# Encoder Decoder based Nepali News Headline Generation

Kaushal Raj Mishra
Electronics and Communication Engineer
Institute of Engineering
Tribhuwan University

Jayshree Rathi
Electronics and Communication Engineer
Institute of Engineering
Tribhuwan University

Janardan Banjara
Electronics and Communication Engineer
Institute of Engineering
Tribhuwan University

## ABSTRACT

In this paper, a method for Nepali News Headline Generation is presented. The proposed method uses GRUs, in an encoder-decoder fashion, taking the news content as input and generating a headline as the output. The news is converted into word tokens, which are vectorized using FastText, trained on a corpus of Nepali news articles and headlines collected from several web portals. The headline generation model is also trained on the same corpus. A sequence to sequence model, with an encoder and a decoder GRU is used as the generation model. The model was able to attain a BLEU score of 22.1 on the test set.

### Keywords

Recurrent Neural Network, Gated Recurrent Unit, FastText, Bilingual Evaluation Understudy (BLEU), Encoder, Decoder

## 1. INTRODUCTION

News headlines provide a short description of the news articles and are used to effectively capture the essence of an article. It can be considered a type of text summarization problem. With the increasing amount of textual data in the form of news articles, blog posts and social media headline generation becomes a very important task in Natural Language Processing. After having obtained the short descriptions for text as headlines, additional analysis can be done only for titles like sentiment analysis, topic modeling and document classification.

Most existing headline generation and text summarization approaches are extractive [1] and abstractive [13]. Extractive summarization selects a few important sentences from a given document and reorders them into a summary. Abstractive approaches involve generating summary, using words and sentences not originally present in the document. In this paper, we propose an abstractive approach using sequence to sequence encoder decoder model using RNN. GRUs [6] are used for their capability of learning long term dependencies. The news articles and headlines are represented in vector space using FastText [3] Word embedding.

## 2. RELATED WORK

Headline Generation and text summarization has been practised widely in other language. In early days, [1] mostly extractive techniques were for text summarization. However, in recent days various [13] abstractive techniques are being proposed for text summarization and headline generation.

In Automated natural language headline generation using discriminative machine learning models [8] a discriminative learning framework and a rich feature set for the headline generation task is presented for English language along with a novel [14] BLEU measure based scheme for evaluation of headline generation models, which does not require human produced references.

Generating News Headlines with Recurrent Neural Networks [10] presents an application of an encoder-decoder recurrent neural network with LSTM units and attention to generating headlines from the text of news articles for English Language. It is found that the model is quite effective at concisely paraphrasing news articles.

In addition to this, Automatic Text summarization has been researched for a long time. Some methods take a single document as input, but are fed other documents as well during training [11] [2]. Other methods involve aggregating similar documents from various sources and using them together, referred to as Multi-Document summarization [12].

Hindi language is similar to Nepali language and they both have the same origin: Sanskrit [9]. For Hindi, several text summarization models have been proposed. Both abstractive and extractive methods have been researched [4][7].

## 3. GATED RECURRENT UNIT

A gated recurrent unit (GRU) [6] is an enhancement of recurrent neural networks that uses gates to enhance its performance over longer sequences than vanilla Recurrent Neural Networks. Gated recurrent units help tackle the vanishing gradient problem that is a common issue with recurrent neural networks, by using a gating mechanism. Gated recurrent units [5] have 2 gates: an update gate and a reset gate. Using these two gates, the GRU generates outputs by selecting and controlling which information to pass through the model. This helps to retain information over a longer sequences. The output of a GRU is given by the following set of equations.

$$z_t = \sigma(W_{xz}x_t + U_{hz}h_{t-1} + b_z) \tag{1}$$

$$r_t = \sigma(W_{xr}x_t + U_{hr}h_{t-1} + b_r) \tag{2}$$

$$H_t = tanh(W_{xh}x_t + U_{rh}([r_t \otimes h_{t-1}) + b_h) \tag{3}$$

$$z_t = (1 - z_t) \otimes h_{t-1} + z_t \otimes H_t \tag{4}$$

where $\sigma$ is sigmoid function. $\otimes$ is an the element-wise multiplication operator. $z_t$, $r_t$, $H_t$ are the update gate, reset gate and candidate-activation function respectively. W, U and b are related parameter matrices.

