Implementation of Vision Techniques on NLP for Summary Evaluation and Generation

Priyank Pathak [1,2], **Prof. Harish Karnick** [2] , **Vivek** [3]

[1]Undergraduate 5th year EE and CS Double Major

[2]Department of Computer Science – IIT KANPUR

[3]Microsoft Research, Bangalore

{ppriyank,hk}@iitk.ac.in , keviv9@gmail.com

Abstract. *Currently, Text Evaluation is limited to use of ROUGE, BLEU, F1 Scores, which don't match with human judgment scores. Taking an image from Computer Vision as an analogy for the document in a Linguistic space, we try to implement similar approaches for it in the hope of surpassing results of BLEU scores and ROUGE scores for text evaluation as well as text generation. The project report includes A proposal of sentence vector based evaluation metric (DUC-2004 dataset). Furthermore, a theoretical proposal for techniques of incorporating state-of-the-art caption generators for Headline generation. Treating summarization process as a caption generation problem as implemented in Computer Vision, thus a joint modal learning model for evaluating Headlines with respect to the whole document (Twenty News Dataset)*

***Keywords**: Text Evaluation, Summarization, NLP, Vision.*

1. Introduction

The project is based on the idea of taking the advantage of the similarity between images and documents for caption/headline generation, thus the implementation of various Vision techniques from Vision. The presentation primarily focuses on the indulgence of DeepNets in summary evaluation.

Text evaluation of a summary refers to the Comparison of a summary with respect to its original/reference document and giving it a score, on how accurately it covers the topic. The reasons NLP is difficult is: a) No common notion of smallest unit in literature as opposed to a pixel in Vision. It can be characters, words, sentences or even paragraphs [1]. If sentence vector really the best strategy? (3-word sentences and 91 words having the same dense representation). b) No Universal tokenizer.Common tokenizers from Keras, NLTK have shortcomings. The common notion, ”internet is full of articles, offers sustainable training data”, is a myth, since hours and hours of pre-processing is needed, unlike Image.

The primary question one needs to ask when working with summarization is, what makes a good summary ”good”? [2]
1) Coverage of important topics from the reference document 2) Good Coherent Structure
3) Generic if “Abstractive” 4) Non-redundant 5) Good Readability

While summarizing is still an open problem, the following are the question one needs to ask oneself before evaluating the summary. 1) What are the important points/sentences in a reference document? 2) More sentences from "central idea" better than covering the entire document? 3) How to give an "Abstractive" summary, written with no "Extractive" words from the reference document, the same score as that from the extractive one (generalized)? (for a valid scoring metric) 4) Can incorporating "Literary devices" help? 5) How to ensure grammatical readability? 6) How to keep a check on the coherent structures which prevents misinterpretations, even if 2 sentences are taken from the document without any context. 7) Assigning weights to sentences based on their importance/idea conveying power in the reference document? 8) How can the golden summary help if the scoring is taking place between the reference document and the generated summary? 9) Although Word order is important but is longest subsequence matching really the best method? 10) How does Humans create Summaries or do they Judge them? 11) What are current methods of evaluation? 12) What dataset to use for evaluation?

Our contribution

- Proposal of Sentence Vector based method for scoring a summary with respect to original document.
- Treating summarization as the image segmentation problem.
- Treating summarization as Caption Generation Problem.
- Use of cross modalilty of caption generation for evaluation, using a differentiator architecture.

2. Previous Works

We have gone through various document representation techniques and evaluation methods before stating our approach. We also preform deep analysis of caption generation while stating it as a Headline generation problem.

- Character based NLP processing and CNN Understanding Text from Scratch. [1]
- EVALUATION MEASURES FOR TEXT SUMMARIZATION, Detailed analysis [2]
- Skip thought Vectors for converting sentences to Vector format. [3]
- FCN for semantic Segmentation [4]
- Where to put the Image in an Image Caption Generator [5]
- Automatic Description Generation from Images: A Survey of Models, Datasets, and Eval-uation Measures [6]
- Architecture for Caption Generation [7]
- CNN base model for caption generation [8]

3. Approaches

We first use DUC-2004 data set and Twenty News Data set, to create sentence vectors for the articles and abstract.

