Learning to Generate Rock Descriptions from Multivariate Well Logs with Hierarchical Attention

Bin Tong Martin Klinkigt
Makoto Iwayama
Research and Development Group, Hitachi, Ltd.
bin.tong.hh,martin.klinkigt.ut, makoto.iwayama.nw@hitachi.com

Toshihiko Yanase
Yoshiyuki Kobayashi
Research and Development Group, Hitachi, Ltd.
toshihiko.yanase.gm,yoshiyuki.kobayashi.gp@hitachi.com

Anshuman Sahu
Ravigopal Vennelakanti
Big Data Laboratory, Hitachi America, Ltd.
anshuman.sahu,ravigopal.vennelakanti@hal.hitachi.com

ABSTRACT

In the shale oil & gas industry, operators are looking toward big data analytics to optimize operations and reduce cost. In this paper, we mainly focus on how to assist operators in understanding the subsurface formation, thereby helping them make optimal decisions. A large number of geology reports and well logs describing the sub-surface have been accumulated over years. Issuing geology reports is more time consuming and depends more on the expertise of engineers than acquiring the well logs. To assist in issuing geology reports, we propose an encoder-decoder-based model to automatically generate rock descriptions in human-readable format from multivariate well logs. Due to the different formats of data, this task differs dramatically from image and video captioning. The challenges are how to model structured rock descriptions and leverage the information in multivariate well logs. To achieve this, we design a hierarchical structure and two forms of attention for the decoder. Extensive validations are conducted on public well data of North Dakota in the United States. We show that our model is effective in generating rock descriptions. The two forms of attention enable the provision of a better insight into relations between well-log types and rock properties with our model from a data-driven perspective.

CCS CONCEPTS

• **Computing methodologies** → **Natural language generation**; **Neural networks**;

KEYWORDS

Geology Report; Well Log; Neural Generation; Attention

KDD'17, , August 13–17, 2017, Halifax, NS, Canada.

ACM ISBN 978-1-4503-4887-4/17/08...$15.00
https://doi.org/10.1145/3097983.3098132

1 INTRODUCTION

Technology advances such as horizontal drilling and hydraulic fracturing have been resulting in a booming period in the shale oil & gas industry. However, according to a survey [17] conducted among companies in the shale gas industry, the production of about 40% of shale wells underperformed expectations. The major cause resulting in this underachievement is due to inaccurate exploration and inappropriate understanding of the subsurface. Traditionally, reservoir modeling is applied for exploring the subsurface. Recently, big-data analytics [4, 12] have been emerging as promising tools to improve the understanding of the subsurface. In this paper, we focus on formation evaluation [15], an essential step in understanding the subsurface by using the information commonly gathered during the drilling process. An appropriate formation evaluation will assist operators in making optimal decisions which in turn will dramatically reduce the costs.

Geology reports and well logs are most commonly used for formation evaluation. Well logs are collected during the drilling of a well. A well log records continuous sensor values along the depth of the formation rocks inside the wellbore. It often consists of a number of measurements, such as gamma radiation and resistivity, to detect and quantify oil and gas reserves. A geology report is of text in natural language issued by engineers analyzing the properties of sample rocks taken at depth intervals. The process of issuing the geology report is cumbersome. The drill bit cuts rocks into pieces and a mixture looking like mud is forced down the drill pipe to the bottom of the hole. The mud comes out of the drill bit and flows back to the surface. After pieces of rocks reach the surface with the mud, the engineer analyzes them carefully through a microscope or chemical test to see whether the rocks being drilled are sandstone, limestone, or shale, whether they have porosity, and whether they are bearing oil or gas inside.

Writing the geology report is quite subjective and its quality highly depends on the expertise of the engineer. Furthermore, the rock description is only based on a few rock samples arriving at the surface and might not cover all the information of rock samples in a certain depth interval. Also, compared to acquiring the well logs, recording rock descriptions requires much more time. To assist in issuing the geology report, we examine if a rock description can be automatically generated from multivariate well logs. In this paper, we propose an encoder-decoder-based model to solve this problem. The encoder extracts representative features of a fixed dimension

from multivariate well logs and the decoder generates rock descriptions.

There are two challenges in this task. First, how do we represent a rock description? A rock description is not a plain text but a well-structured text. It is composed of words in a number of rock properties. This is different from the plain text in image and video captioning [22] [21]. Second, how do we represent the information in different well logs? A traditional encoder processes a single image or a sequence of video frames. In our use case, the multivariate well logs are independent sensors, each of which is a 1-dimensional sequence. The representation for the well logs differs dramatically from that in the previous studies. Furthermore, well-log types and rock properties have inherent relations in nature. Different rock properties produce different values for different well-log types. It is natural to ask if these inherent relations can contribute to our task and how to leverage them.

To the best of our knowledge, this is the first work that formally defines the problem of generating text from multivariate sensors. This work is applied to a potential use case in the oil & gas industry. The main contributions of this paper are as follows:

- Due to the characteristics of rock descriptions, we design a hierarchical structure for the decoder to generate rock descriptions. We leverage the inherent relations between well-log types and rock properties by using an attention mechanism. This gives engineers an insight into which types of rock descriptions are likely to be affected by which types of sensors.
- To boost the performance of text generation, we design an enhanced hierarchical attention to identify typical patterns of a well log for a rock property. We also use external memory in the rock-property layer of the decoder to mitigate the memory decay along rock properties.

The remainder of this paper is organized as follows. In Section 2, we introduce the geology reports and well logs. In Section 3, we review previous works. In Section 4, we introduce recurrent neural network and convolutional neural network. In Section 5, we define our problem setting. We describe the proposed models in Section 6 and describe the data set in detail in Section 7. We present the experimental results on real shale well data in Section 8 and conclude this paper in Section 9.

2 PRELIMINARIES ON GEOLOGY REPORTS AND WELL LOGS

In this section, we give a brief overview of geology reports and well logs.

2.1 Geology Report

A sample geology report is shown in Figure 1. A geology report is composed of rock descriptions (right column) for certain depth intervals (left column) over the wellbore. Each depth interval covers about 30 to 50 feet. A rock description is composed of a list of phrases depicting the properties of the sample rocks, such as rock type (e.g., sandstone), rock color (e.g. light to medium tan), and shape (e.g., finely grained). Some phrases express the opinions of engineers with respect to those properties and indicate whether the depth interval covers an oil or gas bearing zone. It is often the case that writing a rock description needs to follow a guideline, in which the rock properties and their order are predefined.

