Custom Deep Neural Network for Message Scoring

Shailendra Singh Kathait
Co-Founder and Chief Data Scientist, Valiance Analytics Pvt. Ltd.
Noida, Uttar Pradesh

Ashish Kumar
Data Scientist, Valiance Analytics Pvt. Ltd.
Noida, Uttar Pradesh

Ikshu Chauhan
Research Scholar
Doon University

ABSTRACT

This paper offers customized message scoring creation of Technical Literature of Drugs using Text Analytics and Deep Learning. A customized Deep Learning driven model has beendeveloped that feeds scientific data (texts) about medicine(s) (written by medical researchers) & creates multiple sub messages that can be used for marketing research to effectively communicate benefits of drugs. Model helped in reducing manual intervention by automating the complete process and replacing Linguists.

Keywords

Natural Language Processing, Word Cloud, Deep Neural Network, CustomDeep Networks, Word Tokenization, Word Glove and Word2Vec, Marketing Campaign, Technical Literature. Medical Text

1. INTRODUCTION

The text mining studies are being important recently because of the availability of the increasing number of the electronic documents from a variety of sources.

The resources of unstructured and semi structured information include the world wide web, governmental electronic repositories, news articles, biological databases, chat rooms, digital libraries, online forums, electronic mail and blog repositories. Therefore, proper classification and knowledge discovery from these resources is an important area for research.

Natural Language Processing (NLP)[1], Data Mining, and Machine Learning techniques work together to automatically classify and discover patterns from the electronic documents. The main goal of text mining is to enable users to extract information from textual resources and deals with the operations like, retrieval, classification

Usage of neural networks for NLP [1] applications is attracting huge interest in the field of research and they are systematically applied to all NLP tasks.

Biomedical research is often confronted with large data sets containing vast amounts of free text that have remained largely untapped sources of information. The analysis of these data sets poses unique challenges, particularly when the goal is knowledge discovery. Health care is an information-intensive industry, and health care delivery is dependent on accurate and detailed clinical data.

An important goal of medical informatics is to facilitate access to and improve the quality of this information, thereby enhancing clinical outcomes. Data that are not available routinely in an easily accessible form represent a major challenge to this goal.

Heuristic messages are given by medical researchers about a drug discovery. It includes technical and scientific terms that can't be directly used into marketing campaign as it is not easy to understand for a non-technical person. With a heuristic message, Heuristic Score and Message Score have also been given which describe how effective a message is 1-**Good**, 2-**Average** and 3-**Bad.**

Definition of a heuristic category has also been given by which we can relate a message to its heuristic class. Modeling can be done for two variables 1) **Heuristic Score** and 2) **Message Score**.

In message score modeling, heuristic score predicted by Heuristic score model is also being used

- **Heuristic score 1**

Key words that speak to the heuristic, does this message leverage the heart of the heuristic, Would the readers take action, based on this message, Strong heuristic delivery

- **Heuristic score 2**

Weak, dial up heuristic, doesn't fully leverage the heuristic, Uses words in the name of the heuristic but fails to leverage the heuristic

- **Heuristic score 3**

Is it correctly tagged? Within family miscoding, does it touch on the irrationality at all? Does it get to the behavior changing aspect of heuristics, Poor or wrong heuristic usage

- **Message score 1**

Definitely better than the original, Approvable, It is clear and well-written to the right audience, No grammatical errors or typos, potentially fix grammar and typo issues when encountered, Should be clever and engaging, Draws you in the first few words – evokes an emotion in you/makes you feel something, Easy to digest

- **Message score 2**

Similar to original, Average message, nothing stands out particularly, at the same time, it is not a bad message either, nothing new when compared to the original – doesn't offer a new perspective, only mild changes to the original message

- **Message score 3**

Worse off than the original, Clunky, poor grammar, Factually incorrect, Incorrect data, Not approvable, Impossible or not worthwhile salvaging, Going to be rejected by consumer or will be mocked at, Too long or verbose, Too much punctuation, Sentence structure is highly complex, Too cheesy, Clumsy language, Half, written, Point is buried, Too colloquial, Tonality is off – original is official and authoritative/ours is colloquial.

In this article,it is being explores that heuristic message text as a raw text into different classification model and train them on

Heuristic Score and Message Score. As raw data is in text form and every word is related to other word this is how order of words matter. By contrast, order-sensitive models based on neural networks are becoming increasingly popular. Thanks to their ability to capture word order information.

