Comparing Similarity Between the Title and the Content of News Articles Using Deep Learning

M.H.E.U. Jayasundara* and R. Yasotha

Department of Physical Science, Vavuniya Campus of the University of Jaffna, Sri Lanka
*2014erandra@vau.jfn.ac.lk

Abstract

The volume of the online news has rapidly increased for recent years. People who consume the news online have increased. New technologies have changed the way of consuming online news. It has become a huge adoption of accessing news with the help of smart devices. When people read online news, they scan the headlines first and then pay attention to the content. Online Readers visit the articles in brief based on the headlines. Spotting fake news from factual news is a challenging task. This paper presents a method to separate fake news from factual news by comparing the similarities between the headlines and the content of the articles using deep learning methods. In this proposed method, the abstractive text summarization technique is used to extract the summary and then generate new headlines for the articles. The selected dataset has 60 000 news articles and collected from a benchmark dataset, BBC news. This model performs the best effort and achieves the accuracy 74% for selected samples.

Keywords: deep learning, recurrent neural networks, text summarization

Introduction

The emerging technologies in the Internet, web and the mobile have become embedded in every aspect of our day to day lives. These are completely changing the way people interact with others. The way people consume newspaper articles has been changed by the development in the technologies.

In recent years, there has been a considerable increase in the number of people accessing the news via online. A news media may be a print media including newspapers and magazines or broadcast news using radio or television or the online news. There is a significant difference in the way people consume news on print media and online. When people read a newspaper, they read it from page to page while scanning headlines, and reading articles that are more interesting. In contrast, when people read an online news, they definitely scan the headlines first and then pay attention deeply to it. Online Readers visit the articles in brief based on the headlines. Therefore, the function of the headlines increases the interest of the readers to read the whole article.

Primary function of a headline was to give the reader, who was scanning the newspaper, a clear understanding of what the article was about (Van Dijk 1988). But since many headlines are not read within the context of a newspaper anymore, the function of the headline has shifted. The headline is being one of the primary ways to attract the readers' attention. The headlines should make the reader curious as to what the article is about, so that it lures the reader into opening the article (Chen, Conroy, and Rubin 2015). Therefore, it is important to have a good understanding of the characteristics of an effective headline. This research aims to compare the similarity between the headlines and the content of the online news articles.

Literature Review

A lot of work has been carried out on generating headlines. Text summarization methods are statistical based and focus on extracting and compressing a sentence (Knight and Marcu, 2000). Recently, summarization methods on large corpus size have been applied using the neural network

and worked on an abstractive summarization method (Rush et al., 2015). The abstractive method makes summaries based on the article comprehension which may be considered as what humans usually do on summarization. By understanding the content and making summarization, sorting the leading words which originate from the parent article, (*Nallapati et al.,* 2016). In the Nallapati model, he proposed several methods that address some problems on summarization that are not addressed by the basic architecture, such as modelling the keywords capturing the hierarchy of sentence-to-word structure and avoiding words that are unseen at the training time. It was reported to further improve in performance.

Another learning approach, reinforcement learning is also used in both abstractive text summarization (Paulus, Xiong and Socher, 2018) and extractive text summarization (Narayan et al. 2018a). Narayan et al. introduced a novel method which summarized a whole news article to one sentence and gave what actually news was about. In 2020, Xiaotao Gu et al. developed a distant supervision approach to train a large-scale generation model without any human annotation.
In 2020, *Yun-Zhu Song et al.* worked on generating an attractive headline, which is the best described news. They have introduced a novel method called Popularity-Reinforced Learning for inspired Headline Generation (PORL-HG). In their proposed method, significantly exceeded the state-of-the-art headline generation models in terms of attractiveness, while the faithfulness of PORL-HG is the same as the state-of-the-art generation model.
However, none of the existing approaches has been considered in determining the headline will best describe the content of the article. All works have been carried out on the aspect of the writer's perspective. Very few approaches on the readers side to verify the headlines.

Methodology

Dataset

This study works with two datasets. In the first dataset, there are 60 000 news articles collected from a benchmark dataset, BBC news (Greene, D and Cunningham. P, 2005). For the second dataset, data is collected from two popular websites. First one is the Ada Derana (adaderana.lk) which is the brand for newscasts on the TV Derana television network in Sri Lanka, and the next one is the government official news portal (news.lk). The contents in the Ada Derana are available in three languages, Sinhala, Tamil and English. Web-scraping technique is used for extracting data from the above two websites. 38000 web pages were considered for the web scraping process.