15221 – 15250	SANDSTONE: 50% Light to medium tan, off white, cream, cream tan, light to medium brown, light gray, finely grained, subrounded to subangular, poorly sorted, heavily dolomitic, moderately cemented with dolomitic cement, 50% dolomite, light to medium gray, dark gray, cream gray, cream, micro to very fine crystalline, argillaceous, platy, hard, good intercrystalline porosity, very strong petroleum odor, light brown oil staining, faint dull yellow fluorescence, fair cloudy cut.
15251 – 15280	SANDSTONE: 60% Light to medium tan, cream, light to medium brown, light to dark gray, finely grained, well rounded to rounded, moderately sorted, heavily dolomitic, moderately cemented with dolomitic cement, 40% dolomite, light to medium gray, dark gray, cream gray, cream, micro to very fine crystalline, argillaceous, platy, hard, good intercrystalline porosity, very strong petroleum odor, light brown oil staining, faint dull yellow fluorescence, fair cloudy cut.

Figure 1: Example of a geology report

2.2 Well Log

A well log is a series of sensor data indicating rock properties. For example, the gamma ray log is recorded during a sonic shear test for detecting formation change and identifying the location of shale formation. The rate of penetration (rop) records the speed at which a drill bit breaks the rock under it to deepen the borehole. Well logs c1, c2, c3 and c4 measure hydrocarbons, which are methane, ethane, propane, and butane, respectively. The values of well logs are often measured at every 1 to 2 feet. Figure 2 shows an example of multiple well logs mainly used in this paper. The x-axis of this figure represents the measured depth, which is roughly equal to the length of the borehole.

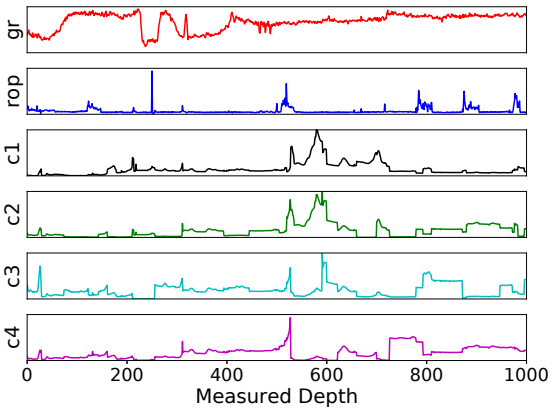


Figure 2: Example of multiple well logs.

3 RELATED WORK

Our study covers various research fields from sequence generation to attention mechanism.

Sequence generation is a general task used in image and video captioning [22] [21], machine translation [2], and document summarization [9]. Image and video captioning generates textual descriptions from images and videos. For image captioning, Karpathy et al. [5] associated detected objects in an image with relations in a dependency tree, and introduced

a structured max-margin objective function to learn their similarities. Based on this work, Karpathy et al. [6] used a recurrent neural network (RNN) to generate a plain image description by using visual-semantic alignments. Vinyals et al. [22] presented a generative model by maximizing the likelihood of the target-description sentence given training images. For video captioning, Venugopalan et al. [21] used a convolutional neural network (CNN) to extract features from video frames, and applied average pooling to these features, which are fed into a long short term memory (LSTM) to generate the caption. Krause et al. [7] used a hierarchical structure for generating descriptive image paragraphs. This model summarizes the features of detected objects in an image into a fixed size of representation. The hierarchical decoder generates multiple sentences for each object in the image. Machine translation is the task that translates a source sentence into a target sentence. Cho et al. [2] designed a framework with an RNN to encode the source sentence and decode the target sentence, which can be seen as a sequence-to-sequence model [18]. Document summarization is the task that generates succinct or conclusive text from a collection of input sentences. Li et al. [9] trained a hierarchical LSTM auto-encoder to preserve and reconstruct multi-sentence paragraphs.

A so-called alignment mechanism, called an attention, has been widely used in the research field of sequence generation. One popular use of an attention is in machine translation. A potential issue of Cho et al.'s model [2] is that all necessary information of a source sentence must be compressed into a fixed-length vector, resulting in the difficulty in coping with long sentences. Bahdanau et al. [1] extended this model by focusing on the most relevant information in the source sentence when decoding the target sentence. In image captioning, Xu et al. [24] incorporated an attention into their model. It allows the model to focus on the salient part of an image while generating its caption. In video captioning, Yao et al. [25] used the attention mechanism to boost the video captioning by exploiting the temporal structure embedded in the video. In document summarization, it has been proven [9] that an attention mechanism is effective in a hierarchical auto-encoder.

There are several variants of attention mechanism. Luong et al. [11] designed both global and local attentions for machine translation. While the global attention looks up every word in the source sentences, the local attention only looks up a subset of words in a source sentence. Graves et al. [3] proposed a neural Turing machine that uses an external memory associated with reading and writing operations. It supports both content-based addressing and location-based addressing.

4 PRELIMINARY ON RNN AND CNN

In this section, we give an overview of recurrent neural networks (RNN) and convolutional neural network (CNN). These two neural networks are fundamental components of our proposed framework.

4.1 Recurrent neural network

RNN is a neural network that operates on sequential data. Long short-term memory network (LSTM), one of the most popular variants of RNN, is capable of learning a longer dependency in a sequential data than RNN.

Given a sequence of inputs $X = \{x_1, x_2, \ldots, x_{T_x}\}$, a LSTM associates each time step with an input. We define the LSTM units at each time step t to be a collection of vectors in $\mathcal{R}^d$, which are an input gate $\mathbf{i}_t$, a forget gate $\mathbf{f}_t$, an output gate $\mathbf{o}_t$, a memory cell state $\mathbf{c}_t$, and a hidden state $\mathbf{h}_t$. σ denotes the sigmoid function. The hidden state $\mathbf{h}_t$ for each time step t is given by

$$\begin{bmatrix} \mathbf{i}_t \\ \mathbf{f}_t \\ \mathbf{o}_t \\ \tilde{\mathbf{c}}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} \mathbf{W} \begin{bmatrix} \mathbf{h}_{t-1} \\ x_t \end{bmatrix} \tag{1}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t \tag{2}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \tag{3}$$

where $\odot$ denotes elementwise multiplication, $x_t \in \mathcal{R}^d$ is the input at the current time step, and $\mathbf{W} \in \mathcal{R}^{4d \times 2d}$. At each time step t, the hidden state of the LSTM is updated by

$$\mathbf{h}_t = LSTM(\mathbf{h}_{t-1}, x_t, \Psi) \tag{4}$$

where Ψ represents all parameters of LSTM. A typical use of a LSTM is to learn a probability distribution over a sequence. To predict the next symbol in a sequence, it needs to calculate the conditional distribution $p(x_t|x_{t-1}, \ldots, x_1)$ for each possible symbol by using a softmax function.

