Neural Clinical Paraphrase Generation with Attention

Sadid A. Hasan[1], **Bo Liu**[2], **Joey Liu**[1], **Ashequl Qadir**[1],
Kathy Lee[1], **Vivek Datla**[1], **Aaditya Prakash**[1,3], **Oladimeji Farri**[1]
[1]Artificial Intelligence Laboratory, Philips Research North America, Cambridge, MA, USA
[2]Auburn University, Auburn, AL, USA
[3]Brandeis University, Waltham, MA, USA
{sadid.hasan,joey.liu}@philips.com, {boliu}@auburn.edu
{ashequl.qadir,kathy.lee_1,vivek.datla,dimeji.farri}@philips.com
{aaditya.prakash,aprakash}@{philips.com,brandeis.edu}

Abstract

Paraphrase generation is important in various applications such as search, summarization, and question answering due to its ability to generate textual alternatives while keeping the overall meaning intact. Clinical paraphrase generation is especially vital in building patient-centric clinical decision support (CDS) applications where users are able to understand complex clinical jargons via easily comprehensible alternative paraphrases. This paper presents *Neural Clinical Paraphrase Generation (NCPG)*, a novel approach that casts the task as a monolingual neural machine translation (NMT) problem. We propose an end-to-end neural network built on an attention-based bidirectional Recurrent Neural Network (RNN) architecture with an encoder-decoder framework to perform the task. Conventional bilingual NMT models mostly rely on word-level modeling and are often limited by out-of-vocabulary (OOV) issues. In contrast, we represent the source and target paraphrase pairs as character sequences to address this limitation. To the best of our knowledge, this is the first work that uses attention-based RNNs for clinical paraphrase generation and also proposes an end-to-end character-level modeling for this task. Extensive experiments on a large curated clinical paraphrase corpus show that the attention-based NCPG models achieve improvements of up to 5.2 BLEU points and 0.5 METEOR points over a non-attention based strong baseline for word-level modeling, whereas further gains of up to 6.1 BLEU points and 1.3 METEOR points are obtained by the character-level NCPG models over their word-level counterparts. Overall, our models demonstrate comparable performance relative to the state-of-the-art phrase-based non-neural models.

1 Introduction

Paraphrasing, the act of generating the same semantic content as the source in the same language, can help gain performance improvements in many NLP applications. Examples include generating query variants or pattern alternatives for information retrieval, information extraction or question answering systems, creating reference paraphrases for automatic evaluation of machine translation and document summarization systems, and generating concise or simplified information for sentence compression or sentence simplification systems (Madnani and Dorr, 2010; Androutsopoulos and Malakasiotis, 2010). In particular, paraphrase generation has a significant value in developing patient-centric intelligent clinical decision support (CDS) applications where users are able to understand complex clinical jargons via easily comprehensible alternative paraphrases (Elhadad and Sutaria, 2007; Deléger and Zweigenbaum, 2009). For example, the complex clinical term *"nocturnal enuresis"* can be paraphrased as *"nocturnal incontinence of urine"* or *"bedwetting"* to better clarify a well-known condition associated with children.

Traditional paraphrase generation methods exploit hand-crafted rules (McKeown, 1983) or automatically learned complex paraphrase patterns (Zhao et al., 2009), use thesaurus-based (Hassan et al., 2007) or semantic analysis driven natural language generation approaches (Kozlowski et al., 2003), or leverage statistical machine learning theory and principles (Quirk et al., 2004; Wubben et al., 2010). In contrast, inspired by the recent success of bilingual neural machine translation (NMT) (Kalchbrenner and

Proceedings of the Clinical Natural Language Processing Workshop,
pages 42–53, Osaka, Japan, December 11-17 2016.

Blunsom, 2013; Sutskever et al., 2014; Cho et al., 2014b; Bahdanau et al., 2015) that shows promising performance compared to the traditional statistical machine translation (SMT) approaches, we propose neural clinical paraphrase generation (NCPG) by casting the task as a monolingual NMT problem. Unlike bilingual machine translation, monolingual machine translation considers the source language the same as the target language, which allows for its adaptation as a paraphrase generation task.

SMT systems (Koehn et al., 2003; Koehn, 2010) use a noisy channel model to identify an optimal target sentence that maximizes its conditional probability given a source sentence. Ideally, this process uses the Bayes' rule to distinctly maximize the KL-divergence between a language model and a translation model from a monolingual and a parallel corpus, respectively. However, NMT models are built from training a single end-to-end neural network architecture on a large parallel corpus that can directly optimize the conditional probability of an underlying sentence pair. Such models typically follow an encoder-decoder approach by building a pair of neural networks, where the first network acts as an encoder to generate a fixed-length vector representation of the source sentence, which is in turn decoded by the second network to form a target sentence (Sutskever et al., 2014; Cho et al., 2014b). Recurrent Neural Network (RNN) architectures with Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) or Gated Recurrent Units (GRU) (Cho et al., 2014a) are generally utilized to train the end-to-end state-of-the-art NMT systems. Another effective NMT model has been proposed recently, which follows an attention-based soft-search approach to improve the performance of the encoder-decoder architectures (Bahdanau et al., 2015). We use an attention-based bidirectional RNN architecture (Schuster and Paliwal, 1997; Bahdanau et al., 2015) with an encoder-decoder framework to build our NCPG models. Bidirectional RNNs have been shown to outperform unidirectional RNNs for sequence to sequence learning tasks (Jean et al., 2015).

