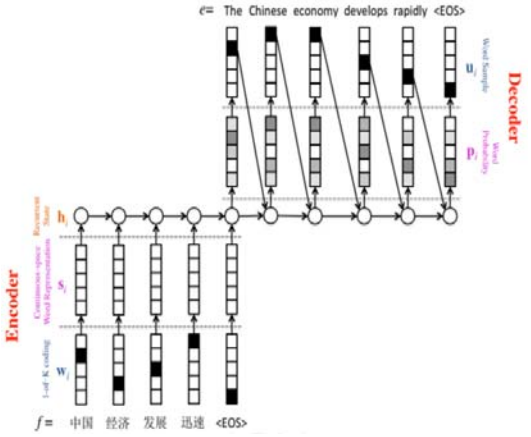

Figure 2. Sequence-to-sequence Neural Network Translation Model

The above model can't handle the translation problem very well. Especially when the sentence is too long, the semantics of the original sentence will be forgotten during decoding, so that the translated content is only partially fluent but the sentence does not match the original. Improving this is another important component of the current popular neural network machine translation model, the attention model. The addition of the attention model makes the neural network model truly transcend the traditional statistical machine translation model. The attention mechanism was first implemented by Bahdanau et al. [5] in the field of machine translation in 2015, and then improved by Luong et al. [6] in 2015. The key to the attention mechanism is to establish a direct connection between the target file and the source file by "paying attention" to the contents of the relevant source file during the translation process. The attention model makes the output from the last RNN of the attention coding side to the hidden layer output of the entire input sentence. In the process of translation, the contribution of each word in the source language to the translation of the current word is different. The attention model enables the neural network translation to pay more attention to the most important source language words for the current translated words. The decoding part of the formula after introducing the attention model is as follows:

$$p(y_i|y_1, \ldots, y_{i-1}, \mathrm{x}) = g(y_{i-1}, s_i, c_i) \quad (6)$$

Where y_i represents the i-th word to be translated by the target language sentence, s_i represents the hidden layer output at the i-th moment of the decoding end, and c_i represents the attention vector obtained by the attention model when the i-th word is translated. The attention vector c_i is the weighted sum of the output of the hidden layer at the encoding end, and needs to be calculated once each time a word is translated. The calculation of the weight here needs to calculate the alignment score of the hidden layer output of the encoding end and the decoding end. The alignment score measures the correlation between each word of the source language end and the output of the current hidden layer. Currently, there are various calculation methods.

The model of this evaluation uses two layers of LSTM networks at the encoding and decoding ends, where the encoding end uses bidirectional LSTM and the decoding end is unidirectional LSTM. The attention model uses a global attention model with a single-layer perceptron [Luong et al 2016], and the network model is shown in Figure 3.

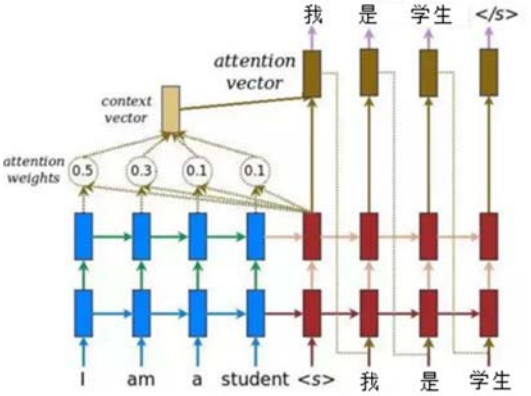

Figure 3. Sequence-to-sequence Neural Network Translation Model with Attention Model

3.3 Solving the problem of neural network open vocabulary by using Subword Unit

The problems of OOV (outer words) and rare words (Rare Words) in the neural network translation model are usually not translated or solved by the back-off dictionary. Since the neural network translation model needs to maintain a fixed dictionary, it can't cover all the words, and the bilingual control dictionary often cannot translate properly. Therefore, these two methods can't solve the problem that the words cannot be translated well.

Considering the word structure and laws of named entities, homologous words, foreign words,

compound words (there are a large proportion of rare words are the above), Sennrich et al. [7] split rare words into "subword units" (subword units). The combination of this can effectively alleviate the problem of OOV and rare word translation in neural network machine translation.

The splitting strategy of subword units is based on a data compression algorithm Byte Pair Encoding (BPE) (Gage, 1994). Unlike the Huffman coding proposed by (Chitnis and DeNero, 2015), the compression algorithm here is not for variable length coding of words, but for subwords. In this way, even if new words are not seen in the training corpus, the translation can be generated by splicing the subwords. Subword unit is a text representation unit between characters and words, and also different from character n-gram. It draws on the BPE compression algorithm to achieve a more balanced state in terms of vocabulary size and text length. The source/near source language pair has a good effect, and it also has a good performance in the English-Chinese model. It has a great help in dealing with rare words, especially reducing the size of the dictionary.

4 Experiments and analysis

4.1 System hardware configuration

The computer configuration and operating system used by the Chinese-English statistical machine translation model are shown in Table 1.