4.2 Convolutional neural network

CNNs have been proven to extract high level features in local areas effectively. For feature extraction from image, 2-dimensional kernels with a specified size move over the image. By using such an operation in multiple layers, higher order patterns can be detected. However, for a temporal sequence, a 1-dimensional kernel is used in a temporal convolution. This utilization of CNN is also widely applied to the action recognition with wearable sensors [13].

5 PROBLEM DEFINITION

In this section, we formally define our problem setting. Suppose we have data from multivariate well logs and their corresponding rock description at the same depth interval. Our target is to generate a rock description from the multivariate well logs at a given depth interval. Mathematically, each well log is denoted by a fixed length of a sequence $\mathbf{x}^s = \{x_1^s, x_2^s, \ldots, x_L^s\}$, where $s \in \{1, 2, \ldots, N_s\}$ is the s-th well log among N_s types and L is the length of log values. We use x_l^s to denote the value of the s-th well log at the l-th location. The rock description is denoted by $\mathbf{Y} = \{\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^{N_p}\}$ where N_p is the number of rock properties. Each rock property $\mathbf{y}^p$ ($p \in \{1, 2, \ldots, N_p\}$) can be represented by $\{y_1^p, y_2^p, \ldots, y_{T_p}^p\}$

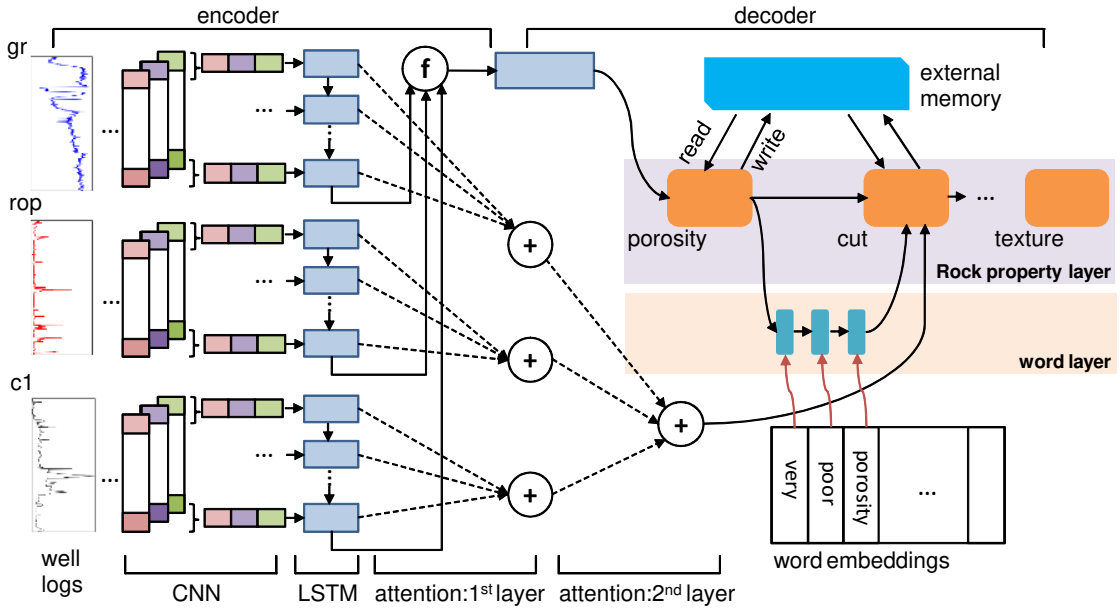


Figure 3: Architecture of proposed model.

where T_p is the number of words used in the p-th rock property. We use y_t^p to denote the t-th word of the p-th rock property.

6 PROPOSED MODEL

In this section, we introduce our model of generating rock descriptions from multivariate well logs. It adopts the typical encoder-decoder structure and is shown in Figure 3.

6.1 Encoder

In the encoder, a CNN with a 1-dimensional kernel is used to extract features for each well log. We use multiple layers of feature maps to convolute the data from raw-data level to abstract levels. The feature maps in the final layer contain features that cover the information of raw well logs along the depth. In the case of multiple well logs, the same convolution operation is applied to each well log while keeping the output of the convolution separated. It is because each well log has its own physical characteristics, and a convolution across different well logs might negatively affect the extraction of the typical patterns in a specific well log.

CNN features are generally not a good representation of series data. To represent the series aspect of well logs, we apply an LSTM to the features extracted by the CNN. For the s-th well log, we will have hidden states of the LSTM, denoted by $R^s = \{\mathbf{r}_1^s, \mathbf{r}_2^s, \ldots, \mathbf{r}_{L_s}^s\}$, where L_s is the number of hidden states. To simplify the symbol, we use $\mathbf{g}_s$ to represent the final hidden state of the LSTM for the s-th well log.

By applying CNNs and LSTMs to all well logs, we can obtain a set of the final hidden states, which is denoted by $G = \{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_{N_s}\}$. Finally, we use a fully-connected layer to transform the set of final hidden states into a vector representation of a fixed length, denoted by $\mathbf{e}_S$. It is also used as the input for the decoder.

6.2 Hierarchical Decoder

An rock description at a certain depth interval has an inherent hierarchical structure. Generally, a rock description contains phrases from a limited number of rock properties. The rock properties used in a rock description are usually rock type, rock color, texture, composition, hardness, etc. For a given rock property, engineers use a limited number of words to explain this rock property. Due to this structure of a rock description, we use a hierarchical structure for the decoder. For clarification, we use the following notations: $\mathbf{h}_{t-1}$ and $\mathbf{h}_{t-1}^w$ denote hidden states at step $t-1$ of the LSTM at the rock-property and word levels, respectively. l_{t-1} denotes the input of the rock-property LSTM at step $t-1$, and $e_{y_{t-1}}$ is the word embedding associated with word y_{t-1}.