NMT models mostly rely on word-level modeling that often causes an out-of-vocabulary (OOV) issue while predicting a target word given an unknown source word (Luong et al., 2015b). To address this limitation, we represent the source and target paraphrase pairs as character sequences and propose a character-level encoder-decoder framework for clinical paraphrase generation. To the best of our knowledge, this work is the first to adapt monolingual NMT for clinical paraphrase generation using an attention-based mechanism and also propose an end-to-end character-level NCPG model.

Extensive experiments on a large curated clinical paraphrase corpus built on a benchmark parallel paraphrase database, PPDB 2.0 (Pavlick et al., 2015b), along with a comprehensive medical metathesaurus (Lindberg et al., 1993) show that the proposed attention-based NCPG model can outperform an RNN encoder-decoder based strong baseline for word-level modeling, whereas character-level models can achieve further improvements over their word-level counterparts. Overall, the proposed models demonstrate comparable performance relative to the state-of-the-art phrase-based conventional machine translation models. The main contributions of our paper can be summarized as follows:

- We presented a novel approach for clinical paraphrase generation by casting the task as a monolingual neural machine translation problem. We proposed an end-to-end neural network model built on an attention-based bidirectional Recurrent Neural Network (RNN) architecture (Bahdanau et al., 2015) with an encoder-decoder framework to perform the task.

- We also presented a novel character-based neural clinical paraphrase generation approach to overcome the OOV issues encountered by the word-level models.

- We built a large curated paraphrase corpus using a benchmark parallel paraphrase database, PPDB 2.0 (Pavlick et al., 2015b) along with a comprehensive medical metathesaurus, UMLS (Lindberg et al., 1993) for our experiments.

- We conducted rigorous automatic and manual evaluations of our models. Results demonstrated that our proposed attention-based NCPG model can outperform an RNN encoder-decoder based strong baseline for word-level modeling, whereas character-level models can achieve further improvements. Overall, our models showed comparable performance relative to the state-of-the-art phrase-based non-neural machine translation models.

2 Related Work

Deep learning has been successfully applied to various NLP tasks in recent years. There are works that effectively apply recursive autoencoders (Socher et al., 2011) and convolutional neural networks (Yin and Schütze, 2015) for paraphrase recognition. However, paraphrase generation is a harder task due to the requirement of constructing semantically similar, grammatically accurate alternatives to a source sentence, and no prior work has attempted to solve this problem using deep learning.

Prior work that regards paraphrase generation as a monolingual machine translation task typically uses (non-neural) statistical machine translation (SMT) principles. Quirk et al. (2004) show the effectiveness of SMT techniques for paraphrase generation given adequate monolingual parallel corpus extracted from comparable news articles. Wubben et al. (2010) propose a phrase-based SMT framework for sentential paraphrase generation by using a large aligned monolingual corpus of news headlines. Zhao et al. (2008) propose a combination of multiple resources to learn phrase-based paraphrase tables and corresponding feature functions to devise a log-linear SMT model. Other models generate application-specific paraphrases (Zhao et al., 2009), leverage bilingual parallel corpora (Bannard and Callison-Burch, 2005) or apply a multi-pivot approach to output candidate paraphrases (Zhao et al., 2010).

Recently proposed NMT systems have shown excellent performance compared to the SMT systems by using RNN-based end-to-end deep neural network architectures (Sutskever et al., 2014; Cho et al., 2014b). Previous works that deploy RNNs have shown favorable results to model variable-length sequential inputs (Schuster and Paliwal, 1997; Sutskever et al., 2011; Graves, 2013; Kalchbrenner and Blunsom, 2013; Sperduti, 2015) while attention-based NMT models have shown better performance than the traditional encoder-decoder frameworks (Bahdanau et al., 2015; Luong et al., 2015a).

State-of-the-art NMT models usually perform word-level computations by limiting the size of the source and the target vocabulary and hence, suffer from OOV issues due to vocabulary incompatibility. This phenomenon may arise when a trained model has to deal with a previously unseen word during the testing phase (Luong et al., 2015b). Jean et al. (2015) use a large target vocabulary to address OOV issues for word-level NMT models while Luong et al. (2015b) introduce a post-processing step to translate OOV words using a dictionary. Since these approaches depend heavily on the time- and cost-effective process of developing or acquiring large volume dictionaries that may not scale across several domains, OOV issues still limit the accuracy of the word-based models. Based on the recent success of character-level modeling in resolving the OOV limitation (Bojanowski et al., 2015; Kim et al., 2015; Ling et al., 2015; Costa-JussÃ and Fonollosa, 2016; Chung et al., 2016), we propose a character-level NCPG model and perform relative comparisons with the word-level models.