3.1. Sentence Vector Based Evaluation

We treat sentence vectors (skip thought vectors[3]) as the basic unit of representation of documents. in DUC-2004 dataset. Since no sentence tokenizer works correctly we deploy our own regular expression based tokenizer. The procedure is as follows:

- Run weighted K-Mean algorithm on these sentences vectors. The Topic priority will be defined as the no of sentence vectors that cluster has

- For each sentence in the summary, match it with the centroid of the closest cluster, and assign it a score. Basic representation is in the form of word vectors and sentence vectors, so they can handle abstractedness.
- For Coherency and Non-redundancy : Sentence(i) will be followed by Sentence(i+1) that shall lie closest to it in the semantic space compared to all S(i+k) for $k >= 2$ and similarly. For a given sentence i, $i + 1^{th}$ and all proceeding sentences should have an angle greater than threshold $(theta\_threshold1)$ for avoiding sentences redundancy about the same idea. The $i + 1^{th}$ sentence should be at most $theta\_threshold2$ away to maintain coherent structure.

Theta(i,j) is the angle between sentence i and sentence j.
Define a Delta Function D(i,j)

if $j = i + 1$

$D(i, j) = W_1 * (Theta(i, j)–theta\_threshold1) + W_2 * (theta_thresh old2 - Theta(i, j)$

elif : $j \neq i + 1$ and $j \geq i + 1$

$$D(i, j) = W_3 * (Theta(i, j)–theta\_threshold1)$$

If $j \leq i$:

$$D(i, j) = 0$$

$$S\_coherent\_redundancy = 1/(m^2) \sum_{i=1}^{K} \sum_{j=1}^{K} D(i, j)$$

$Score(Document, Summary) = W_1 1{*}Score\_Cluster{+}W_2 2{*}S\_coherent\_redundancy$

3.2. Headline Generation as Image Segmentation

A common problem statement in Computer Vision is of image segmentation, i.e. given an image, segment that image in shapes that belong to the same object.[4] Corresponding aspect, in NLP is Topic modelling, that is given a document recognize important topics in it, (or highlighting important sentences) or clustering the sentences under a common topic. So, we took sentence vectors and word vectors of a document. (300 dimensions). Used fully convolutional architecture, with the only difference is the the kernel size, here it's a nx300 (n is the number of sentences or words taken in consideration) whereas, in vision, kernel are generally around 5x5 or 3x3. We used 1,2,3 concatenated version as a 1-gram, 2-gram and 3-gram model. For generation purpose, the model needs to output 300 dimension , i.e. deconvolution of x300!! (model failed miserably, running short of memory). We have tried about 10 different architectures reducing some aspects which paper proposes but still couldn't manage it on a 1 GPU system. (Excessive bulky model)

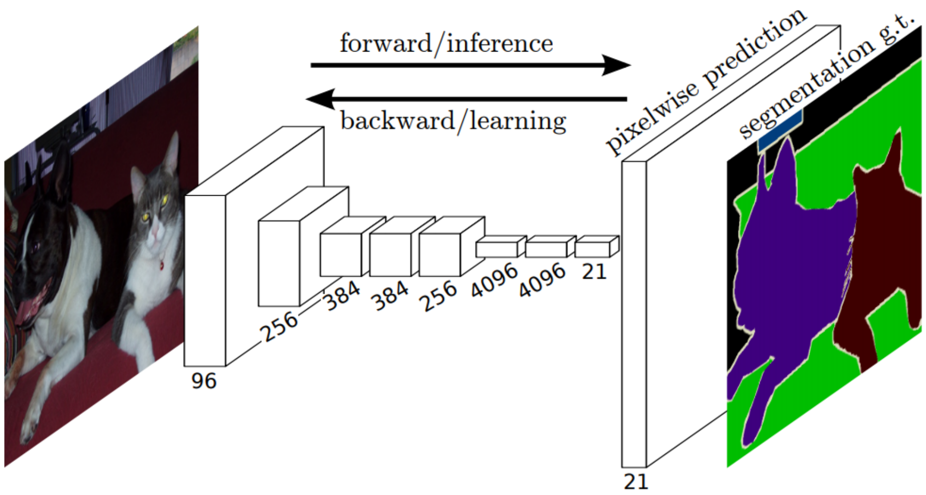


Figure 1. Fully Convolutional Neural Network. Figure taken from Fully Convolutional Networks for Semantic Segmentation [4]. Image is replaced with a Document