The decoding operates on a hierarchical structure with two layers of LSTMs. The output of the LSTM at the rock-property level at step t is given by

$$\mathbf{h}_t = LSTM(\mathbf{h}_{t-1}, l_{t-1}, \Theta) \tag{5}$$

where the first argument of the LSTM is called a hidden state, the second argument is an input, and Θ are parameters. At the initial step, $\mathbf{h}_0 = \mathbf{e}_S$ and l_0 is initialized as a zero vector. We denote the set of hidden states of the LSTM at the rock-property level as $H = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_{N_p}\}$. The hidden state at step t outputted using the LSTM at the rock-property level is used as the input of the LSTM at the word level to predict the words of the t-th rock property. The procedure for predicting words can be performed using a word-level LSTM:

$$\mathbf{h}_t^w = LSTM(\mathbf{h}_{t-1}^w, e_{y_{t-1}}, \Theta). \tag{6}$$

At the initial step, $\mathbf{h}_0^w$ is initialized by the hidden state outputted using the LSTM at the rock-property level. At both training and test phases, e_{y_0} is initialized as the word embedding of the starting symbol. This LSTM predicts words in a specific rock property. For other $e_{y_{t-1}}$, in the training phase, it is the word embedding of the ground-truth word y_{t-1}, while in the test phase, it is the word embedding of the word predicted at the previous step. For the t-th rock property, the final state $\mathbf{h}_{T_p}^w$ of the LSTM at the word level is used as input l_0 of the LSTM for the next rock property.

6.3 Modeling Attention

Well log types and rock properties have inherent relations in nature. We leverage these inherent relations in our model by designing two forms of attention. It allows engineers to have more insight into the rock descriptions and well logs. Unlike most previous studies that apply an attention to most granular units, such as words, our model performs the attention at abstract yet meaningful levels, which are rock property, well-log type, and subsequence of raw well logs.

The first form of attention does the alignment between rock properties and well-log types. This attention can determine which rock property is sensitive to which type of well log. The second form of attention does the alignment at two levels. The first level is between the rock property and hidden states of the LSTM for each well log, and the second level is between well-log types and rock properties. This form of attention can further determine which subsequence of well logs contributes the most to the description in a rock property.

6.3.1 Attention at rock-property level . Inspired by the work by Li et al. [9], this attention aligns the final hidden states of the LSTM in each well log and a hidden state of the LSTM at the rock-property level. The alignment is done as follows. Suppose that $G = \{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_{N_s}\}$ is the set of the final hidden states of LSTMs for the well logs and $\mathbf{h}_{t-1}$ is the hidden state of the LSTM at the rock-property level at the previous step $t-1$. The similarities between $\mathbf{h}_{t-1}$ and $\mathbf{g}_s$ ($s \in \{1, 2, \ldots, N_s\}$) are calculated by

$$v_s = \mathbf{v}^T \tanh(W_1 \cdot \mathbf{h}_{t-1} + W_2 \cdot \mathbf{g}_s) \tag{7}$$

where $W_1, W_2 \in \mathcal{R}^{d \times d}$, $\mathbf{v} \in \mathcal{R}^d$. Then v_s is normalized as

$$\alpha_s = \frac{\exp(v_s)}{\sum_{s'} \exp(v_{s'})} \tag{8}$$

where α_s is the weight between the $t-1$-th rock property and the s-th well log. It can be seen as a metric for measuring the importance of their relation. By weighted averaging over all types of well logs, a context vector $\mathbf{m}_t$ is then calculated by

$$\mathbf{m}_t = \sum_{s \in [1, N_s]} \alpha_s \mathbf{g}_s \tag{9}$$

Then, the hidden state $\mathbf{h}_t$ of the LSTM at the rock-property level can be obtained by combing $\mathbf{h}_{t-1}$, $e_{y_{t-1}}$, and $\mathbf{m}_t$:

$$\begin{bmatrix} \mathbf{i}_t \\ \mathbf{f}_t \\ \mathbf{o}_t \\ \tilde{\mathbf{c}}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} \mathbf{W} \begin{bmatrix} \mathbf{h}_{t-1} \\ e_{y_{t-1}} \\ \mathbf{m}_t \end{bmatrix} \tag{10}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t \tag{11}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \tag{12}$$

where $\mathbf{W} \in \mathcal{R}^{4d \times 3d}$. Then $\mathbf{h}_t$ is used as the input of the word-level LSTM.

6.3.2 Attention at rock-property and word levels . The first form of attention discloses the relations between well-log types and rock properties, showing the degrees of importance of rock properties to different well logs. However, only the final hidden states of LSTMs for the well logs are used. Possible important information in the other hidden states of the LSTMs might be ignored. Remember that a rock description is written based on rock samples collected at a depth interval, but the location of the collected rock samples is unclear. A subsequence of a well log, whose location is consistent with where rock samples are collected, is more important for generating rock descriptions than other subsequences.

Motivated by the above reason, we design a two-layer attention, which can identify which certain rock-property is more sensitive to which specific subsequence of sensor values from which type of well log. As shown in Figure 3, the first layer is to retrieve the information from relevant subsequences of a well log. We achieve this by aligning a hidden state $\mathbf{h}_p$ ($p \in \{1, 2, \ldots, N_p\}$) of the LSTM at the rock property level with with the hidden states of the LSTM that models the change in the well log values over depth intervals. Mathematically, for the s-th well log, we align $\mathbf{h}_p$ with $R^s = \{\mathbf{r}_1^s, \mathbf{r}_2^s, \ldots, \mathbf{r}_{L_s}^s\}$. By using Equations (7), (8) and (9), we can obtain a context vector $\mathbf{p}_s$. For all well logs, we have a set of context vectors $P = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{N_s}\}$. In the second layer, $\mathbf{h}_p$ is aligned with P to derive $\hat{\mathbf{m}}_t$ replacing $\mathbf{m}_t$ as input in Eq. (10).

6.4 External Memory

External memory has been proven to be effective in deep learning tasks, including reasoning [3] and machine translation [23]. By applying an external memory to the LSTM at the rock-property level, it mitigates the information decay in the LSTM. Since information of the LSTM at the word level is only received from the LSTMs at the rock-property level, maintaining the information along rock properties is essential for generating rock descriptions.

The external memory has reading and writing operations to retrieve and update information. Formally, we define the memory at step $t-1$ as $\mathbf{M}_{t-1} \in \mathcal{R}^{k \times m}$, where k is the number of memory cell and m is the dimension of each memory cell. Before using Eq. (5), we use $\mathbf{h}_{t-1}$ as a query to retrieve the information from $\mathbf{M}_{t-1}$. This reading operation is expressed as

$$\hat{\mathbf{h}}_{t-1} = read(\mathbf{h}_{t-1}, \mathbf{M}_{t-1}) \tag{13}$$

Table 1: Ontologies defined for rock properties. Phrase examples are listed below.