Depending on the level of granularity, there can be different types of paraphrasing such as: lexical (e.g. *<automobile, car>*), phrasal (e.g. *<carry on, persist in>*), and sentential (e.g. *<The book was interesting, I enjoyed reading the book>*) (Madnani and Dorr, 2010). Earlier work related to clinical-domain specific paraphrasing uses some unsupervised textual similarity measures to generate/extract lexical and phrasal paraphrases from monolingual parallel and comparable corpora (Elhadad and Sutaria, 2007; Deléger and Zweigenbaum, 2009). Prud'hommeaux and Roark (2015) propose a graph-based word alignment algorithm to examine neurological disorders through analysis of spoken language data. Another loosely related recent work adopts a semi-supervised word embedding model for medical synonym extraction (Wang et al., 2015) that can be regarded as the simplest form of a lexical paraphrase extraction task. Our work is the first to propose a neural network-based architecture that can model word/character sequences to essentially address all granularities of paraphrase generation for the clinical domain.

For our experiments, we combine the *Paraphrase Database (PPDB) 2.0* (Pavlick et al., 2015b) with a large medical metathesaurus, known as *Unified Medical Language System (UMLS)* (Lindberg et al., 1993) to build a comprehensive monolingual parallel paraphrase corpus such that the proposed NCPG models can effectively learn discriminatory features related to complex clinical terms. Similar methods of combining general and domain-specific data have been proven to be useful for domain-focused paraphrasing tasks in the literature (Pavlick et al., 2015a).

3 Model Description

3.1 Task Formulation

Our NCPG system is an attention-based bidirectional RNN architecture (Schuster and Paliwal, 1997) that uses an encoder-decoder framework (Bahdanau et al., 2015). We construct different NCPG models by representing the source and target paraphrase pairs as word or character-level sequences.

The neural clinical paraphrase generation task can be formulated as follows: given a source sequence $x = x_0, \ldots, x_L$, generate a target paraphrase sequence $y = y_0, \ldots, y_M$, where x_i $(0 \leq i \leq L)$ and y_j $(0 \leq j \leq M)$ are the individual textual units (word/character), and L, M are the respective lengths of the source and the target sequences. Ideally, generation of the next target unit y_n depends on the source sequence x and the already generated target units $y_0, \ldots, y_{n-1}$. In the following subsections, we present a description of the generic RNN architecture and the attention-based NCPG model.

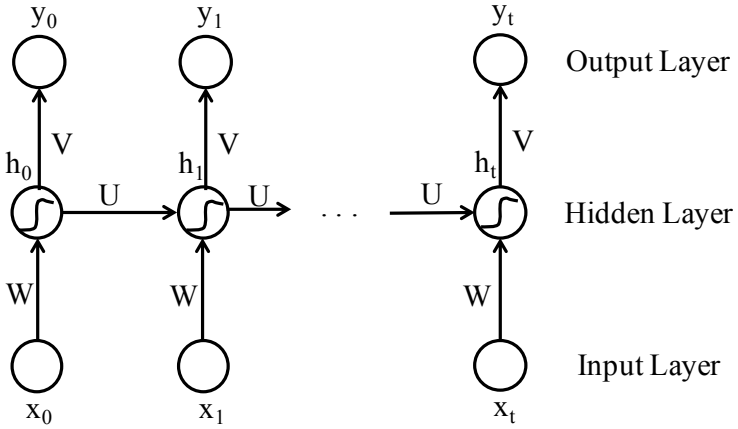

Figure 1: Generic recurrent neural network architecture.

3.2 Recurrent Neural Network (RNN)

RNNs are particularly suitable for modeling sequences and have been shown to perform well to solve various NLP tasks because of their ability to deal with variable-length input and output (Sutskever et al., 2011). The RNN network architecture is similar to the standard feedforward neural network with the exception that hidden unit activation at a particular time t is dependent on that of time $t-1$.

Figure 1 shows an unrolled RNN architecture, where x_t, y_t, h_t are the input, output, and hidden state at time step t, and W, U, V are the parameters of the model corresponding to *input*, *hidden*, and *output* layer weights (shared across all time steps).

The hidden state h_t is essentially the memory of the network as it can capture necessary information about an input sequence by exploiting the previous hidden state h_{t-1} and the current input x_t as follows:

$$h_t = f(Wx_t + Uh_{t-1}), \tag{1}$$

where f is an element-wise nonlinear activation function. The output y_t is computed similarly as a function of the memory at time t: Vh_t. Although RNN is theoretically a powerful model to encode sequential information, in practice it often suffers from the vanishing/exploding gradient problems while learning long-range dependencies (Bengio et al., 1994). LSTM (Hochreiter and Schmidhuber, 1997) and GRU (Cho et al., 2014a) networks are known to be successful remedies to these problems. We use GRU as the hidden layer activation unit in our paraphrase generation framework.