3.3. Caption Generation

Original Problem is a difficult task since one has to map image space to linguistic space.A similar problem can be viewed as headline generation for a document. It can be extended to whole summary generation, but LSTM (basic unit of text generation) holds good up to only 32 – 40 words. On paper, this looks like a simpler problem, since both input and the generated text are in the same linguistic space.

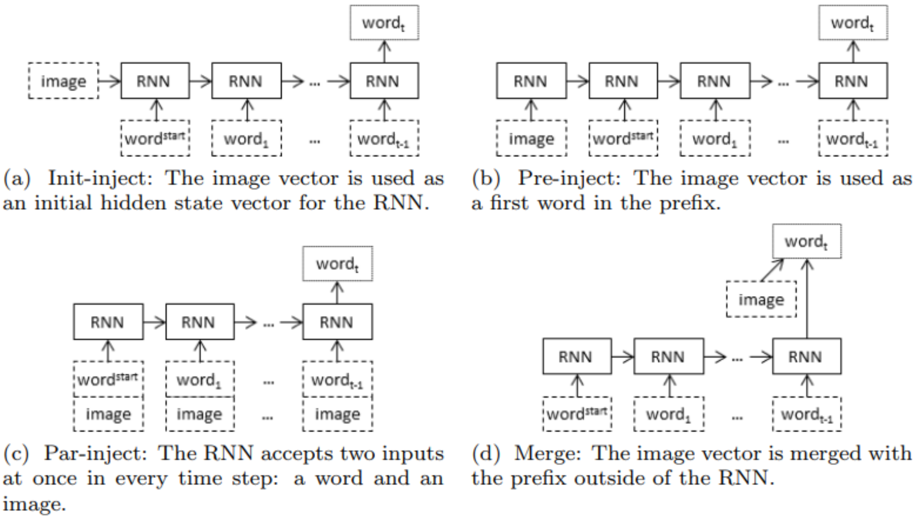
Headline Generation via par-inject method, i.e. feed the document vector for production of all words. Use Paragraph vectors (non-trainable) to feed the RNN in addition to the previous word, similar to caption generation problem. Dataset is TWENTY NEWS DATASET. The architecture is shown in Figure 2

3.4. Caption Generation on Joint Modal System

Use of CNN from the paper: CNN architecture for sentence classification [8]. It's the state-of-the-art for 10 class sentiment classification and outperformed every model for our 18 class classification. Its role is equivalent to what VGG has in Vision, so similarly can be modified for various tasks. Feeding Document to CNN enables us to makes the size of input arbitrary long, something LSTM suffers from. (current headline generation method are restricted to LSTMs, so a CNN model can produce the same result while removing the input size constraints.)

Without Differentiator

- Dataset used is Twenty News Dataset
- Vocabulary = 1.4 lakhs, after intersection with GloVe (50D) dictionary its 65K. Words whose names were not found, were ignored.
- Max caption/Headline is 49 words , shorter headlines were padded with zeros. Similarly, articles are of max length 2742 words.
- Tokenization via NLTK
- 2 parallel set of architectures are used to created 50 dimension outputs each.



(a) Init-inject: The image vector is used as an initial hidden state vector for the RNN.

(b) Pre-inject: The image vector is used as a first word in the prefix.

(c) Par-inject: The RNN accepts two inputs at once in every time step: a word and an image.

(d) Merge: The image vector is merged with the prefix outside of the RNN.