Data cleaning and preprocessing

The first step is to load the data. In this research, considered corpus is to be loaded. Every article in a corpus contains an actual headline and the content. The next important step is data cleaning and preprocessing. During the cleaning stage, as the first task, it has removed the non-relevant information such as white spaces, punctuations, etc. Then all texts in the articles were converted to lower cases in order to map the all words with different cases to the same form. The next step in the preprocessing was stemming. Stemming is the process of reducing inflation in words to their root from. The Steaming process reduces the dimension of the context too. Finally, the stop words have been removed. During the stop word removal task, the low information words were removed from the content. Therefore, the research can only focus on the important words in the content.

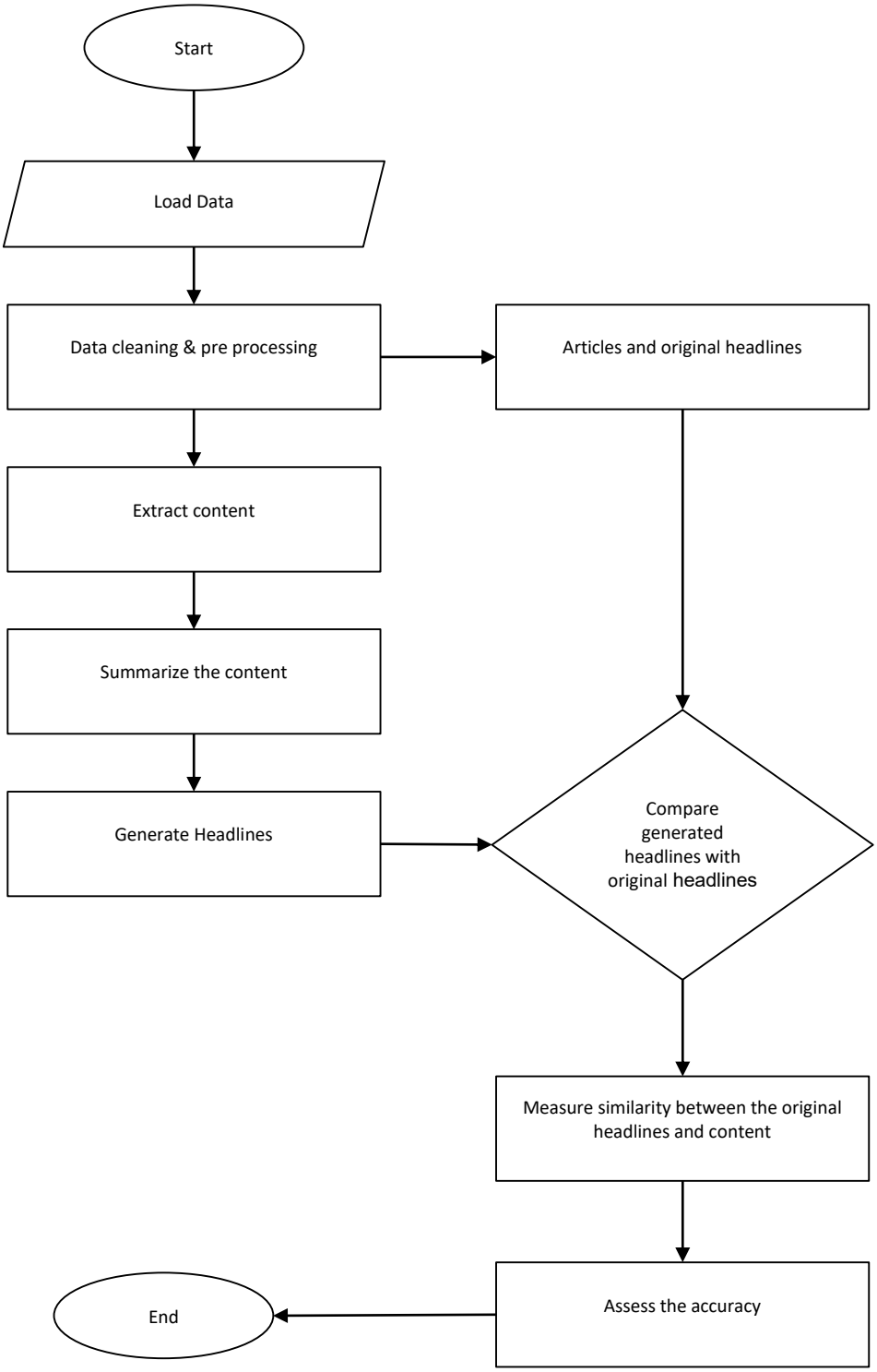
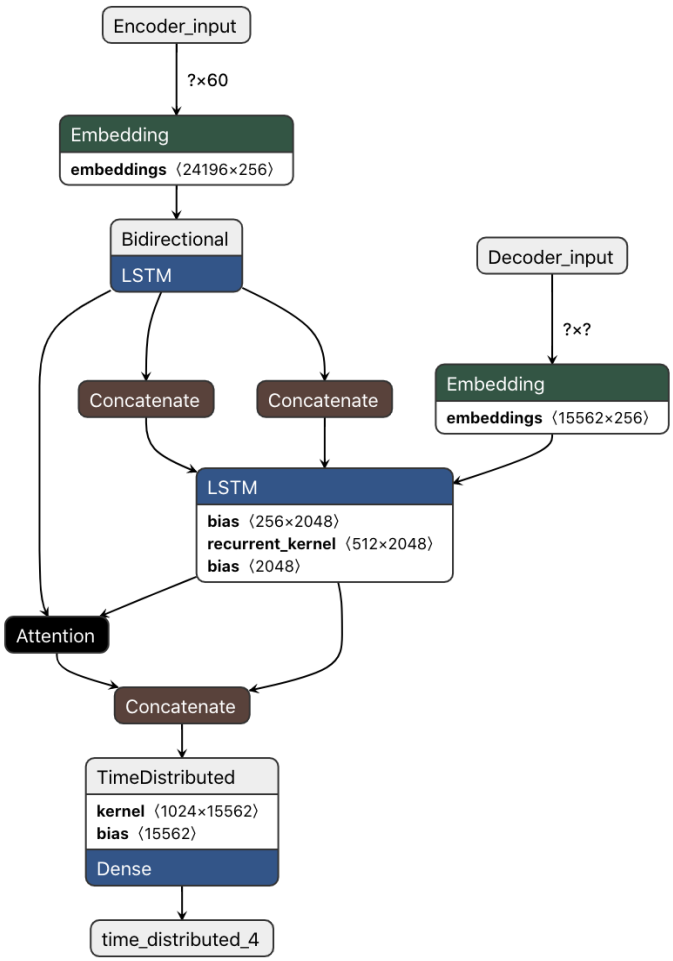