GRU is a simplified version of LSTM with less number of parameters per unit, thus the total number of parameters can be greatly reduced for a large neural network (Cho et al., 2014a). In contrast to LSTM, GRU does not have an internal memory state and the output gate, rather it introduces two gates termed *update* and *reset* as alternatives to the LSTM components. Specifically, GRU computes the hidden state h_t as follows:

$$
\begin{aligned}
z_t &= \sigma(W^z x_t + U^z h_{t-1}) \\
r_t &= \sigma(W^r x_t + U^r h_{t-1}) \\
k_t &= \tanh(W^k x_t + U^k (r_t \odot h_{t-1})) \\
h_t &= (1 - z_t) \odot k_t + z_t \odot h_{t-1},
\end{aligned} \tag{2}
$$

where z_t, r_t are the update gate and the reset gate, and k_t is the candidate hidden state. Note that, z_t, r_t are computed similarly as LSTM (using different weight parameters) where z_t determines how much of the old memory to keep while r_t denotes how much new information is needed to be combined with the old memory. Finally, k_t is computed by exploiting r_t, and h_t is calculated to denote the amount of information needed to be transmitted to the following layers.

3.3 Neural Clinical Paraphrase Generation (NCPG)

The architectural diagram of our paraphrase generation model is presented in Figure 2. In the encoder-decoder framework of our NCPG model, the encoder uses a bidirectional RNN architecture (Schuster and Paliwal, 1997; Bahdanau et al., 2015) where one forward RNN reads the input sequence to generate a hidden state sequence $(\overrightarrow{h_0}, \ldots, \overrightarrow{h_L})$ and one backward RNN reads the input sequence in the reverse order to generate a backward hidden state sequence $(\overleftarrow{h_0}, \ldots, \overleftarrow{h_L})$ using the GRU framework presented in Eq. 2.

Then, an annotation vector h_i for each textual unit x_i is obtained by concatenating the corresponding forward and backward hidden states as follows:

$$
h_i = \begin{bmatrix} \overrightarrow{h_i} \\ \overleftarrow{h_i} \end{bmatrix} \tag{3}
$$

Thus, h_i encodes all relevant information about the neighboring words or characters of x_i that is used in the decoding phase to compute the context vector of a potential target textual unit.

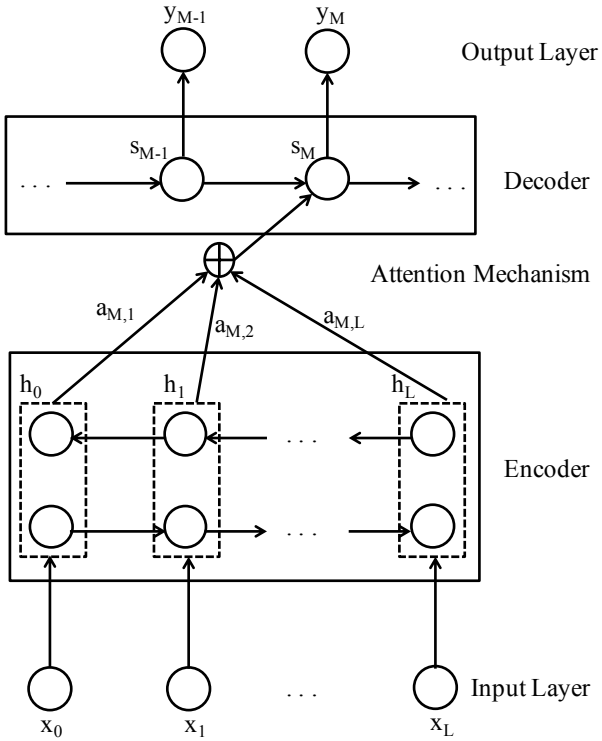

Figure 2: Model architecture for neural clinical paraphrase generation.

The decoder of our model consists of a forward RNN that is built over the generated paraphrase sequence $y = y_0, \ldots, y_{M-1}$ by creating a hidden state sequence $(\overrightarrow{s_0}, \ldots, \overrightarrow{s_{M-1}})$ where s_{M-1} essentially

encodes the context of the currently generated paraphrase units. Ideally, at each time step t, an attention mechanism in the decoder computes a relevance score a_{ti} for each annotation vector h_i and sums the weighted annotation vectors as the context vector c_t while generating the next paraphrase word/character y_t. Formally, c_t is computed as follows:

$$c_t = \sum_{i=0}^{L} a_{ti} h_i \tag{4}$$

The annotation relevance score a_{ti} determines the most relevant source unit to focus on and is computed as:

$$a_{ti} = \frac{\exp(e_{ti})}{\sum_{k=0}^{L} \exp(e_{tk})} \tag{5}$$

where e_{ti} is called the *alignment* model that determines how closely the source context at position i matches with the output at position t. e_{ti} is calculated with a feedforward neural network f based on the candidate annotation vector h_i and the previous hidden state s_{t-1} as:

$$e_{ti} = f(s_{t-1}, h_i) \tag{6}$$

Thus, the hidden state s_t of the decoder is computed by the forward RNN based on the previous hidden state s_{t-1}, previously generated textual units y_{t-1}, and the most relevant source context c_t:

$$s_t = g(s_{t-1}, y_{t-1}, c_t) \tag{7}$$

where g is the GRU unit as described in Eq. 2. The conditional distribution over the textual units is computed similarly using a feedforward neural network as follows:

$$P(y_M \mid y_1, \ldots, y_{M-1}, x) = f(y_{M-1}, s_M, c_M) \tag{8}$$

Thus, our encoder-decoder based NCPG model is jointly trained to maximize the conditional log-likelihood of the underlying monolingual parallel paraphrase corpus.

