

Transliteration of Sanskrit text

Deepanshu Gupta(15075011) & Sujal Maheswari(15075052)

IIT(BHU), Varanasi

01-May-2018

Table Of Contents

- 1 Objective
- 2 Background
 - Recurrent Neural Networks
 - Seq2Seq Model
- 3 Our Approach
- 4 Why use Long Short Term Memory(LSTM)
 - Deciding what information to throw away
 - Deciding what information to store
 - Updating the cell state
 - Giving the output
- 5 Results
- 6 Conclusions

- 1 Objective
- 2 Background
 - Recurrent Neural Networks
 - Seq2Seq Model
- 3 Our Approach
- 4 Why use Long Short Term Memory(LSTM)
 - Deciding what information to throw away
 - Deciding what information to store
 - Updating the cell state
 - Giving the output
- 5 Results
- 6 Conclusions

Objective

- To develop a Transliteration mechanism for Roman to Devanagari and vice versa.
- Transliteration is changing words from one script to another, more commonly the words are proper nouns.
- Sometimes it also means changing sounds from one language to another. For example:

h a j a g i r e e

हजगिरी

- This system is basically designed to help in retrieving old Sanskrit documents and manuscripts using different information retrieval techniques.

Objective

Google

धृतराष्ट्र उवाच

AI Videos Images Maps News More Settings Tools

About 19,800 results (0.56 seconds)

श्रीमद् भगवद्गीता | Gita Supersite
<https://www.gitasupersite.iitk.ac.in/srimad?...> ▾ Translate this page
धृतराष्ट्र उवाच धर्मक्षेत्रे कुरुक्षेत्रे समवेता युयुत्सवः । मामकाः पाण्डवाश्चैव किमकुर्वत सञ्जय ॥ 1.1 ॥ | Hindi Translation By Swami Ramsukhdas ॥ 1.1 ॥ धृतराष्ट्र बोले (टिप्पणी १.0 1.2) हे सञ्जय (टिप्पणी १.0 1.3) धर्मभूमि कुरुक्षेत्रमें युद्ध की इच्छासे इन्द्रपुत्रे हूए मेरे और पाण्डु के पुत्रोंने भी क्या किया. Hindi Translation By Swami Tejomayananda ॥ 1.1 ॥ धृतराष्ट्र ने कहा हे संजय धर्मभूमि कुरुक्षेत्र में एकत्र हुए युद्ध के इच्छुक (युयुत्सव) मेरे और पाण्डु के पुत्रों ने क्या किया. Sanskrit Commentary By Sri ...

Srimad - Two Book View | Gita Supersite
<https://www.gitasupersite.iitk.ac.in/srimad/bookview> ▾ Translate this page
नून रत्नकः । धृतराष्ट्र उवाच धर्मक्षेत्रे कुरुक्षेत्रे समवेता युयुत्सवः । मामकाः पाण्डवाश्चैव किमकुर्वत सञ्जय ॥ 1.1 ॥ श्रीमद् भगवद्गीता. Script. Assamese, Bengali, Devanagari, Gujarati, Gurmukhi, Kannada, Malayalam, Oriya, Roman, Tamil, Telugu. Chapter. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18. Sloka. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47. Show Translations and Commentaries. Hindi Translation By Swami Ramsukhdas. Hindi Translation By Swami ...

धृतराष्ट्र उवाच धर्मक्षेत्रे कुरुक्षेत्रे...
https://hi-in.facebook.com/permalink.php?story_fbid=824321804353187&id...
धृतराष्ट्र उवाच धर्मक्षेत्रे कुरुक्षेत्रे समवेता युयुत्सवः । मामकाः पाण्डवाश्चैव किमकुर्वत सञ्जय ॥ भावार्थः धृतराष्ट्र बोले- हे संजय! धर्मभूमि कुरुक्षेत्र में एकत्रित, युद्ध की इच्छावाले मेरे और पाण्डु के पुत्रों ने क्या किया?॥1॥ संजय उवाच दृष्ट्वा तु पाण्डवाण्येकं जुष्टं सुयोधनरत्नम् । आचार्यमुपसंगम्य राजा वचनमब्रवीत् ॥ भावार्थः संजय बोले- उस समय राजा दुर्योधन ने वृद्धरत्नमुक्त पाण्डवों की सेना को देखा और द्रोणाचार्य के पास जाकर यह वचन कहा ॥2॥ पर्यैतां ...