Neural Networks are basically divided into two types RNN[2] and CNN[3] where CNN only considers current input layer while RNN works on the principle of saving the output of a layer and feeding this back to the input in order to predict the output of the layer

Bi-LSTM [4] is a type of special case of RNN where it feeds the learning algorithm with the original data once from beginning to the end and once from end to beginning.

In the given Medical research message, there are different terms of medical and generally non-technical persons don't use them. These observations motivate us to construct a textual modeling architecture that captures long-term dependencies without relying on meaning of words in a context.

2. RELATED WORKS

Text classification goes back to the early '60s [5] and in late '90s, techniques of machine learning have been continuously applied to text classification. Support Vector Machines were applied to text classification in a paper [6]

In recent decades, there has been an exponential growth in the number of complex documents and texts that require a deeper understanding of machine learning methods to be able to accurately classify texts in many applications. Deep Learning impact among industries came into picture in early 2000s, when CNNs [7] already processed an estimated 10% to 20% of all the checks written in the US, according to YannLeCun[8]. As RNN (alike CNN) works on the principle of saving the output of a layer and feeding this back to the input layers

Bi-LSTM deals with the exploding and vanishing gradient problems. The initial version of LSTM blocks includes cells, input and output gate. Among other successes, Bi-LSTM achieved record results in natural language text compression [9]. LSTM networks were a major component of a network that achieved a record 17.7% phoneme error rate on the classic TIMIT natural speech dataset (2013)[10]

Many approaches for medical text classification rely on biomedical knowledge sources [11]. On the other hand, some medical text classification studies use various types of information instead of knowledge sources. Although, these methods used rules, knowledge sources or different types of information in many ways. They seldom use effective feature learning methods, while deep learning methods are recently widely used for text classification and have shown powerful feature learning capabilities.

3. METHODOLOGY

Most of the text available in the data is simply a string of characters. Machine Learning Technique directly on such texts can't be applied directly. Hence, the primary step involves cleaning the input texts and translating them in to some numbers or vectors so that processing can be done in later phases. To achieve this, Natural Language library is used which is a popular platform for building python programs to work with human language data. The nltk/gensim provides most of the tools that is required for text cleaning and processing.

1. **Text Cleaning**: The first step to remove unnecessary numbers and special characters (punctuations and the other non-alphanumeric characters) from all sentences. This involves reducing data sparsity. All this performed by using python library **re**
2. **Stop word elimination and stemming**: There are many stop words in every natural language. Stop words are generally the most common words. There is no single universal list of stop words used by all natural language processing tools, and indeed not all tools even use such a list. Some tools avoid removing stop words to support phrase search. Removal of commonly used words (stop words) unlikely to be useful for learning.
   Reducing of related words to a common stem is called stemming. It is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form-generally a written word form.

3. **Tokenization:** As the name suggests tokenizing is splitting a string, text into a list of tokens. Whole text or each of the sentences is converted into tokenized words. This converts each sentence into a list of words, perhaps at the same time throwing away certain characters, such as punctuation form individual tokens. Converting the text into a list of tokens is important because this helps applying the tools of natural language to the text.
4. **Text to sequence and Padding**: Deep Learning algorithm can be performed on number or vector not on texts. All words need to be transformed into numbers or vector.
   Padding simply makes length of sentence same of those transformed texts by putting zero. In the Keras, deep learning library can be used to pad variable length sequences. The padding to be applied to the beginning or the end of the sequence, called pre- or post-sequence padding, can be specified by the "padding" argument.
5. **Bi-LSTM (Bi-Directional Long Short-Term Memory)**:Bi-LSTM Neural Network is a special case of RNN. In neural networks, there is a very specific use case where RNNs are required. In order to explain RNNs,it is needed to first understand what a sequence is. It is a stream of data (finite or infinite) which are interdependent. Examples would be time series data, informative pieces of strings, conversations etc. In a conversation a sentence means something but the entire flow of the conversation mostly means something completely different. Also in a time series data like stock market data, a single tick data means the current price, but a full day data will show movement and allow us to take decision whether to buy or sell.
   RNNs can be used in a lot of different places. Language Modeling is one of the most applicable use cases. Given a sequence of word, one may try to predict the likelihood of the next word. This is useful for translation since the most likely sentence would be the one that is correct.
   Sometimes, one only need to look at recent information to perform the present task. For example, consider a language model trying to predict the next word based on the previous ones. If it is being trying to predict the last word in "the clouds are in the sky," which is not needed any further context – it's pretty obvious the next word is going to be sky. In such cases, where the gap between the relevant information and the place that it's needed is small, RNNs can learn to use the past information.