Figure 2. The Figure is taken from Where to put the Image in an Image Caption Generator [5]. The above are all the possible methods where one can insert document vectors to generate Headline

- Weights are l2 regularized and dropout is applied before final output layer (keep prob = 0.5)

With Differentiator

- All the previous entries remain the same.
- 2 parallel set of architectures are used to created 50 dimension outputs each. The input are now 3: Golden summary, Garbage Summary and Original Doc. The Summary CNN share weights, so garbage summary and golden summary provided create D_{real} and D_{fake} thus again 2 parallel architectures are used.
- Weights are l2 regularized and dropout is applied before final output layer (keep prob = 0.5)
  This time the output of differentiator is $Dreal$ and D_{fake}.

$$D_loss = -tf.reduce_mean(tf.log(D_real) + tf.log(1. - D_fake))$$

$$G_loss = -tf.reduce_mean(tf.log(D_fake))$$

4. Results

4.1. Sentence Vector Based Evaluation

The sentence vectors didn't form proper clusters, with sentence vectors mainly concentrated in the 0th cluster with other clusters having only one vector for the sake of having one. Even online clustering algorithms such as Birch was unable to distribute proper clusters. 77 sentences and 72 clusters and adjusting it, finally fixed it to 19 clusters, but mostly had only one vector. The theta1 threshold comes out to close 62 degree and theta 2 threshold comes to be 65 deg. With no gap as such found, this assumption fails. Even the Golden Summary doesn't hold up the theta assumption.