4 Experimental Setup

4.1 Corpus

We combine a publicly available large paraphrase corpus, *Paraphrase Database (PPDB) 2.0* (Pavlick et al., 2015b) with a large clinical database curated from the UMLS metathesaurus (Lindberg et al., 1993) to build a comprehensive monolingual parallel corpus. PPDB leverages multiple bilingual parallel corpora to construct millions of general domain paraphrases in different languages. PPDB 2.0 uses a supervised regression model-based ranking strategy to generate six database categories based on size. In this work, we use the English *S-size* pack[1] database with lexical and phrasal paraphrases.

We extract a subset of 1.2M paraphrases from PPDB with 3.3M words that contain only alphabetic characters. In addition, we consider all unique fully specified terms along with corresponding description terms from SNOMED CT (Cornet and de Keizer, 2008) as source and target paraphrases (total $140K$). The SNOMED CT terms are selected based on UMLS concept unique identifiers (CUI). For example, the fully specified term *"sensorineural hearing loss"* is set as the source and the corresponding description terms such as *"perceptive hearing loss"*, *"perceptive deafness"*, *"sensorineural deafness"*, and *"neurosensory deafness"* are set as the target paraphrases. Three-fifth of the combined corpus is used as the training set while the rest is equally divided into two parts to produce validation and test sets. We use a randomly selected subset of 5000 paraphrases from the test set to evaluate the performance of the proposed models.

We perform normalization with respect to case and standard tokenization to pre-process the dataset. For word-level models, a list of $30K$ most frequent words in each of the source and the target paraphrase set is used for training, while any out-of-vocabulary word is treated as a special *UNK* token. For char-level models, we tokenize text sequences into white-space delimited characters and use a special character (#) to preserve word boundaries.

[1]The S-size database pack is used since it contains only the highest scoring paraphrase pairs.

4.2 Models

For comparison, two types of models are trained. The first model (*NCPG-1*) is our baseline, which is built on a non-attention based RNN encoder-decoder framework (Cho et al., 2014b; Cho et al., 2014a; Sutskever et al., 2014), where an encoder (RNN) generates a fixed-length vector representation of the input sequence and a decoder (another RNN) is used to form a output sequence from this representation. The second model (*NCPG-2*) is our proposed attention-based bidirectional RNN encoder-decoder framework. Both models are trained with word-level and character-level sequences (for source-target paraphrase pairs) resulting in four neural clinical paraphrase generation models.

We use a one-hot vector approach to represent the textual units (words/chars) in all models. Each RNN is built with 1000 hidden units (i.e. GRU as discussed in Section 3.2). Models are trained with a stochastic gradient descent (SGD) algorithm with update direction computed using a mini batch of 32 paraphrase pairs. Due to the large size of recurrent networks, the batch-size was limited to 32. We train the models for approximately 150 hours using multiple GPU machines (Tesla K20m, and Tesla K80).

We use *Theano* (Bergstra et al., 2011) for all our experiments. We use RNN templates provided by the *GroundHog* library[2]. For training, we use the *Adadelta* learning scheme (Zeiler, 2012) with ρ as 0.95 and `eps` as 1e-6. We use early stopping to prevent overfitting.

We use a beam search algorithm to generate optimal paraphrases by exploiting the trained models in the testing phase (Sutskever et al., 2014). We also create a SMT model to compare the performance of the proposed models. We use the *Moses* package (Koehn et al., 2007) for this purpose, which uses a phrase-based approach by combining a translation model and a language model to generate paraphrases. We use the default settings to create the SMT model.

4.3 Evaluation and Analysis

4.3.1 Automatic Evaluation

To quantitatively evaluate the performance of our paraphrase generation models, we use two well-known automatic metrics for machine translation evaluation: BLEU (Papineni et al., 2002) and METEOR (Lavie and Agarwal, 2007). Previous work has shown that these metrics can perform well for the paraphrase recognition task (Madnani et al., 2012) and correlate well with human judgments in evaluating generated paraphrases (Wubben et al., 2010). BLEU considers exact matching between target paraphrases and system generated paraphrases by considering n-gram overlaps. Meanwhile, METEOR improves upon this measure via stemming and synonymy using WordNet. We compute BLEU scores with jBLEU V0.1.1 (an exact reimplementation of NIST's mteval-v13.pl without tokenization) and METEOR scores using METEOR V1.4 with all default settings (Clark et al., 2011).