Type	Phrases
Rock Type	sandstone, mudstone, limestone, dolomite, cement, shale, claystone, anhydrite
Rock Color	light gray, cream, tan, off white, light gray, dark gray, light tan, cream tan, gray brown
Texture	fine microcrystalline, trace microcrystalline, very fine crystalline, fine crystalline, intercrystalline
Composition	slightly sandy, very sandy, trace sandy, very silty, moderately silty, fragmental, commonly fragmental
Hardness	very hard, moderately hard, trace hard, slightly friable, rare friable, very soft, predominantly soft
Grain Shape	angular, subangular, subrounded to subangular, blocky, angular blocky, subrounded, well rounded
Mineral	halite, scattered halite, trace halite, disseminated pyrite, trace pyrite, well calcite, slight calcite
Porosity	porosity, intergranular porosity, poor intergranular porosity, no visible porosity, fair porosity
Hydrocarbon	oil stain, even oil stain, no visible oil stain, spotty stain, brown oil stain, trace oil, even brown oil stain
Cut	no cut, milky cut, cloudy cut, streaming cut, instant cloudy cut, slow streaming, cloudy yellow cut
Fluorescence	no fluorescence, dull yellow fluorescence, even yellow green fluorescence, scattered yellow fluorescence

After this reading operation, $\hat{\mathbf{h}}_{t-1}$ replaces $\mathbf{h}_{t-1}$ as the input of Eq. (5). After we obtain $\mathbf{h}_t$, we update the information to the memory. The writing operation is expressed as

$$\mathbf{M}_t = write(\mathbf{h}_t, \mathbf{M}_{t-1}) \tag{14}$$

where $\mathbf{M}_t$ is the memory state at the step t. Due to the limited space, we do not elaborate on the reading and writing operations, which we refer from the work by Wang et al. [23].

7 DATASET

In this section, we give the details of the required preprocessing of geology reports and well logs.

7.1 Data preprocessing

We study the geology reports and well logs of shale wells in North Dakota in the United States, which are publicly available at the government site[1]. The geology reports are saved as scanned PDF files. To extract the depth intervals and rock descriptions from a scanned PDF file, we used the techniques of optical character recognition (OCR) and page layout analysis [19]. We parsed the scanned PDF files into a list of pairs, each of which consists of a depth interval and rock description.

For well logs, most of the wells have multiple types of sensor data in the Log ASCII Standard (LAS) format, and different wells have different sets of sensor types. In this study, we focus on six types of sensors, which are gamma rays, rate of penetration (rop), c1, c2, c3, and c4, since most wells use these types of sensors. In accordance with the LAS specifications, we parsed the six types of sensors into a list of tuples, each of which consists of a measured depth and sensor values.

We made a statistic on the frequency of the depth intervals used in the geology reports. According to our analysis, more than 60% of the depth intervals is 30 feet which dominates the size of depth intervals. We used this depth interval to collect rock descriptions and well logs.

7.2 Geology reports

After investigating the geology reports, we found that phrases in the rock description are not plain text. A number of specific rock properties are often used. A set of keywords is often associated with a specific property. For example, in phrases about texture, words such as intercrystalline, granular, microcrystalline, and crystalline are frequently used. Due to this fact, we define the most frequently used properties as ontology types. In accordance with the work by Tong et al. [20], the definition of ontology types and phrase examples are listed in Table 1.

In a rock description, some ontology types correspond to only one phrase, while others may correspond to multiple phrases. We also observe a regulation in the composition of specific phrases. Such a phrase can be decomposed into a descriptor followed by a keyword. The descriptor often depicts a degree or status. For example, words such as 'porosity', 'intergranular', 'poor intergranular', 'no visible', and 'fair' show different degrees of porosity. Words such as 'cut', 'streaming', 'instant cloudy', 'slow streaming' show different statuses of bubbles that come out from the rock.

Table 2: Statistics on rock description.

#Phrase	#word$_{all}$	#word$_{cor}$	#word$_{use}$	Ratio
763588	4250	1998	443	0.9437

We made statistics on the phrases and words in the rock descriptions. As shown in Table 2, among 763588 different phrases, there are 4250 unique words. By removing the words that are not correctly recognized through OCR and whose frequency is lower than 5, we used 443 words and replaced the other words with the word 'unknown (unk)' in our experiment. These 443 words occupy 94.37% frequency of all words. Therefore, even a small number of the selected words do not result in much information loss on the data. In addition, a percentage sometimes precedes a phrase, such as 50% porosity. Due to much variance in the numbers, the numbers are replaced with 'unk'.

8 EXPERIMENT

In this section, we explain the parameter configuration, discuss the experimental results, and discuss an in-depth analysis on the two forms of attention and external memory.

8.1 Parameter settings

We set the size of the convolution window to 5 and designed four convolution layers for each well log. The filter sizes in

[1] http://www.dmr.nd.gov/oilgas

Table 3: Performances of generating rock descriptions with models in different configurations

Method	Bleu-1	Bleu-2	Bleu-3	Bleu-4	Meteor	Rouge-1	Rouge-2
flat decoder	87.10	69.16	50.29	34.48	74.63	74.37	39.90
flat decoder + $attn_w$	88.27	70.16	51.13	35.13	75.09	74.94	40.58
hierachical decoder	89.96	75.13	57.58	41.63	79.98	79.95	54.10
hierachical decoder + $attn_r$	90.43	75.56	57.97	41.96	80.22	80.12	54.43
hierachical decoder + $attn_{l2}$	90.16	75.51	58.14	42.30	80.25	80.13	54.65
hierachical decoder + $attn_r$ + extm	90.37	76.08	59.10	43.51	80.64	80.52	55.58
hierachical decoder + $attn_{l2}$ + extm	**91.13**	**76.84**	**59.85**	**44.21**	**81.00**	**80.92**	**56.19**

the convolution layers for each well log were set to 5×1. The number of filters for each convolution was set to 256, 256, 128, and 128. The filter at the same layers was shared across different well logs, which is a typical approach to reduce computational cost.