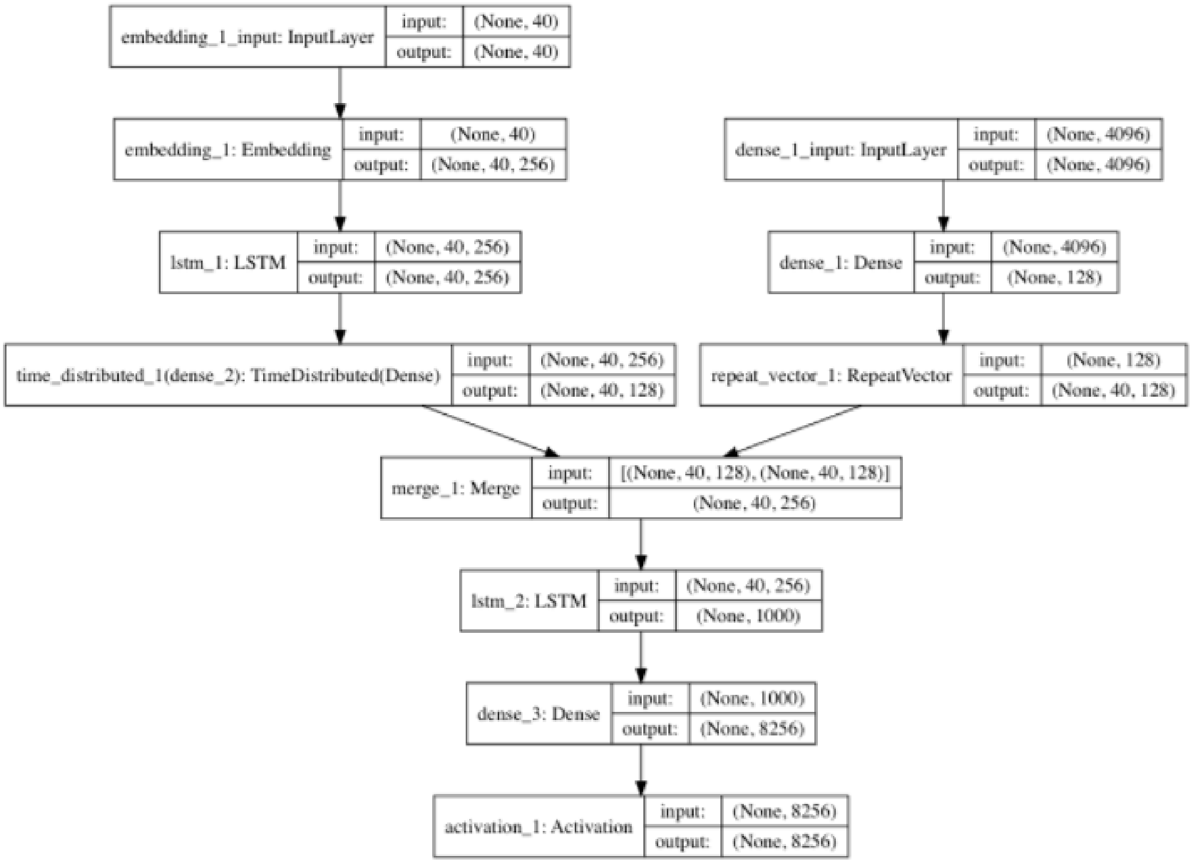


Figure 3. The is the architecture taken from github of image caption generation [7]

4.2. Caption Generation

2 sample outputs:

”re : new study out on gay percentage encryption method ? ? ? ? ? ? ? ? ? ? ! ! ! ! ! ! ! – ¿ last it should they mean anything ?) ? ? ! ! ! ! !) long (first your code can car”

and

”re : new study out on gay percentage encryption method ? ? ? ? ? ? ? ? ? ? ? ? ! ! ! ! ! ! ! – – murders apr questions – ! ! ! (was : re : new no good clinton [1] part were”

Model seems to change only the last few tokens

4.3. Caption Generation Joint Modal System

Without Differentiator Results are quite impressive yet the difference was not substantial, more analysis will be needed.

With Differentiator See Figure 8. Yellow denotes the result scores for Golden summary and Blue shows the scores for garbage summary by the ROUGE metric and Our trained Model. Model without sigmoid and with sigmoid is the output from diffentiator after applying the sigmoid layer.

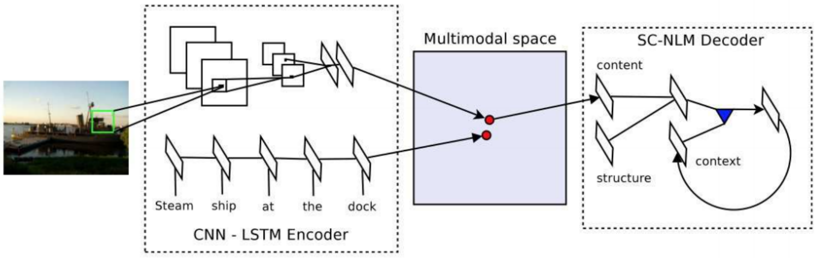


Figure 4: The encoder-decoder model proposed by Kiros et al. (2015).

Figure 4. Cross modal system, figure taken from Automatic Description Generation from Images: A Survey of Models, Datasets, and Evaluation Measures.[6]. The architecture shows the joint learning system, but our case the cross modal space is the same

5. Future Proposals

- Extensive testing on sequential degradation
- Getting hold of DUC-2005, which have human annotated scores to match sequential degraded summary to get a correlation matrix with human judgment and comparison with ROUGE/BLEU scores.
- Changing the absolute label 0,1 to intermediate label such as rouge scores
- To show its works better than Rouge Scores and BLEU
- To increase body size limit to show its effectivity on large input (CNN - Daily Mail Dataset)
- Use of Attention. Modifying the CNN architecture for headline generation. Removing the Fully connected layer from the CNN model leaves the model with n-filter channels, that can be substituted for dimension of hidden layer of a LSTM, to train a headline generation model. Github available for Show and Tell model.

References

[1] Text Understanding from Scratch
https://arxiv.org/pdf/1502.01710.pdf

[2] EVALUATION MEASURES FOR TEXT SUMMARIZATION, Detailed analysis.
http://www.cai.sk/ojs/index.php/cai/article/viewFile/37/24

[3] Skip-Thought Vectors
https://arxiv.org/pdf/1506.06726.pdf

[4] Fully Convolutional Networks for Semantic Segmentation
https://arxiv.org/pdf/1605.06211.pdf

[5] Where to put the Image in an Image Caption Generator
https://arxiv.org/pdf/1703.09137.pdf