Table 1 shows the average BLEU and METEOR scores for the NCPG models considering the source and the target paraphrases as references to the system generated paraphrases. The input/prediction level for all models are denoted in parenthesis. *Moses* is the word-level statistical paraphrase generation model trained using the Moses package. *Source-Target* refers to the scores computed between the source and the target paraphrase pairs of the test set, because the source text is also a paraphrase of the target text. This can essentially serve as an upper bound of the paraphrasing scores (Wubben et al., 2010).

Our results show that all NCPG models perform relatively better than *Source-Target* in terms of BLEU scores. Similar trend is also seen for METEOR scores. We also observe that *Moses* obtains the highest scores, which is expected because Moses uses an additional monolingual training corpus of 418M words that was not used to train our NCPG models. Moreover, as BLEU and METEOR scores consider the number of word/synonym overlaps between the source and target paraphrase pairs, our qualitative evaluation (reported in the next subsection) reveals that Moses often repeats the source text as the generated target paraphrases and achieves higher scores for exact matching. This phenomenon is also evident from the *Source-Target* scores, which denote that models can achieve lower BLEU/METEOR scores even though they generate better quality paraphrases.

The results also reveal that the attention-based NCPG models mostly outperform the RNN encoder-decoder models, and char-level NCPG models perform considerably better than their word-level counterparts. Qualitative analysis revealed that word-level NCPG models largely suffered from OOV issues

[2]https://github.com/lisa-groundhog/GroundHog

while char-level models could efficiently deal with this problem. This is a noteworthy achievement because our character-level models do not require language-dependent grammatical pre-processing and they learn from efficient encoding of character sequences while being tolerant to spelling errors, a very common occurrence in clinical documents. We hypothesize that the results of the char-level models would further improve if pre-trained character embeddings based on a large background clinical corpus (e.g. biomedical literature corpus such as PubMed Central[3]) can be used during training.

Model	BLEU	METEOR
NCPG-1 (Word)	18.8	30.5
NCPG-1 (Char)	31.3	32.1
NCPG-2 (Word)	24.0	31.0
NCPG-2 (Char)	30.1	32.3
Moses	50.2	47.0
Source-Target	14.6	26.2

Table 1: Automatic evaluation scores for all models.

4.3.2 Human Evaluation

Automatic evaluation of paraphrasing is difficult as BLEU and METEOR can capture the textual similarity while disregarding the novelty of the generated paraphrases (Callison-Burch et al., 2008). Hence, we conduct human evaluation to qualitatively evaluate the performance of our NCPG models. We use a methodology derived from Wubben et al. (2010) for this purpose. Five judges (familiar with the clinical domain) evaluated the quality of a randomly selected subset (2%) of the paraphrases from the test set using three criteria: 1) *semantic relatedness:* whether the overall meaning is preserved in the paraphrase, 2) *novelty*[4]*:* if the paraphrase is considerably different from the source text, and 3) *grammaticality:* if the paraphrase is syntactically correct and fluent. The judges were presented with the source and the target text along with the system generated paraphrases. Note that, the target text is considered as one of many candidate paraphrases of the source text. For each of the criteria, the judges assigned an integer score between 1 (very poor) and 5 (very good) to each paraphrase. System settings and model identities were not disclosed to the judges during evaluation.

Table 2 shows the average quality scores for all models. These results demonstrate that on average, our attention-based models (*NCPG-2*) outperform the *NCPG-1* models, and char-level models perform better than word-level models in terms of semantic relatedness and grammaticality while underperforming in terms of novelty. Furthermore, our word-level NCPG models perform better than *Moses* in terms of novelty (up to 22% improvement) as *Moses* often generates the same paraphrase as the source sequence. These results show that on average, our proposed models perform on par with *Moses* and *Source-Target*.

Model	Meaning	Novelty	Grammaticality	Average
NCPG-1 (Word)	3.23	2.65	3.78	3.22
NCPG-1 (Char)	3.28	2.23	4.02	3.18
NCPG-2 (Word)	3.18	2.90	3.84	3.31
NCPG-2 (Char)	3.36	2.30	3.95	3.20
Moses	3.83	2.38	4.06	3.42
Source-Target	3.47	2.90	4.16	3.51

Table 2: Human evaluation scores for all models.

[3]http://www.ncbi.nlm.nih.gov/pmc/

[4]Novelty is inherently dependent on semantic relatedness because new words that do not preserve the overall meaning of the source text are undesirable.

4.3.3 Example Paraphrases

Table 3 presents some example source and target texts with corresponding system generated paraphrases from our models. These examples suggest that the word-level NCPG models generate better quality clinical paraphrases similar to *Moses*[5]. Also, char-level NCPG models perform well in generating comparable paraphrase texts. This confirms the effectiveness of the proposed NCPG models. Note that our curated corpus is mostly built on lexical and phrasal paraphrases. In future, we plan to construct a sentence-level parallel clinical paraphrase corpus to test the performance of our NCPG models for sentential paraphrasing.