प्रथम अध्याय / Chapter 1 - VedicScripturesInc - Google Sites
<https://sites.google.com/site/vedicscripturesinc/home/.../chapter-one> ▾ Translate this page

Figure: Google search results for Sanskrit query.

Objective

The screenshot shows a Google search interface with the query 'dhritarashtra uvach' in the search bar. The results page displays 'About 1,310 results (0.51 seconds)'. The first result is titled 'Showing results for dhritarashtra uvacha' with a subtext 'Search instead for dhritarashtra uvach'. Below this, there are three search results:

- Result 1:** "Dhritarashtra said: O Sanjaya, after my sons and the sons of Pandu ...
www.kandamangalam.com > Itihasa > Bhagavad Gita
dhritarashtra uvaca dharma-kshetre kuru-kshetre samaveta yuyutsavaḥ māmakaḥ pandavaś caiva kim akurvata sanjaya. "Dhritarashtra said: O Sanjaya, after my sons and the sons of Pandu assembled in the place of pilgrimage at Kurukshetra, desiring to fight, what did they do?" » Bhagavad Gita 1.2. sanjaya uvaca drishta ...
- Result 2:** **Bhagavad Gita 1.1**
<https://www.bhagavad-gita.us/bhagavad-gita-1-1/>
Sep 14, 2012 · dhr̥tarāṣṭra uvāca dharma-kṣetre kuru-kṣetre samavetā yuyutsavaḥ māmakaḥ pāṇḍavās caiva kim akurvata sañjaya. Translation of Bhagavad Gita 1.1. Dhritarashtra said: O Sanjaya, after my sons and the sons of Pandu assembled in the place of pilgrimage at Kurukshetra, desiring to fight, what did ...
- Result 3:** **Bhagavad Gita Chapter 1 Verse 1- Dhritarashtra uvaca ...**
bhagavadgita.wiki/1/1
Dhritarashtra said: O Sanjaya, after gathering in the holy field of Kurukshetra with the intent to fight, what did my sons and the sons of Pandu do? ... Dhritarashtra, the blind king was the father of Kauravas and the uncle of Pandavas. ... Sage Vyasa grants his disciple Sanjaya (and devoted ...

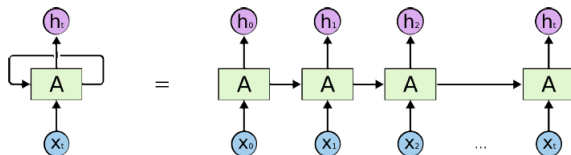
At the bottom, there is a link to '01-01 dhritarashtra uvaca dharma-kshetre kuru-kshetre samaveta ...' with a subtext 'schriften.yoga-vidya.de/.../01-01-dhritarashtra-uvaca-dharma-kshet...' and a 'Translate this page' button.

Figure: Google search results for Sanskrit query.

- 1 Objective
- 2 Background
 - Recurrent Neural Networks
 - Seq2Seq Model
- 3 Our Approach
- 4 Why use Long Short Term Memory(LSTM)
 - Deciding what information to throw away
 - Deciding what information to store
 - Updating the cell state
 - Giving the output
- 5 Results
- 6 Conclusions

Recurrent Neural Networks

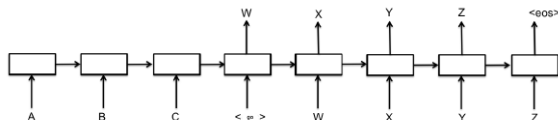
- RNN are a type of Neural network which contains loop in them, which basically allows information to persist.



- RNN emerged because of the need to address the issue of long term dependencies.
- A recurrent Neural Network can be considered as a repetition of the same network with each one of them passing message to its successor.

Seq2Seq Model

- Consists of two recurrent neural networks (RNNs).
- One of them is the encoder that processes the input given to it and the another one is the decoder that generates the output.



- The output of the decoder at time t is fed back to the algorithm and it becomes an input for the algorithm at time $t+1$.