Figure 1 Pipeline Representation

Text summarization

From the extracted content, this research generates headlines using the text summarization technique. Two broad categories of approaches to text summarization are extraction and abstraction. Extractive methods select a subset of existing words, phrases, or sentences in the original text to form a summary. In contrast, abstractive methods first build an internal semantic representation and then use natural language generation techniques to create a summary. This research extracts the summary using the abstractive method and generates new headlines.

Headline Generation



Encoder & Decoder

Sequence to sequence recurrent neural network architecture is developed for machine translation has proven effective when applied in the text summarization. Sequence to sequence means when the neural network is divided into two separate classifiers, one called an encoder and the other called the decoder. The encoder learns and generates a single embedding that effectively summarizes the input whereas the decoder learns from this single embedding and generates a sequence of output.

Embedding

Neural network uses the numbers to train and predict data. Embedding is a kind of mapping of discrete to represent each text in the articles. In the context of neural networks, embeddings are low dimensional learned continuous vector representation of discrete variables. In the proposed architecture, encoders generate embeddings in order to reduce the dimensionality of categorical variables and meaningfully represent categories in the transformed space.

Long Short Term Memory (LSTM)

Since words in a sentence may depend on the previous words on the sentence. RNN is capable of predicting the next word by observing the immediately previous word. When considering the two consequence sentences: *Today has a beautiful blue.... Today has a beautiful blue sky.* Where the context words help to predict the next word *'sky'*. But, it may encounter cases where the context word that is required to predict the next word may present at the beginning of the sentence. When considering the two consequence sentences: *I lived in Sri Lanka, I can Speak ... I lived in Sri Lanka, I can Speak Sinhala.* Where the context word "*Sri Lanka*" appears much earlier in the sentence, but RNN only try to predict the word "*Sinhala*" take in to account the word immediately preceding it "*Speak*", "*Can*", "*I*" but none of them will not help to predict the word "*Sinhala*". Therefore, it is needed to add another component to the model called Long Short Term Memory (LSTM) to avoid this problem. LSTM has an extra pipeline of context called cell state which passes through the network.