[6] Automatic Description Generation from Images: A Survey of Models, Datasets, and Evaluation Measures
https://arxiv.org/pdf/1601.03896.pdf

[7] Architecture for Caption Generation
https://github.com/anuragmishracse/caption$_{g}enerator$

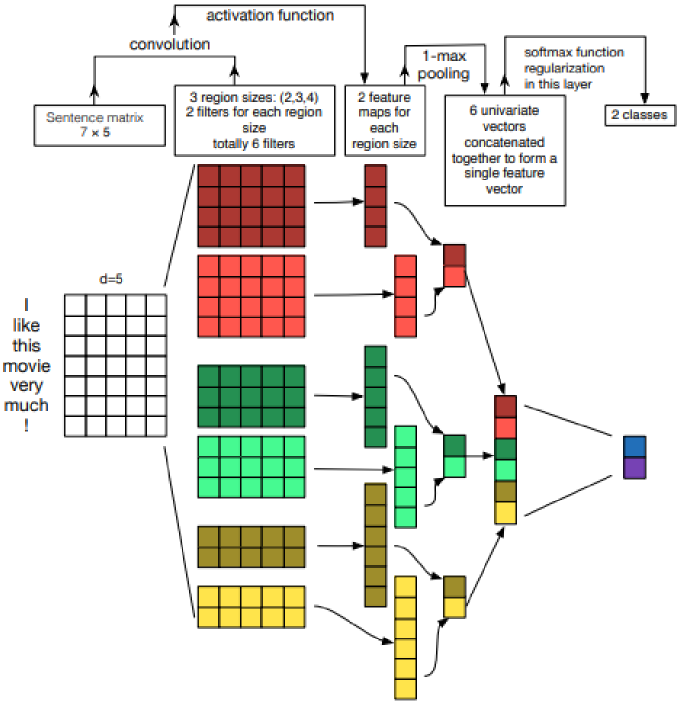

activation function
convolution
1-max pooling
softmax function regularization in this layer
Sentence matrix 7 × 5
3 region sizes: (2,3,4) 2 filters for each region size totally 6 filters
2 feature maps for each region size
6 univariate vectors concatenated together to form a single feature vector
2 classes
d=5
I
like
this
movie
very
much
!

Figure 5. CNN architecture taken from A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification [8]

[8] A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification
https://arxiv.org/pdf/1510.03820.pdf

[9] Show, Attend and Tell: Neural Image Caption Generation with Visual Attention
https://arxiv.org/pdf/1502.03044.pdf

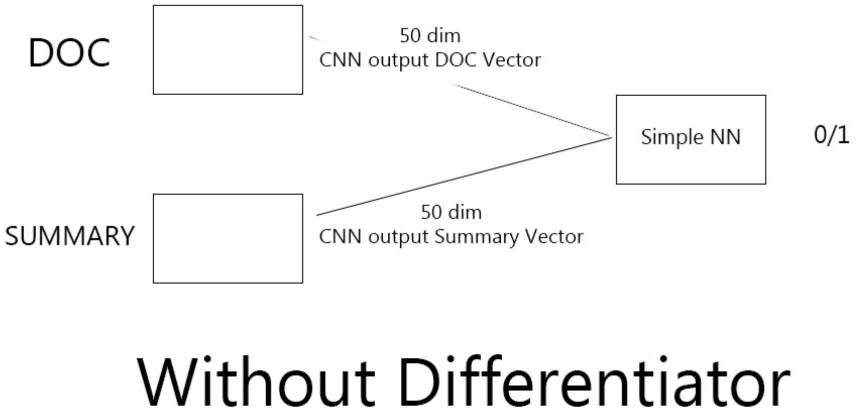


Figure 6. The output NN takes the concatenated 100 dim input and gives a score based on overlap