- 1 Objective
- 2 Background
 - Recurrent Neural Networks
 - Seq2Seq Model
- 3 Our Approach
- 4 Why use Long Short Term Memory(LSTM)
 - Deciding what information to throw away
 - Deciding what information to store
 - Updating the cell state
 - Giving the output
- 5 Results
- 6 Conclusions

Our Approach

- Main problem faced by us in this task was to gather enough data so that our model could be trained efficiently.
 - To handle this problem we first converted Sanskrit text to Itrans notation.
 - This Itrans notation was then converted to Roman script by creating all possible mapping from Itrans to Roman characters which was created manually based on the phoneme that the Itrans was capturing.
 - The main advantage of this technique was that our data set was enriched with multiple ways of writing a given word in Roman script. Finally we trained our model on 1,52,000 words and cross-validated it on 38,000 words to fine tune the parameters and tested it on 10,000 words.

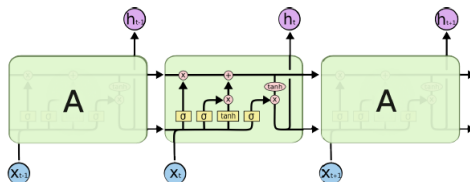
Our Approach

- To build our model we have used Seq2Seq model and then while decoding an unknown input sequence we go through a slightly different process.
 - Encode the input sequence into state vectors.
 - Start with a target sequence of size 1 (just the start-of-sequence character).
 - Feed the state vectors and 1-char target sequence to the decoder to produce predictions for the next character.
 - Sample the next character using these predictions (we simply use `argmax`).
 - Append the sampled character to the target sequence.
 - Repeat until we generate the end-of-sequence character or we hit the character limit.

- 1 Objective
- 2 Background
 - Recurrent Neural Networks
 - Seq2Seq Model
- 3 Our Approach
- 4 Why use Long Short Term Memory(LSTM)
 - Deciding what information to throw away
 - Deciding what information to store
 - Updating the cell state
 - Giving the output
- 5 Results
- 6 Conclusions

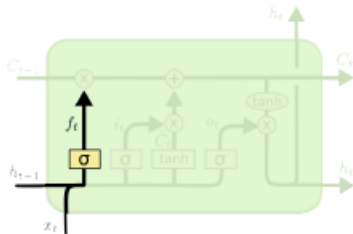
Why use Long Short Term Memory

- Long Short Term Memory networks usually just called LSTMs are a special kind of RNN, capable of learning long-term dependencies.



- In normal RNNs the repeating module generally have a simple structure like a \tanh layer but in LSTMs the repeating module has four neural network layers instead of one which interact with each other in a very trivial way.
- There are basically 4 steps to reach the output of a particular cell:
 - Deciding what information to throw away.
 - Deciding what information to store.
 - Updating the cell state.
 - Giving the output.

Deciding what information to throw away

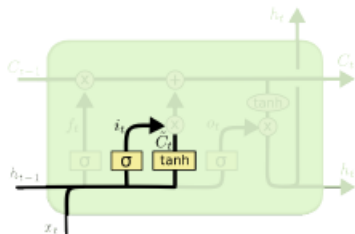


$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Figure: Forget Gate Layer.

- For this purpose, the "forget gate layer" is used.
- It is a sigmoid layer that looks at previous output(h_{t-1}) and the input for this layer(x_t) and generates a number between 0 and 1 that decides how much to keep and how much to throw away.

Deciding what information to store



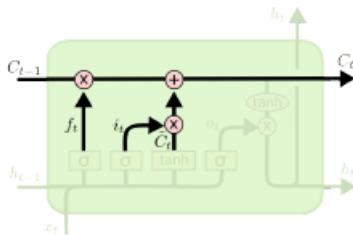
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

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

Figure: Input Gate Layer.

- To store information in a cell state, our second sigmoid layer "input gate layer" along with a tanh layer is used.
- First, the sigmoid layer chooses which information is to be added to cell state then the tanh layer prepares a vector (\tilde{C}_t) which can be combined with the above result to update the cell state.

Updating the cell state

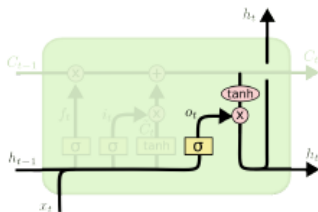


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Figure: Updating C_{t-1} to C_t .

- To update the cell state from C_{t-1} to C_t we multiply the old state by output from forget layer f_t to forget the old information and add it to the multiplication of it (what information to add) and C_t (the vector for new candidates). So the new state C_t becomes the above mentioned

Giving the output



$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

Figure: Output layer.