## 4. ENCODER DECODER ARCHITECTURE

Encoder Decoder model is used for sequence to sequence[16] learning tasks. The Encoder can convert the input sequence into a single vector called the latent vector, or it can pass its internal states directly to the decoder. The decoder then converts the input from the encoder to an output sequence. These models are trained together. The goal is to maximize conditional probability of the target sequence given the input sequence. In general, multiple RNN-style cells are stacked together to form the encoder. RNNs are used for their capability to learn temporal dependencies in the input.

In practice, training RNN becomes difficult due to the problems of vanishing gradient (where the gradient during backpropagation becomes 0) and exploding gradient (where the gradient during backprogation becomes very large). To tackle this, Bidirectional GRUs [15] are implemented in this paper, which are an improved version of simple RNNs. The encoder GRUs can be stacked on top of each other. In this paper, multiple encoder GRUs have been stacked as shown in figure 1. The first layer takes words vectors as input. After all the inputs are read by encoder model, the final hidden states of the final encoder GRU [6] is a dense representation of the input sequence.

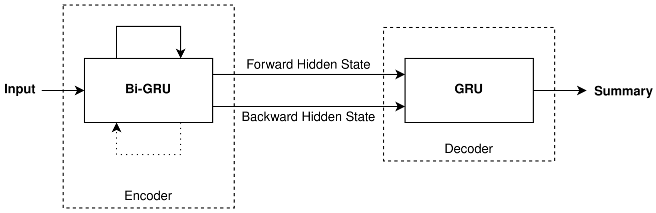

Fig. 1. Encoder-Decoder Architecture

## 5. BLEU SCORE

The Bilingual Evaluation Understudy Score (BLEU) [14] is an automatic sentence evaluation metric. It is used to evaluate a sentence generated by some procedure to a given reference sentence. When the sentences match exactly, BLEU gives a score of 100, whereas a perfect mismatch yields a score of 0. Originally, it was developed for evaluation in automatic machine translation systems. The scores are calculated by generating n-grams from the generated sentence and comparing the n-grams to the reference sentence.

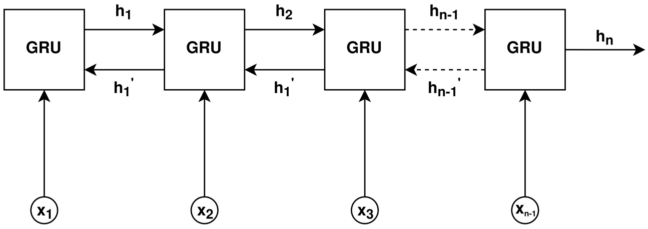

Fig. 2. Encoder Architecture

## 6. EXPERIMENTAL SETUP

A novel corpus of Nepali news is created specifically for the research. Nepali news articles and headlines are scraped from different Nepali news web portals. The scraped Nepali news Corpora had 10 genres, with a total of 149,775 news articles. After removal of empty articles and headlines, we have a total of 137,695 articles. 120,000 articles are used for training and 17,695 articles are used for model evaluation. The detailed data statistics for text data is presented in table 1.

Table 1. Data Statistics I

| S.N. | Genre | Number of Articles |
|---|---|---|
| 1 | National | 30,154 |
| 2 | Politics | 12,363 |
| 3 | Economics | 10,568 |
| 4 | Social | 17,159 |
| 5 | Arts | 6,515 |
| 6 | Sports | 8,253 |
| 7 | International | 19,346 |
| 8 | Health | 11,768 |
| 9 | Entertainment | 10,365 |
| 10 | Others | 11,374 |
|  | Total | 137,695 |

Nepali language has a complex morphology and complex preprocessing techniques need to be applied for obtaining a properly processed corpus. For text preprocessing, first the articles are tokenized into sentences. Tokenization is the act of breaking up a sequence of strings into words, phrases and other elements called tokens. Vertical bar (‘|’) (unicode U+0964) is used to break down the sentences while space bar (‘ ’) is used to break down the words.

Further, the collected data consists of many characters not found in Nepali Devanagari Character Set. These characters degrade overall text quality and impact model performance negatively. These unwanted characters are characters (a-z, A-Z), Arabic numerals (0-9), several unicode characters and special symbols like !,@,/, , *,(,),-,+,=,],[,;,,/,. They are removed from the dataset. Only characters in the range U+0900 to U+097F are retained. The data statistics after preprocessing is summarized in Table 2.