CPU	Memory	OS
Xeon(R) CPU E5-2630 v4 @ 2.20GHz	256G	CentOS7

Table 1. Machine configuration used in Chinese-English models

The computer configuration and operating system used in the English-Chinese neural network machine translation model are shown in Table 2.

CPU	Memory	OS	GPU
Xeon(R) CPU E5-2630 v4 @ 2.20GHz	64G	CentOS7	GTX 1080

Table 2. Machine configuration for English and Chinese models

4.2 Experimental data and data processing

4.2.1 Experimental data

This usage data is the training data provided by UN data. The pre-processed training data is shown in Table 3.

4.2.2 Experimental Data Processing

Since many of the data are obtained from public data sets such as the Internet, some of the data quality is not high, there are problems such as poor sentence alignment, or useless html tags, so this experiment first made basic data. filter. At the same time, considering that the neural network has higher corpus requirements and is more sensitive to noise than statistical-based machine translation, the standard setting of the screening is stricter.

Experiment	Train set	Validation set	Model
English-Chinese news	8904525 parallel sentences	1004 parallel sentences 1 reference	18897730 parallel sentences (neural network model not used)
Chinese-English news	8904525 parallel sentences	4793 parallel sentences 4 reference	12395041parallel sentences

Table 3. System usage data

After understanding the basic corpus situation, the basic rules of the screening are set as follows. Deleting refers to deleting the Chinese and English sentence pairs:

1. A sentence pair containing some html tags is directly deleted, such as a sentence containing a span class in a sentence;
2. There is a duplicate sentence pair in the previous sentence and this sentence, and the quality of the statement in this case is often not high;
3. If there is no Chinese character in the Chinese translation, delete it;
4. Delete if English is all uppercase;
5. If English characters appear in Chinese but do not appear in English, delete them;
6. If Chinese quotation marks cannot be matched before and after, delete them.

In the above screening process, the corpus is also repaired for the common errors. The basic rules for repair are as follows:

1. Remove the empty quotes. If there are pairs of quotes, but there is no content in the quotes, remove the quotes;
2. Remove the spaces between the numbers and remove the comma separator in the number, such as 2,000 modified to 2000;
3. Modify the errors in which multiple English words in Chinese are linked together;
4. Modify the English abbreviation and the quotation mark to separate the error, such as repairing the error data don't to don't, and changing the error data Tom's book to Tom's book.

After the above screening and repair is completed, the training set, verification set and test set required for the original model training are obtained. The processing of the data includes Chinese word segmentation and Chinese and English tokenize. For the neural network model, in addition to the operation, it is necessary to perform Chinese and English joint bpe processing.

Chinese word segmentation uses JIEBA word segmentation, Chinese tokenize uses MOSES tokenizer.perl. For machine translation models based on hierarchical phrases, English is treated in lowercase. The Chinese and English joint extraction subword unit, the iteration step is set to 32000.

4.3 Network Structure and Model Training

The encoding end uses two layers of LSTM, the bottom layer LSTM uses bidirectional LSTM, and the decoding end uses two layers of LSTM, all of which are single layers. The word vector dimensions of the source language and the target language are both set to 512. The number of nodes in all hidden layers is set to 1024, and the vector dimension in the attention model is also set to 512.

The sentence is set to a maximum of 100, and if it is exceeded, it is truncated. The dropout is set to 0.3, the batch size is set to 45, and one check breakpoint is set for every 5000 batches. The optimization algorithm uses Adam, the learning rate is initialized to 0.0003, and the learning rate is halved if the confusion level does not decrease for three consecutive times. The training ends when the number of times does not decrease for 8 consecutive times. Using a GTX 1080 GPU, the model trained for a total of about 4 days and experienced nearly 650,000 batches.

4.4 Experimental results and comparative analysis

4.4.1 Results of machine translation evaluation in English and Chinese news fields

The translation results are shown in Table 4:

System	Test set	BLEU4-SBP
PBMT	2017-ec-news	0.1593
NMT	2017-ec-news	0.2028

Table 4. Statistical machine translation and neural network machine translation results

4.4.2 Chinese-English news field machine translation evaluation results and word segmentation

The translation results are shown in Table 5:

System	Test set	BLEU-4
PBMT	2017-ce-news	0.1203

Table 5. Statistical machine translation results

4.4.3 Comparative analysis

From the comparison results, the number of scores of neural network machine translation is significantly higher than that of traditional phrase-based statistical machine translation. Manually comparing the sentences translated by the two models also found that the neural network model obtained a more fluid translation and closer to human language.

The sentences obtained by the neural network model translation are superior to the results obtained by the hierarchical phrase model in most lengths, but as the length of the sentence is longer, the performance is lower than that of the hierarchical phrase model. In this experiment, the sentence set by the neural network model is no longer than 100 words. When the input sentence exceeds 100 words, the sentence will be truncated. The hierarchical phrase model does not make this restriction, so when the sentence length becomes very long, the neural network The model cannot get the full semantics.