- The output we give is not the cell state but a filtered version of it because to predict the next thing maybe only some part of the information updated is required.
- This is done by another sigmoid layer known as "output layer" which controls which information should be passed ahead.
- This output o_t is multiplied to \tanh of the cell state (to reduce it to $[-1,1]$) to give the output.

- 1 Objective
- 2 Background
 - Recurrent Neural Networks
 - Seq2Seq Model
- 3 Our Approach
- 4 Why use Long Short Term Memory(LSTM)
 - Deciding what information to throw away
 - Deciding what information to store
 - Updating the cell state
 - Giving the output
- 5 Results
- 6 Conclusions

- We have experimented our model by taking different test-sets. BLEU score, Word-error rate and accuracy were different metrics that we used to capture the quality of the system developed.
- **BLEU (bilingual evaluation understudy)** is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another.
- **Word error rate (WER)** is a common metric of the performance of a speech recognition or machine translation system.

Results

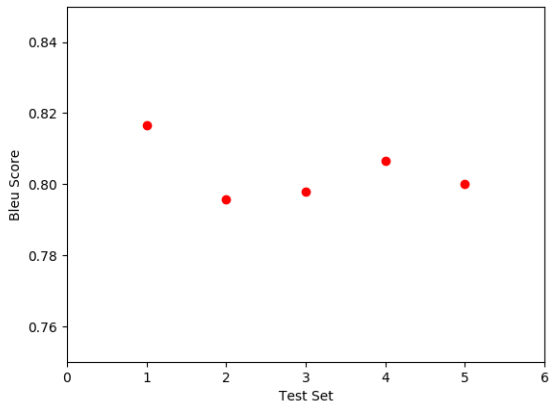


Figure: BLEU score vs Test-set

Results

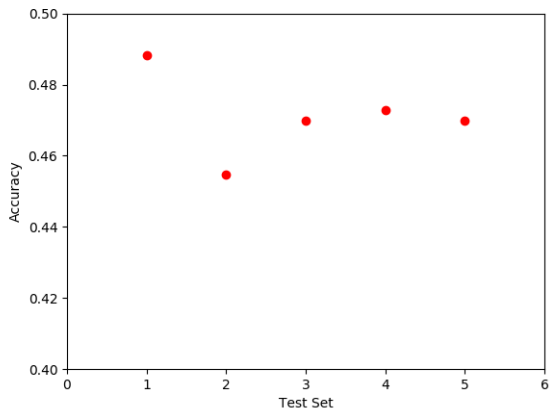


Figure: Accuracy vs Test-set

Results

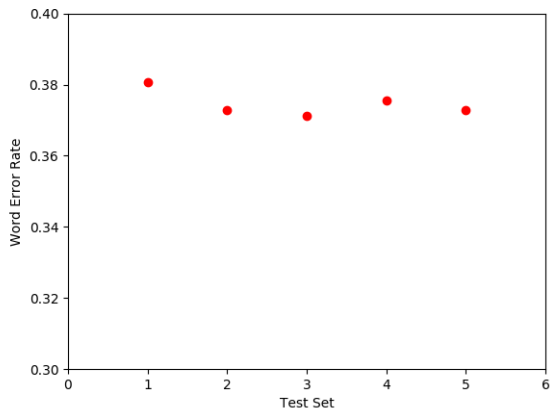


Figure: Word Error-Rate vs Test-set

- 1 Objective
- 2 Background
 - Recurrent Neural Networks
 - Seq2Seq Model
- 3 Our Approach
- 4 Why use Long Short Term Memory(LSTM)
 - Deciding what information to throw away
 - Deciding what information to store
 - Updating the cell state
 - Giving the output
- 5 Results
- 6 Conclusions

Conclusions

- We have achieved an overall accuracy of 47.11% and a BLEU score of 80.34 .
- We believe that if we train the model with better data-set, there is a good hope of getting our accuracy improved.
- The accuracy is low because it is calculated by matching the whole word with one another, so for a better measure we calculated BLEU score and word-error rate. The overall word-error rate was found to be 37.43%.
- We plan to extend our model for cross-lingual information retrieval and apply information retrieval techniques on both, the transliterated and the original query in order to retrieve the set of relevant documents of the both the source and target language.

Thank You