### 6.1 Encoding

The labels of the text data for both News articles and headlines were Nepali words. For converting these strings to numbers, each word was mapped to a distinct number. As a result, during training the neural network reads an array of numbers rather than the words

Table 2. Data Statistics II

| S.N. | Metric | Value |
|---|---|---|
| 1 | Average number of words in an article(before stop word removal) | 266 |
| 2 | Average number of words in an article (after stop word removal) | 206 |
| 3 | Average number of sentences in an article | 17 |
| 4 | Median number of sentences in an article | 12 |
| 5 | Median number of words in an article (before stop word removal) | 186 |
| 6 | Median number of words in an article (after stop word removal) | 145 |
| 7 | Coefficient of variation for word lengths | 0.94697 |
| 8 | Coefficient of variation for sentence lengths | 1.06756 |

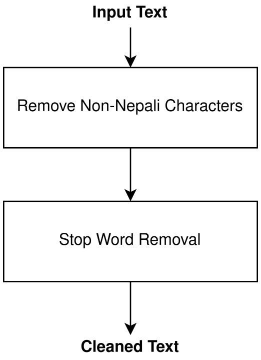

Fig. 3. Work flow in Text Preprocessing

itself. During inference, the neural network predicts word vectors, which are mapped to words using FastText.

Each individual word is converted to its corresponding vector using gensim library. We used FastText continuous bag of words model to generate the feature vectors for our model. FastText looks at n-gram representations of words and is designed to solve the Out of Vocabulary (OOV) issue.

### 6.2 Model Implementation

The model was implemented in TensorFlow using Keras. The model comprises of an encoder-decoder architecture. The encoder consists of a Bidirectional GRU layer, which is the input to the model. The encoder forms an internal representation of the input and it is passed to the decoder. There are two states from the Bidirectional GRU. The forward hidden state and the backward hidden state are concatenated and passed to the decoder as its initial state. The decoder, therefore, has twice the number of units as the encoder.

Rather than the usual approach to Headline Generation, which involves using a fully connected layer with a softmax activation at the output and posing the problem as a classification task, the problem is formulated as a regression task instead. The activation of the output layer in the model is linear. The use of a linear activation follows from the logic that the word embeddings are values in arbitrary range. Activations like ReLU, Sigmoid and tanh restrict the values to be in a certain range. Thus, a linear activation is more favorable. The output of the decoder is passed to another GRU layer which predicts the output sequences. The whole summary is predicted from the output of the GRU. Each timestep of the output layer is a word vector. The word vectors are converted back to words using FastText. Thus, the headline of a given article is obtained.

## 7. RESULT AND EVALUATION

Some of the headlines generated from model is shown in Table 3. The model is evaluated using the BLEU[14] score metric. For the training data, the average BLEU score is 32.6. For the testing set, the BLEU score is 22.1.

The results are evaluated manually as well. Since FastText was trained on the whole corpus, it treated the different forms of the same word as a separate entity. During testing, the generated headlines were found to contain multiple forms of the same word. The automatic evaluation does not take into consideration the different forms of the same word in Nepali language. As such, the BLEU score of the results suffers.

## 8. DISCUSSION

As seen from the results, the model does not fit the data well on some articles and the generated sequences do not match the headline perfectly. This is due to the improper training of FastText vectors. The dataset needs further cleaning. Also, inherent complexity in Nepali Language makes the problem of headline generation difficult as well. A Nepali word has several meanings and forms, and since they appear in similar context in literature, they are mapped closely in the vector space. This results in inconsistency during training and produces inaccurate results.

Stemming or lemmatizing the word would cause a loss in the actual meaning of the generated headline. Coherence between words is lost in this case. So, we decided to train the FastText model without removing the different forms of the word. Also, the problem was formulated as a regression task rather than a classification task because classification would involve using a fully connected layer at the last layer, causing the number of parameters to increase quadratically. This complicates the training time as well as memory requirements. Using regression, this problem is avoided. However, the regression of the vectors is not perfect and even if the value of some numbers in the vector is off by a little, we can get a completely different word. Also, since similar forms for a word are mapped closely, there is a higher chance of a misprediction.