   In theory, RNNs are absolutely capable of handling long-term dependencies too. A human could carefully pick parameters for them to solve toy problems of this form. But in practice, RNNs don't seem to be able to learn them. The problem was explored in depth by [12] who found some pretty fundamental reasons why it might be difficult

In this medical text classification task, model was trained for medical researcher's messages using Recurrent Neural Network (RNN), to be exact; it is examined Bidirectional LSTM [13] to process the messages. LSTM firstly introduced by [14] has proven to be stable and powerful for modeling long-time dependencies in various scenarios such as speech recognition and machine translations. Bidirectional LSTM [15] is an extension of traditional LSTM to train two LSTMs on the input sequence. BiLSTM is a reversed copy of the LSTM, so that It can take full advantage of both past and future input features for a specific time step. Training of Bidirectional LSTM networks using back-propagation through time (BPTT) [16] is developed. After the embedding layer, the sequence of word vectors is fed into a single-layer LSTM or Bidirectional LSTM to achieve another representation of h = LSTM/BiLSTM(s).

LSTM is denoted in Figure 1(c). It models the word sequence x as follows:

$$i_t = \sigma(x_t U^i + h_{t-1} W^i + b_i) \qquad (6)$$
$$f_t = \sigma(x_t U^f + h_{t-1} W^f + b_f) \qquad (7)$$
$$o_t = \sigma(x_t U^o + h_{t-1} W^o + b_o) \qquad (8)$$
$$q_t = \tanh(x_t U^q + h_{t-1} W^q + b_q) \qquad (9)$$
$$p_t = f_t * p_{t-1} + i_t * q_t \qquad (10)$$
$$h_t = o_t * \tanh(p_t) \qquad (11)$$

LSTM has three gates a) input gate i_t, b) forget gate f_t and c) output gate o_t. All gates are generated by a sigmoid function over the ensemble of input x_t and the preceding hidden state h_{t-1}. In order to generate the hidden state at current step t, it first generates a temporary result q_t by a tanh nonlinearity over the ensemble of input x_t and the preceding hidden state h_{t-1}, then combines this temporary result q_t with history p_{t-1} by input gate i_t and forget gate f_t respectively to get an updated history p_t, finally uses output gate o_t over this updated history p_t to get the final hidden state h_t.

6. **Layers explanation in Bi-LSTM:**
   - **A) Embedding Layer:**It is processed with Word Embedding as first layer Input (heuristic message and heuristic class definition).
   - **B) Bi-LSTM Layer:** Second layer using Bi-LSTM function given by Keras. It will give 512 output neurons for next layer
   - **C) Concatenate Layer:** Other handcrafted features and both message and definition of Heuristic class in string form.
   - **D) Output Layer:** Above concatenated layer is passed into a dense layer and results into an output layer
     - a. **Dense Layer:** where activation function is "relu" Rectified Linear Unit With default values, it returns element-wise max(x, 0). Otherwise, it follows:
       f(x) = max value for x >= max value,
       f(x) = x for threshold <= x < max value,
       f(x) = alpha * (x - threshold) otherwise.
     - b. **SoftMax Layer:** that gives 512 output neurons and this layer is again passed into final layer with softmax activation which gives three different probability values.
7. **Model Training:** The entire model is trained end to end using **categorical cross-entropy** loss and **Adam**optimizer as there are three categories.

4. IMPLEMENTATION

The above proposed method was implemented in python 3.6 using KERAS==2.2.4 and NLTK==3.3 library to pre-process the text. The program was run on data which have been provided by medical researchers and heuristicized definition by linguist. Code has been written with the help of libraries example numpy==1.15.3 pandas==0.23.4 scikit-learn==0.20.0 scipy==1.1.0 sklearn==0.0 tensorflow==1.11.0

With the help of Deep Learning, A model has been developed which can help us to identify message category feeding heuristic definitions into the model network. The method proposed was successfully developed and deployed the APIon server which classify new message given by medical researchers into Good, Average or Bad.

5. OBSERVATIONS

The proposed algorithm was tested for accuracy on different types of texts content of varying categories from which different key-phrases were extracted and checked for accuracy that is whether it is describing content of the texts completely or not. The results obtained were better than those obtained using statistical methods.