Source: contagious diseases	**Target:** communicable diseases
Model	**Paraphrase**
NCPG-1 (Word)	habitat
NCPG-1 (Char)	contact diseases
NCPG-2 (Word)	an infectious disease
NCPG-2 (Char)	the diseases
Moses	infectious diseases
Source: secondary malignant neoplasm of spleen	**Target:** secondary malignant deposit to spleen
Model	**Paraphrase**
NCPG-1 (Word)	secondary cancer of spleen
NCPG-1 (Char)	separation of spleen
NCPG-2 (Word)	secondary malignant neoplasm of spleen
NCPG-2 (Char)	secondary malignant neoplasm
Moses	metastatic ca spleen
Source: abdominal lymph node structure	**Target:** intraabdominal lymph node
Model	**Paraphrase**
NCPG-1 (Word)	abdominal lymph node
NCPG-1 (Char)	abdominal lymph nodes
NCPG-2 (Word)	abdominal lymph nodes
NCPG-2 (Char)	abdominal lymph
Moses	retroperitoneal node sructure

Table 3: Paraphrase examples.

5 Conclusion and Future Work

In this paper, we proposed a novel approach called *neural clinical paraphrase generation* by using the monolingual NMT principles. We used an attention-based bidirectional RNN encoder-decoder framework to build an end-to-end architecture to accomplish the task by considering both word-level and char-level computations. To the best of our knowledge, this work is the first that uses attention-based RNNs for clinical paraphrase generation and also proposes an end-to-end character-level modeling for this task. Extensive automatic and human evaluation on a large curated parallel corpus demonstrated that the proposed NCPG models can outperform an RNN encoder-decoder based strong baseline while performing on par with the traditional SMT models. We also showed that character-based NCPG models can often outperform word-level models to remedy the OOV issues while generating paraphrases. In future, we will experiment with alternative structures for character-level RNN-based (Bojanowski et al., 2015) neural paraphrase generation architectures, and exploit a larger monolingual clinical paraphrase corpus to enhance the performance of our models.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and feedback.

[5]Note *Moses* has misspelled the word "structure" in the last example.

References

I. Androutsopoulos and P. Malakasiotis. 2010. A Survey of Paraphrasing and Textual Entailment Methods. *Journal of Artificial Intelligence Research*, 38(1):135–187.

D. Bahdanau, K. Cho, and Y. Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of ICLR*, pages 1–15.

C. Bannard and C. Callison-Burch. 2005. Paraphrasing with Bilingual Parallel Corpora. In *Proceedings of ACL*, pages 597–604.

Y. Bengio, P. Simard, and P. Frasconi. 1994. Learning Long-Term Dependencies with Gradient Descent is Difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166.

J. Bergstra, F. Bastien, O. Breuleux, P. Lamblin, R. Pascanu, O. Delalleau, G. Desjardins, D. Warde-Farley, I. Goodfellow, A. Bergeron, et al. 2011. Theano: Deep Learning on GPUs with Python. In *NIPS 2011, BigLearning Workshop, Granada, Spain*.

P. Bojanowski, A. Joulin, and T. Mikolov. 2015. Alternative structures for character-level RNNs. In *arXiv:1511.06303 [cs.LG]*.

C. Callison-Burch, T. Cohn, and M. Lapata. 2008. ParaMetric: An Automatic Evaluation Metric for Paraphrasing. In *Proceedings of COLING*, pages 97–104.

K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio. 2014a. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111.

K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. 2014b. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of EMNLP*, pages 1724–1734.

J. Chung, K. Cho, and Y. Bengio. 2016. A Character-level Decoder without Explicit Segmentation for Neural Machine Translation. In *arXiv:1603.06147 [cs.CL]*.

J. H. Clark, C. Dyer, A. Lavie, and N. A. Smith. 2011. Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability. In *Proceedings of ACL-HLT*, pages 176–181.

R. Cornet and N. de Keizer. 2008. Forty Years of SNOMED: A Literature Review. *BMC Medical Informatics and Decision Making*, 8(S-1):S2:1–7.

M. R. Costa-JussÃ and J. A. R. Fonollosa. 2016. Character-based Neural Machine Translation. In *arXiv:1603.00810 [cs.CL]*.

L. Deléger and P. Zweigenbaum. 2009. Extracting Lay Paraphrases of Specialized Expressions from Monolingual Comparable Medical Corpora. In *Proceedings of the 2nd Workshop on Building and Using Comparable Corpora: From Parallel to Non-parallel Corpora*, pages 2–10.

N. Elhadad and K. Sutaria. 2007. Mining a Lexicon of Technical Terms and Lay Equivalents. In *Proceedings of the Workshop on BioNLP*, pages 49–56.

A. Graves. 2013. Generating Sequences With Recurrent Neural Networks. In *arXiv:1308.0850 [cs.NE]*.

S. Hassan, A. Csomai, C. Banea, R. Sinha, and R. Mihalcea. 2007. UNT: SubFinder: Combining Knowledge Sources for Automatic Lexical Substitution. In *Proceedings of SemEval*, pages 410–413.

S. Hochreiter and J. Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

S. Jean, K. Cho, R. Memisevic, and Y. Bengio. 2015. On Using Very Large Target Vocabulary for Neural Machine Translation. In *Proceedings of ACL*, pages 1–10.