In both the encoder and decoder, we used a simpler yet effective variant of an LSTM, called gated recurrent unit (GRU). The dimension of the state in the LSTM for well logs was set to 128. A fully connected layer was used to summarize the last states of the LSTM for multiple well logs. The input of the fully connected layer is a feature vector, which is the concatenation of the last states of the LSTMs. We used 512 neurons in the fully connected layer. The dimension of the state in the LSTM for the decoder was set to 512, and that of the word-embedding was set to 512 as well. The word embedding lookup table was initialized by sampling from the normal distribution of mean 0 and variance σ^2. The σ is $d^{-\frac{1}{2}}$ where d is the dimension of the word embedding. As a default setting, the activation functions used in the convolution layers and fully connected layer were set to sigmoid. The dropout rate was set to 0.7. For the external memory, the number of memory cells was set to 5, 10, 50 or 100, and the dimension of the memory cell was set to 64, 128, 256, 512 or 1204. Each memory cell was initialized by $\tanh(\mathbf{W}^e \sum_{i=0}^{N_s} \mathbf{l}_i)$ with a Gaussian noise, where $\mathbf{l}_i$ is the mean of all LSTM hidden states in the i-th well log and $\mathbf{W}^e$ is a weighting matrix.

Adam was used as the gradient descent algorithm, and the initial learning rate was set to 0.001. The L2 regularization was used to avoid the over-fitting of the training model, and the balancing parameter for the L2 norm was set to 10^{-5}, 10^{-6} or 10^{-7}. The batch size of samples was set to 64. We found empirically that the choice of parameters mentioned above tends to produce the best performances.

8.2 Experiment Results

Rock descriptions do not necessarily contain phrases from all ontology types and well logs might have missing sensor types. To avoid the side-effect of missing values, we filtered out the data if both of the following criteria are satisfied. First, a rock description has phrases from less than 50% of the ontology types. Second, the number of sensor types that have missing values is larger than 50% of all sensor types. By using the filter, we collected 77803 pairs of data, each of which includes a rock description and its well logs at a depth interval. Due to the batch size, we had a total of 62272 pairs of data for training and 7744 pairs of data for validation and test.

To evaluate the performance of our models, we use three metrics: Bleu [14], Meteor [8], and Rouge [10]. These three metrics are widely used in image and video captioning and machine translation. Bleu is a precision-oriented measure that uses n-gram precision between candidate text and reference text. We used Bleu-1, Bleu-2, Bleu-3, and Bleu-4 in this experiment. Meteor is a metric based on the harmonic mean of unigram precision and recall, with recall weighted 9 times higher than precision. Rouge is a recall-oriented measure that calculates n-gram recall between candidate text and reference text. We used Rouge-1 and Rouge-2 in this experiment. For each of these three metrics, the larger the value of a metric, the better the performance of text generation.

We compare the performance of our proposed models with a number of baselines. We keep the encoder of the models the same, and change configurations of the decoder. The term *flat decoder* denotes the decoder that has only the word-level LSTM. $attn_w$ denotes the alignment between each word and the last states of the LSTM for the well logs. *hierarchical decoder* denotes the decoder that has an LSTM at both the rock-property and word levels. $attn_r$ denotes the alignment between rock properties and the last states of the LSTM for the well logs. $attn_{l2}$ denotes the alignment at both two layers. *extm* denotes the use of external memory for the LSTM at the rock-property level. Table 3 shows the performance of the models in different configurations. The performances with the hierarchical decoder are significantly better than those with the flat decoder. The bleu-4 score of *hierarchical decoder* is 7.15 points higher than that of *flat decoder*. Using attention for both *hierarchical decoder* and *flat decoder* slightly improves their bleu scores. The improvement in $attn_r$ and $attn_{l2}$ with the hierarchical decoder shows that the inherent relations between well-log types and rock properties are effective. Using two layers of attention is better than the one layer. The bleu-4 score of $attn_{l2}$ with the hierarchical decoder is 0.34 points higher than that of $attn_r$. Using the external memory does improve the performances of the models for the two forms of attention. It improves 1.55 and 1.91 points for *hierarchical decoder+$attn_r$* and *hierarchical decoder+$attn_{l2}$*, respectively. Table 4 shows examples of generated rock descriptions. Due to limited space, we only show a fair and bad case. The ground truth are the descriptions after replacing the 'unk' words. We can see that a generated description with a fair quality covers most information in the ground truth. In the base case, even if the generated text is semantically different from the ground truth, it is readable text

Table 4: Examples of generated rock descriptions

quality	ground truth	generated descriptions
fair	sandstone, trace off white to cream, medium to dark gray, friable to firm, very fine grained to fine grained, subangular to subrounded, trace <unk> very trace nodular pyrite, possible intergranular porosity, trace brown spotty oil stain	sandstone, occasional off white to cream, medium to dark gray, friable to firm, very fine grained to fine grained, subangular to subrounded, trace <unk> very trace nodular pyrite, possible intergranular porosity, trace brown spotty oil stain
bad	dolomite, light brown moderate brown, trace light brown staining, microcrystalline framework, silty, scattered <unk> pyritic specks, soft to scattered firm, round to <unk> to subblocky, tight pinpoint porosity, good streaming cut	sandstone, black to dark gray <unk>, dark brown in part, argillaceous, firm to hard, subangular, sub rounded inch part, fine grained, slightly fair to good intergranular porosity, fast broad streaming yellow green instant cloudy cut

that mentions almost the same rock properties as the ground truth.

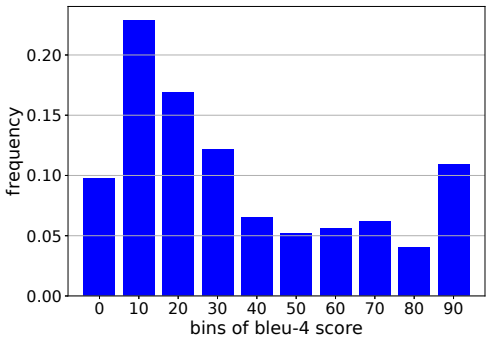


Figure 4: Distribution of bleu-4 scores

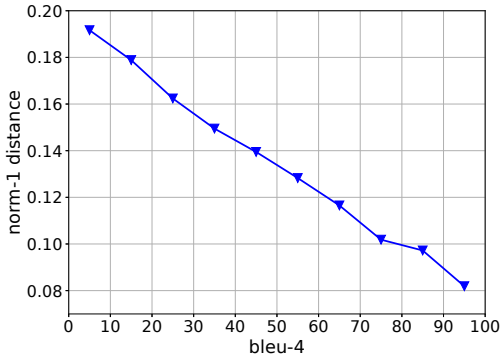


Figure 5: Distance change over bleu-4 score

We plot the distribution of bleu-4 scores in Figure 4. We divided the range of bleu-4 scores, $[0, 100]$, into ten bins evenly. For example, the bin named 10 represents the bin of bleu-4 scores falling in the range $(10, 20]$. We can observe from Figure 4 that more than 40% of test samples have bleu-4 scores in the range $(10, 30]$. Surprisingly, more than 10% of test samples have bleu-4 scores larger than 90.