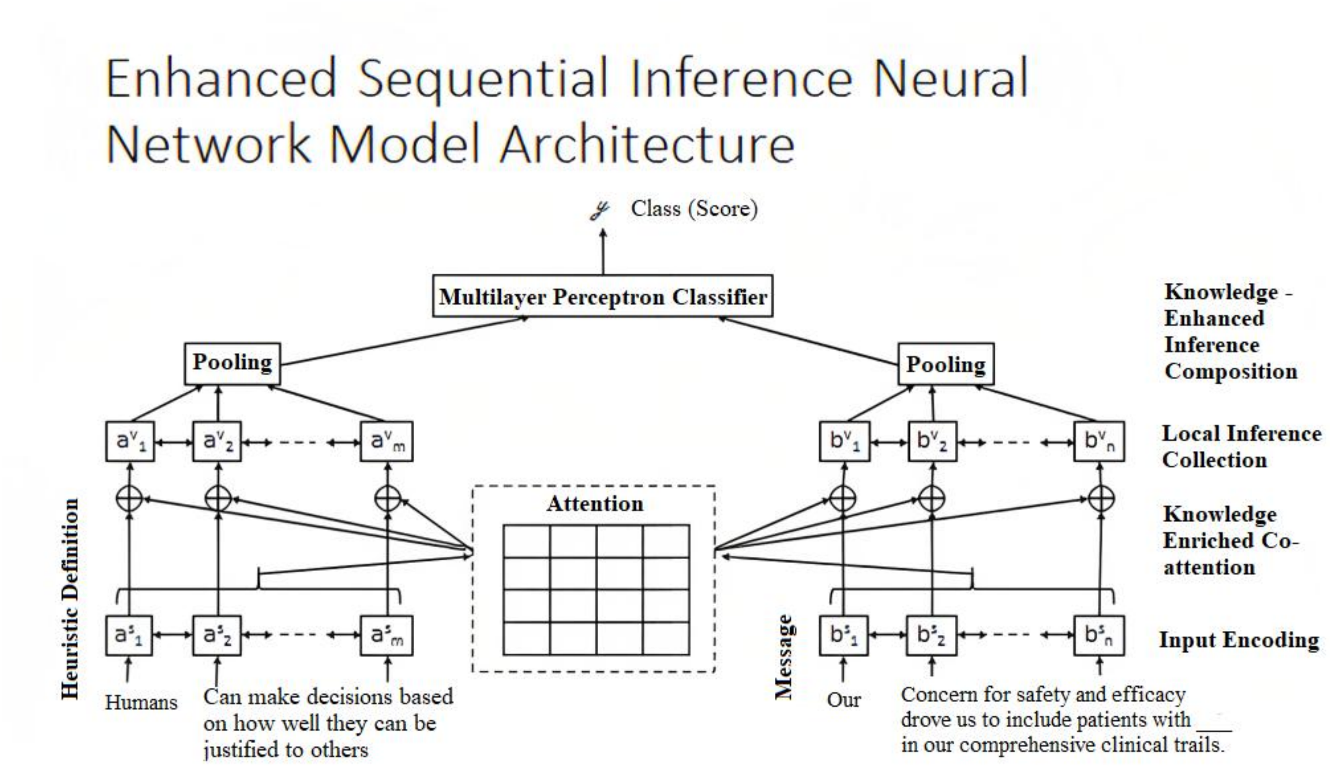
Enhanced Sequential Inference Neural Network Model Architecture
y Class (Score)
Multilayer Perceptron Classifier
Pooling
Pooling
Knowledge - Enhanced Inference Composition
Local Inference Collection
Knowledge Enriched Co-attention
Input Encoding
Attention
Heuristic Definition
Message
Humans
Can make decisions based on how well they can be justified to others
Our
Concern for safety and efficacy drove us to include patients with ___ in our comprehensive clinical trails.