Neural network machine translation seems to have certain advantages in vocabulary-rich text, and the processing of word order is better than the phrase model. For example, the neural network model is better at dealing with verb positions. The neural network translation model also has its own advantages. For example, the sequence-to-sequence framework avoids many grammatical problems in traditional phrase-based statistical machine translation. The source language and the target language are respectively at both ends, and training does not require much extra. Processing, without a separate language model, translation model and

sequencing model, the model training process is also relatively linear and simple.

However, the neural network model also has its own shortcomings that are difficult to overcome. The multi-layer neural network in the neural network machine translation model can't have the physical meaning that is easy to explain like the phrase model. The training of the model and the propagation of errors are hard to split like a black box. There are many parameters of the neural network. The initialization of the parameters largely affects the final model effect. In addition, the process of model training may need to pay attention to relevant indicators from time to time, and change parameters or optimizers in time, which leads to the same model not necessarily every time. Can get the best results. At the same time, the neural network needs to maintain a fixed vocabulary. Once the input sentence contains a word that does not appear in the vocabulary or a word that does not appear in the vocabulary in the translated sentence, it will not be translated. The current subword unit handles it. This problem has been alleviated, but it has not been solved. The splitting of words will cause semantic loss, and there is no uniform standard for how to split. At the same time, Chinese characters cannot continue to be split, which also limits the use of subword units in Chinese.

Neural network translation models are more sensitive to data noise and require more data, which is unrealistic for translation in many languages. At the same time, the neural network machine translation model can only use bilingual parallel sentence pairs, and a large number of monolingual texts cannot be directly used, which further aggravates the problem of data scarcity.

In addition, the under-translation and over-translation problems of the neural network machine translation model are more prominent. Although the network can produce relatively smooth output, it often misses some of the semantics in the source language, and the more prominent is the modified part of the central word. Since the corpus is difficult to cover all situations, it is easy to miss the qualifier when attention is aligned, so that the fluency of the entire sentence is not affected, but the semantics are missing. Over-translation problems often occur along with under-translation problems. Sometimes translated sentences fall into a certain part, and a certain part of the translation is repeated, and even cannot be jumped out. This is especially serious in the translation of the connector "--". Based on the language model, the previous word is translated by "-", and the next big probability is also "-". Since the two sub-parts are identical, it is easy to fall into repeated translation. Although Input-feeding implies a coverage model, this issue is still not well resolved. Tu et al. [8] proposed a coverage model in 2016 papers, drawing on the concept of “coverage” in traditional statistical machine translation, and introducing a coverage model in neural machine translation. The model configures an overlay vector for each word of the source language sentence to store historical coverage information, but the model is not really improved in this evaluation when combined with the Luong model.

In general, the neural network machine translation model is a very good model. The semantic processing is very different from the traditional phrase model. It turns discrete words and phrases into continuous high-dimensional word vectors, although many times it is impossible. A good explanation, but this does not prevent it from succeeding in translating traditional phrase-based statistical translation models in more and more language translations.

5 Conclusions

This paper mainly introduces the progress in the neural machine translation domain. The research is based on Joshua and the neural network translation model Sockeye respectively trained the statistical machine translation model of the hierarchical phrase and the RNN and attention model. The experimental comparison between the statistical model and the neural network model of this research in English-Chinese translation shows that the neural network model has been much better than the traditional statistical machine translation model. It should be noted that the data used in this evaluation does not include the UN corpus. The overall corpus size is less than 10 million. I believe that more data can be obtained to get better results.

This evaluation is more to verify that neural network machine translation has a larger improvement than the traditional phrase-based statistical machine translation model. However, there is not enough corpus added, and the pre-processing and post-processing of the data are not optimized too much. The model is relatively simple and there is no resmble, which leads to relatively poor evaluation results and is inferior in the same kind of comparison. The next step is to add more corpus, to explore the help of corpus growth to improve the model's effect, and to further study the intrinsic physical meaning of the neural network

model, understand the training process, especially the principle and mechanism of the attention model.

References:

[1] Post M, Cao Y, Kumar G. Joshua 6: A phrase-based and hierarchical statistical machine translation system[J]. *Prague Bulletin of Mathematical Linguistics*, 2015, 104(1):5-16.

[2] Och F J, Ney H. A systematic comparison of various statistical alignment models[J]. *Computational linguistics*, 2003, 29(1): 19-51.

[3] Och F J. Minimum error rate training in statistical machine translation[C] //Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1. *Association for Computational Linguistics*, 2003: 160-167.

[4] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[J]. *Computer Science*, 2014.

[5] Luong M T, Pham H, Manning C D. Effective Approaches to Attention-based Neural Machine Translation[J]. *Computer Science*, 2015.

[6] Wu Y, Schuster M, Chen Z, et al. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation[J]. 2016.

[7] Sennrich R, Haddow B, Birch A. Neural Machine Translation of Rare Words with Subword Units[J]. *Computer Science*, 2015.

[8] Tu Z, Lu Z, Liu Y, et al. Modeling Coverage for Neural Machine Translation[J]. 2016:76-85.