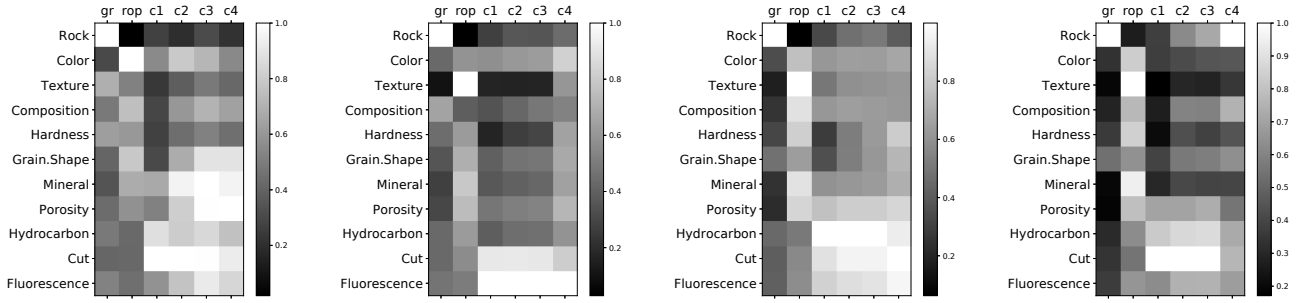
We further study the relationship between attention matrix and bleu-4 score. We generated an attention matrix for each test sample in two different configurations. In the first configuration, we used the ground truth of test samples without updating learned parameters, while in the second configuration, we carried out real text generation. We calculated the distance between two attention matrices obtained from the same test sample in different configurations. We averaged the distances of the test samples whose bleu-4 scores fall into each of the evenly-divided 10 bins. Figure 5 shows that the norm-1 distance decreases as bleu-4 scores increase, meaning the samples with higher bleu-4 scores have less difference between attention matrices in the two configurations. This implicitly shows that our model appropriately learns the attention from the training data.

8.3 Analysis of attention

In this subsection, we discuss an in-depth analysis of $attn_r$ and $attn_{l2}$, which shows the relations between well-log types and rock properties from the data-driven perspective.

First, we examine the relation matrix for each rock type in the training data. We collected the four most frequent rock types, namely, limestone, sandstone, dolomite, and shale. For each rock type, we collected relation matrices in the training phrase, averaged them and normalized the averaged matrix column-wise. Figure 6 shows the relation matrix for each rock type. In the four types of rocks, the rock properties, such as cut and hydrocarbon, are more important than others for well-log types, such as c1, c2, c3, and c4. This might be because these well logs directly measure methane and other hydrocarbon gases. Cut is the rock property in which a chemical test is done to determine if hydrocarbons are present in the rock. It can also be seen that the property 'rock' is strongly related to the well-log type 'gr', and the property 'texture' is important for the well-log type 'rop'.

Based on the relation matrices, we examine the rankings of sensitive rock properties with respect to each well-log type in all rock types. It can be seen from Table 5 that the well logs, such as c1, c2, c3, and c4, are sensitive to the rock properties, such as hydrocarbon, porosity, cut, and fluorescence. This might be because these rock properties are highly related to appearance of oil and gas. We can also see that texture, mineral, shape, and hardness are sensitive rock properties



(a) Limestone (b) Sandstone (c) Dolomite (d) Shale

Figure 6: Visualization of relationship between rock properties and well log types.

Table 5: Rankings of sensitive rock properties: hydrocarbon(Hydro), fluorescence(Fluo) and composition(Comp).

well log	1	2	3	4	5
gr	Rock	Comp	Fluo	Cut	Hydro
rop	Texture	Mineral	Shape	Hardness	Color
c1	Cut	Fluo	Hydro	Mineral	Porosity
c2	Cut	Fluo	Porosity	Hydro	Mineral
c3	Cut	Fluo	Porosity	Hydro	Mineral
c4	Fluo	Porosity	Cut	Mineral	Hydro

with respect to rop. These rock properties in physical aspect might determine the drill speed.

Second, we conduct an analysis on the subsequences of a well log derived from the first layer of $attn_{l2}$. The subsequence is obtained by calculating the similarity between each state of LSTM for each well log and hidden states of the LSTM at the rock-property level. This subsequence can be regarded as information that mostly contributes to the description generation for a rock property. In this experiment, we consider two rock properties, porosity and cut. As explained in Section 7.2, phrases in these two rock properties often depict a degree or a status. Tong et al. [20] verified that the degree and status can be seen as a kind of sentiment. We collected frequent phrases of these two rock properties for each category of sentiment. The two frequent positive phrases we collected are 'good intergranular porosity' and 'good intercrystalline porosity', two frequent neutral phrases are 'fair intergranular porosity' and 'fair intercrystalline porosity', and two frequent negative phrases are 'poor intergranular porosity' and 'rare intergranular porosity'.

Because different operators may use different devices to record well logs, the values of well logs may have different scales. Thus, instead of comparing the mean of subsequences, we focus on the change in values, such as standard variance. Table 6 shows the standard variance of each well log in different sentiments. In both porosity and cut, the values of c1 and c4 in the positive phrases have smaller variance than those of neutral and negative phrases. The values of c2 in the positive phrases have larger variance than those of neutral and negative phrases. The values of rop in the neutral phrases have larger variance than those of positive and negative phrases.

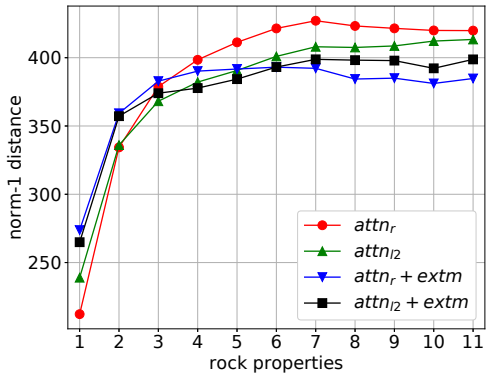


Figure 7: Information change over LSTM

8.4 Analysis of external memory

First, we conduct an analysis on the information decay in the LSTM at the rock-property level in different settings. We calculated the norm-1 distance between the fixed length of representation $\mathbf{e}_S$ in the encoder and the hidden states of LSTM for each rock property. Figure 7 shows the change in the distance over rock properties, whose order is the same as that in Table 1. Their indexes are labeled starting from 1. We can see that, compared with $attn_r$, $attn_r + extm$ has less difference from the 4-th rock property. $attn_{l2}$ and $attn_{l2} + extm$ have the similar trend. It means that the external memory is helpful for mitigating the information decay. $attn_{l2}$ has a larger difference than $attn_r$ at the first rock property. It is partly due to the fact that, compared with $attn_r$, $attn_{l2}$ uses different data sources that are used to derive $\mathbf{e}_S$.