N. Kalchbrenner and P. Blunsom. 2013. Recurrent Continuous Translation Models. In *Proceedings of EMNLP*, pages 1700–1709.

Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush. 2015. Character-Aware Neural Language Models. In *arXiv:1508.06615 [cs.CL]*.

P. Koehn, F. J. Och, and D. Marcu. 2003. Statistical Phrase-based Translation. In *Proceedings of NAACL-HLT*, pages 48–54.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of ACL Interactive Poster and Demo. Sessions*, pages 177–180.

P. Koehn. 2010. *Statistical Machine Translation*. Cambridge University Press, New York, NY, USA.

R. Kozlowski, K. F. McCoy, and K. Vijay-Shanker. 2003. Generation of Single-sentence Paraphrases from Predicate/Argument Structure Using Lexico-grammatical Resources. In *Proceedings of the 2nd International Workshop on Paraphrasing*, pages 1–8.

A. Lavie and A. Agarwal. 2007. METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231.

D. Lindberg, B. Humphreys, and A. McCray. 1993. The Unified Medical Language System. *Methods of Information in Medicine*, 32(4):281–291.

W. Ling, I. Trancoso, C. Dyer, and A. W. Black. 2015. Character-based Neural Machine Translation. In *arXiv:1511.04586 [cs.CL]*.

T. Luong, H. Pham, and C. D. Manning. 2015a. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of EMNLP*, pages 1412–1421.

T. Luong, I. Sutskever, Q. V. Le, O. Vinyals, and W. Zaremba. 2015b. Addressing the Rare Word Problem in Neural Machine Translation. In *Proceedings of ACL-IJCNLP*, pages 11–19.

N. Madnani and B. J. Dorr. 2010. Generating Phrasal and Sentential Paraphrases: A Survey of Data-driven Methods. *Computational Linguistics*, 36(3):341–387.

N. Madnani, J. Tetreault, and M. Chodorow. 2012. Re-examining Machine Translation Metrics for Paraphrase Identification. In *Proceedings of NAACL-HLT*, pages 182–190.

K. R. McKeown. 1983. Paraphrasing Questions Using Given and New Information. *Computational Linguistics*, 9(1):1–10.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of ACL*, pages 311–318.

E. Pavlick, J. Ganitkevitch, T. P. Chan, X. Yao, B. Van Durme, and C. Callison-Burch. 2015a. Domain-specific paraphrase extraction. In *Proceedings of ACL-IJCNLP*, pages 57–62, Beijing, China.

E. Pavlick, P. Rastogi, J. Ganitkevitch, B. Van Durme, and C. Callison-Burch. 2015b. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of ACL-IJCNLP)*, pages 425–430.

E. Prud'hommeaux and B. Roark. 2015. Graph-Based Word Alignment for Clinical Language Evaluation. *Computational Linguistics*, 41(4):549–578.

C. Quirk, C. Brockett, and W. Dolan. 2004. Monolingual Machine Translation for Paraphrase Generation. In *Proceedings of EMNLP*, pages 142–149.

M. Schuster and K.K. Paliwal. 1997. Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

R. Socher, E. H. Huang, J. Pennington, A. Y. Ng, and C. D. Manning. 2011. Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection. In *Advances in Neural Information Processing Systems*, pages 1–9.

A. Sperduti. 2015. Equivalence Results between Feedforward and Recurrent Neural Networks for Sequences. In *Proceedings of IJCAI*, pages 3827–3833.

I. Sutskever, J. Martens, and G. E. Hinton. 2011. Generating Text with Recurrent Neural Networks. In *Proceedings of ICML*, pages 1017–1024.

I. Sutskever, O. Vinyals, and Q. V. Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Annual Conference on Neural Information Processing Systems*, pages 3104–3112.

C. Wang, L. Cao, and B. Zhou. 2015. Medical Synonym Extraction with Concept Space Models. In *Proceedings of IJCAI*, pages 989–995.

S. Wubben, A. van den Bosch, and E. Krahmer. 2010. Paraphrase Generation As Monolingual Translation: Data and Evaluation. In *Proceedings of INLG*, pages 203–207.

W. Yin and H. Schütze. 2015. Convolutional Neural Network for Paraphrase Identification. In *Proceedings of NAACL-HLT*, pages 901–911.

M. D. Zeiler. 2012. ADADELTA: An Adaptive Learning Rate Method. In *arXiv:1212.5701 [cs.LG]*.

S. Zhao, C. Niu, M. Zhou, T. Liu, and S. Li. 2008. Combining Multiple Resources to Improve SMT-based Paraphrasing Model. In *Proceedings of ACL-HLT*, pages 1021–1029.

S. Zhao, X. Lan, T. Liu, and S. Li. 2009. Application-driven Statistical Paraphrase Generation. In *Proceedings of ACL-IJCNLP*, pages 834–842.

S. Zhao, H. Wang, X. Lan, and T. Liu. 2010. Leveraging Multiple MT Engines for Paraphrase Generation. In *Proceedings of COLING*, pages 1326–1334.