## 9. CONCLUSION

This paper has demonstrated that Nepali news headline Generation can be carried out with GRUs with proper word vector representation. It can be concluded that the approach presented in this paper, can be well-used for Nepali Text Summarization and Headline Generation.

Headlines encapsulating abstraction could be implemented to a greater extent in subsequent papers on the topic by collecting data from other sources including blog posts, novels, and stories for

Table 3. Results from the model

| **Actual** | **Predicted** | **BLEU** |
|---|---|---|
| प्रधानमन्त्री ओलीद्वारा मनमोहन एनेक्स सेन्टर शिलान्यास | प्रधानमन्त्रीनिवास ओलीद्वारा मनमोहन एनेक्स सेन्टर शिलान्यास | 0.87 |
| फेरि बस्ने गरी सकियो राप्रपा सरकारबीचको वार्ता | केन्द्रीय सदस्य सहमतिपछि चार सकियो | 0.18 |
| सार्वजनिक कार्यक्रममै आरोप प्रत्यारोप | सार्वजनिक वार्ता वाग्लेद्वारा अस्वीकार माग | 0.30 |
| अब्बासी ओलीका पहिलो पाहुना | अब्बासी ओलीका पहिलो पाहुना | 1.00 |
| शुक्रबार पनि डोल्पाको सदरमुकाम दुनै पूर्ण रुपमा बन्द | सोमबारमा पनि डोल्पाको सदरमुकाम दुनै पूर्ण रुपमा बन्द | 0.87 |
| लालपुर्जा पाएर पनि सुकुम्बासी | लालपुर्जा पनि जग्गासमेत डोल्पाको | 0.46 |
| छुट्टै मगरात प्रदेश माग | छुट्टछुट्टै मगरात प्रदेश माग | 0.88 |
| कान्तिपुरको बयान आइतबारबाट मात्रै भिडियोसहित | कान्तिपुरमा बयान आइतबारबाट आजमात्रै भिडियोसहितको | 0.82 |

a wider vocabulary and a greater context. Also, models based on transformers that are state of the art in other language based tasks, like transformers can be implemented. Further preprocessing in the dataset, before training word vectors could also possibly increase the score of the model. Classification could also be done instead of the regression approach that has been used in this paper.

## 10. REFERENCES

[1] Text summarization: An extractive approach. In *Soft Computing: Theories and Applications*, pages 629–637, Singapore, 2020. Springer Singapore.

[2] P. B. Baxendale. Machine-made index for technical literaturean experiment. *IBM Journal of Research and Development*, 2(4):354–361, 1958.

[3] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[4] Latesh G. Malik Chetana Thaokar. Test model for summarizing hindi text using extraction method. *IEEE Conference on Information and Communication Technologies*, pages 1138–1143, 2013.

[5] Kyunghyun Cho, Bart Van Merrinboer, Caglar Gulcehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. 2014.

[6] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv.org*, 2014.

[7] Apurva Khude Dipali Telavane. Automatic summarization of hindi text documents using supervised learning method. *International Journal for Research in Engineering Application Management*, 4(10), 2019.

[8] Akshay Kishore Gattani. *Automated natural language headline generation using discriminative machine learning models*. PhD thesis, School of Computing Science-Simon Fraser University, 2007.

[9] D. Jain and G. Cardona. The indo-aryan languages. 2007.

[10] Konstantin Lopyrev. Generating news headlines with recurrent neural networks. 12 2015.

[11] H. P. Luhn. The automatic creation of literature abstracts. *IBM J. Res. Dev.*, 2:159–165, 1958.

[12] R. Kathleen Mckeown and R. Dragomir Radev. Generating summaries of multiple news articles. *SIGIR*, pages 74–82, 1995.

[13] Joel Larocca Neto, Alex A. Freitas, and Celso A. A. Kaestner. Automatic text summarization using a machine learning approach. In Guilherme Bittencourt and Geber L. Ramalho, editors, *Advances in Artificial Intelligence*, pages 205–215, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[14] Kishore Papineni, Salim Roukos, Todd Ward, and Wei jing Zhu. Bleu: a method for automatic evaluation of machine translation. pages 311–318, 2002.

[15] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[16] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.