Second, we conduct an analysis on how parameter settings of the external memory affect performance. We tuned the parameters for $attn_{l2} + extm$ with the hierarchical decoder. The number of memory cells was tuned in the set of values $\{5, 10, 50, 100\}$ and the dimension of memory cells was tuned in the set of values $\{64, 128, 256, 512, 1024\}$.

Figure 8 shows that $attn_{l2} + extm$ with different numbers of memory cells and the dimensions of memory cells does not have drastic fluctuations in performance, which mainly falls

Table 6: Standard deviation of attended subsequences

Sentiment	Porosity						Cut					
	gr	rop	c1	c2	c3	c4	gr	rop	c1	c2	c3	c4
Good	3.27	2.78	29.25	12.08	2.20	8.36	4.97	0.97	7.85	38.87	10.03	0.31
Neutral	4.41	6.33	29.53	8.56	4.59	94.03	3.93	15.63	22.99	10.14	3.04	5.00
Negative	4.01	0.28	232.16	11.97	54.87	43.32	5.45	1.10	11.22	10.34	5.53	6.11

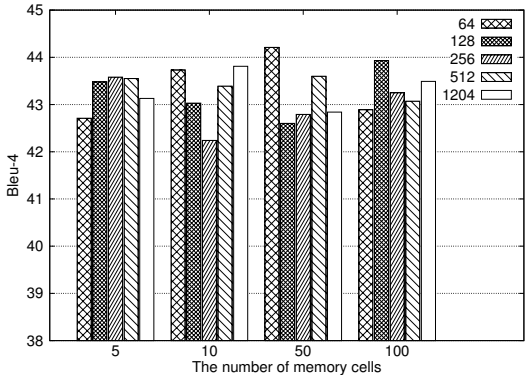


Figure 8: Performance change in different parameters of external memory

in the range between 42 and 44. It can also be seen that the performance with the dimension of memory cells, such as 64, increases when the number of memory cells increases from 5 to 50. Given the number of memory cells, the dimension of the memory cells, such as 512, seems not be able to achieve the best performances compared with other dimensions.

9 CONCLUSIONS

In this paper, we presented the problem in oil & gas industry of generating rock descriptions in human-readable text from multiple well logs automatically. Our approach can reduce the cost and time required by engineers to write such reports. To the best of our knowledge, this is the first work to methodically study this problem. Our experiments conducted on real well data from North Dakota in the United States give significant insight into this problem. The experiment results show that our models with a hierarchical decoder and techniques such as the two forms of attention and external memory are effective in generating rock descriptions.

For future work, we would like to explore three possible paths. First, we will try different addressing methods to read and write the external memory, and determine which kind of addressing method is most suitable for this task. Second, the metric used for model optimization is different from the metric used for evaluating the quality of the generated text. We would like to use reinforcement learning to directly optimize the metrics of interest [16]. Third, even the metrics used in this paper are widely applied in machine translation and image captioning tasks, it can not evaluate the quality of generated text thoroughly. Therefore, how to design a metric for our specific domain is another interesting work.

REFERENCES

[1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.
[2] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP*, pages 1724–1734, 2014.
[3] A. Graves, G. Wayne, and I. Danihelka. Neural turing machines. *arXiv:1410.5401*, 2014.
[4] K. Holdaway. *Harness Oil and Gas Big Data with Analytics: Optimize Exploration and Production with Data Driven Models*. Wiley, 2014.
[5] A. Karpathy, A. Joulin, and F. Li. Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*, pages 1889–1897, 2014.
[6] A. Karpathy and F. Li. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, pages 3128–3137, 2015.
[7] J. Krause, J. Johnson, R. Krishna, and F. Li. A hierarchical approach for generating descriptive image paragraphs. In *arXiv:1611.06607*, 2016.
[8] A. Lavie and A. Agarwal. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *StatMT*, pages 228–231, 2007.
[9] J. Li, M. Luong, and D. Jurafsky. A hierarchical neural autoencoder for paragraphs and documents. In *ACL*, pages 1106–1115, 2015.
[10] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *ACL Workshop*, pages 74–81, 2004.
[11] M. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, pages 1412–1421, 2015.
[12] S. D. Mohaghegh. Reservoir simulation and modeling based on pattern recognition. In *SPE*, 2011.
[13] F. J. O. Morales and D. Roggen. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1):115, 2016.
[14] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.
[15] S. Pirson. *Handbook of Well Log Analysis for Oil and Gas Formation Evaluation*. Prentice-Hall, 1963.
[16] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba. Sequence level training with recurrent neural networks. In *ICLR*, 2016.
[17] A. Scollard. Petrel shale-engineered for exploiting shale resources. In *URTec*, 2014.
[18] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.
[19] B. Tong, M. Klinkigt, M. Iwayama, Y. Kobayashi, A. Sahu, and R. Vennelakanti. Deep match between geology reports and well logs using spatial information. In *CIKM*, pages 1833–1842, 2016.
[20] B. Tong, H. Ozaki, M. Iwayama, Y. Kobayashi, A. Sahu, and R. Vennelakanti. Production Estimation for Shale Wells with Sentiment-Based Features from Geology Reports. In *ICDM Workshop*, pages 1310–1317, 2015.
[21] S. Venugopalan, M. Rohrbach, J. Donahue, R. J. Mooney, T. Darrell, and K. Saenko. Sequence to sequence - video to text. In *ICCV*, pages 4534–4542, 2015.
[22] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, pages 3156–3164, 2015.
[23] M. Wang, Z. Lu, H. Li, and Q. Liu. Memory-enhanced decoder for neural machine translation. In *EMNLP*, pages 278–286, 2016.
[24] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, pages 2048–2057, 2015.
[25] L. Yao, A. Torabi, K. Cho, N. Ballas, C. J. Pal, H. Larochelle, and A. C. Courville. Describing videos by exploiting temporal structure. In *ICCV*, pages 4507–4515, 2015.