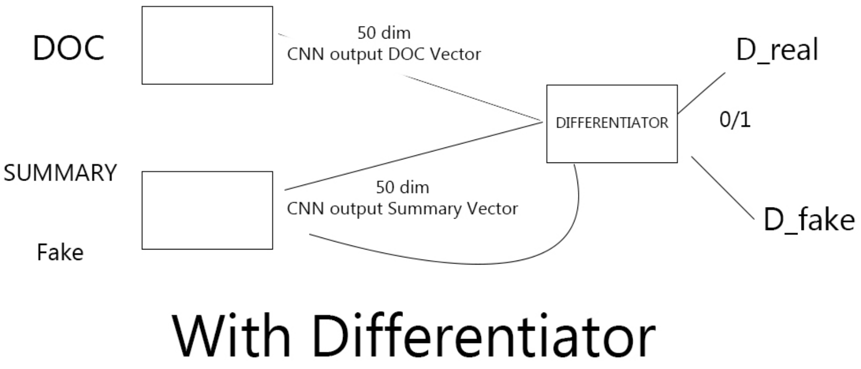


Figure 7. The output gives D_{fake} and D_{read} based on summary input

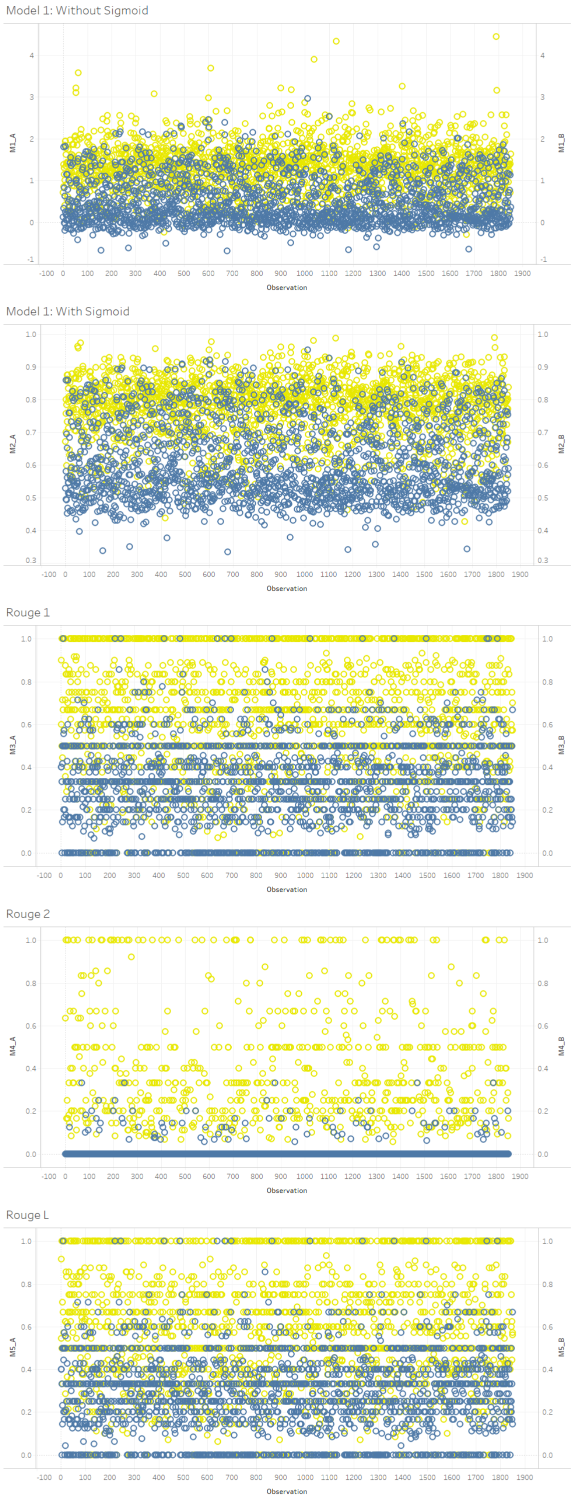


Figure 8. Yellow Circle represents the score on golden summary and Blue represents the scores on Garbage summary. Top Most Model is The score by the differentiator model with out sigmoid function. second is with sigmoid function. 3rd is ROUGE-1 scores on the same. 4th are the ROUGE-2 scores on the same. The last image is the ROUGE-L scores.