Figure 1: Neural Network Model Architecture

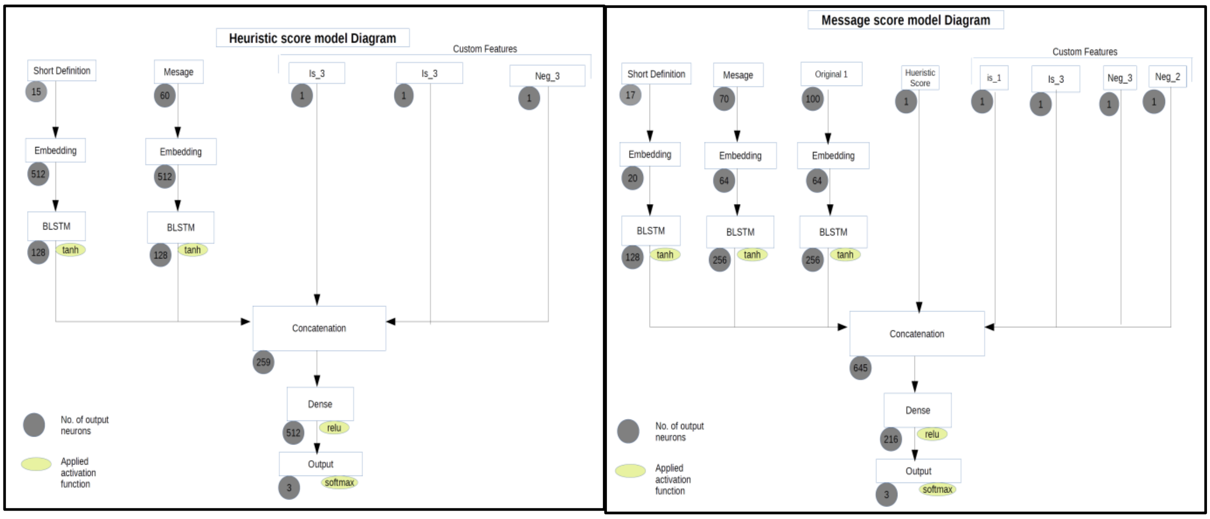

Figure 2: a) Heuristic Score

Figure 2: b) Message Score

Table 1: Observed Accuracy Table

Type of Network	Number of total Neurons	Extraction Accuracy
Sequential Inference	11346422	0.75
Custom-1	14534147	0.73
Custom-2	12353565	0.83

6. CONCLUSION

This paper introduces two models, one is Sequential Inference and other is custom layer implementation using Bi-LSTM. Both models can hold not only word importance but also probability of coming next word using previous word in a sentence. The experiments are conducted by changing number of layers and neurons in a number. Additionally, there are some Heuristic classes which belong more to a specific heuristic score or a message score on an average as it is checked by plotting bi-variate on them.

Comparing two models, extracted accuracy is better on an average and all three categories individually. A logic has also been put in assigning category. If probability of being 1 is less than 0.55 then it is put as new message into maximum of probability of 2 and probability of 3.

7. LIMITATIONS AND FUTURE WORK

This model can further be modified by using Word2vec or Glove dictionary but some medical terms with their antonyms and synonyms also needed to take care of by providing manual tagged features and looking at the importance of a word for a specific score.

8. REFERENCES

[1] Natural language processing (NLP) (Liu, 2015, Ravi, Ravi, 2015)

[2] Jeffrey L. Elman. 1990. Finding structure in time. Cognitive Science 14(2):179–211.

[3] YannLeCun, Leon Bottou, YoshuaBengio, and Patrick ´ Haffner. 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11):2278–2324.

[4] SeppHochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". Neural Computation. 9 (8): 1735–1780. doi:10.1162/neco.1997.9.8.1735. PMID 9377276

[5] Sebastiani, 2002

[6] Joachims, 1998; Dumais et al., 1998

[7] Waibel, Alex (December 1987). Phoneme Recognition Using Time-Delay Neural Networks. Meeting of the Institute of Electrical, Information and Communication Engineers (IEICE). Tokyo, Japan

[8] YannLeCun (2016). Slides on Deep Learning Online

[9] "The Large Text Compression Benchmark". Retrieved 2017-01-13 6

[10] Graves, Alex; Mohamed, Abdel-rahman; Hinton, Geoffrey (2013-03-22). "Speech Recognition with Deep Recurrent Neural Networks"

[11] Wilcox AB, Hripcsak G. The role of domain knowledge in automating medical text report classification. J Am Med Inform Assoc. 2003; 10(4):330–8

[12] Hochreiter (1991) [German] and Bengio, et al. (1994),

[13] (Zeng et al., 2016)

[14] (Hochreiter and Schmidhuber, 1997)

[15] Graves and Schmidhuber, 2005; Graves et al., 2013

[16] Chen and Huo, 201