Figure 3: Model accuracy, loss variance with epochs without using LSTM

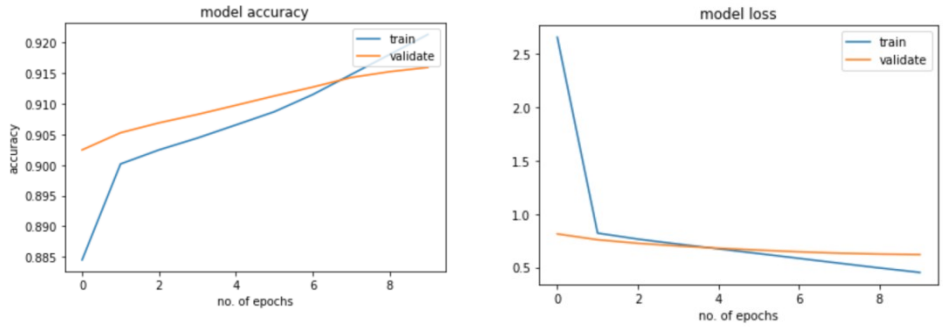


Figure 3 shows how accuracy and loss changes with epochs for the model that has only encoder and decoder and not an LSTM. It has achieved better accuracy in train dataset as well in the validate dataset but the predicted summary was not a good, understandable sequence of words and it was limited to few words or one, shown in Figure 4.

News	**Original Headline**	**Predicted Headline**
"the court of appeal today issued an interim injunction order restraining prime minister mahinda rajapaksa and his cabinet from functioning until the hearing of the writ of quo warranto filed against them holding office is concluded"	*"interim order issued preventing pm cabinet functioning"*	*"court pm"*
"skywatchers are preparing for the latest supermoon as earth s satellite makes its closest approach since"	*"supermoon viewers get the closest glimpse since"*	*"supermoon"*

"the main suspect accused of instigating the incidents in kandy has been arrested this morning along with other suspects"	*"main suspect kandy incident arrested"*	*"kandy arrested"*

Figure 4: Table of predicted headlines from train model without LSTM

Then, it is needed to re-train the model with a unidirectional LSTM and it gives the same accuracy and loss level but this time, the predicted summary was not limited to a few words. It gave a sequence of words but some words were repeated over and over, shown is Figure 5.

News	**Original Headline**	**Predicted Headline**
"the court of appeal today issued an interim injunction order restraining prime minister mahinda rajapaksa and his cabinet from functioning until the hearing of the writ of quo warranto filed against them holding office is concluded"	*"interim order issued preventing pm cabinet functioning"*	*"court pm pm pm"*
"skywatchers are preparing for the latest supermoon as earth s satellite makes its closest approach since"	*"supermoon viewers get the closest glimpse since"*	*"supermoon viewers viewers closest"*
"the main suspect accused of instigating the incidents in kandy has been arrested this morning along with other suspects"	*"main suspect kandy incident arrested"*	*"main kandy kandy kandy kandy"*

Figure 5: Table of predicted headlines from train model with unidirectional LSTM

This happens due to the LSTM layer used in the model as unidirectional and the same word predicted once and continues to pass forward influencing the words being predicted later in the sentence. This issue was overcome by putting a bidirectional LSTM layer. Thus, the words present even after the target word to influence the prediction.

Concatenate
Since the bidirectional LSTM generates two states: *forward* and *backward.* The final state should be a combination of these two. Therefore, the function of the concatenate layer merges these two states to one.

Attention
A neural network layer will assign a static weight for a token, but that may fail the accuracy of the summarization. Therefore, assigning weight should be learned by the value itself and needed dynamic weight for the token when required.

Similarity measures
After a generation of headlines from input corpus then it analyses the similarity between the predicted headlines and original headlines using three different methods and then chooses the best method which provides a higher rate of accuracy. There are three methods that are considered to measure the accuracy: Jaccard similarity, cosine similarity and the universal sentence encoder.

Jaccard Similarity
Jaccard similarity or intersection over union can be defined as the size of the intersection divided by the size of the union of two sets.

$$similarity\ score = \frac{size\ of\ the\ intersection}{total\ sentence\ size}$$

I lived in Sri Lanka so I can speak Sinhala.
I have been living in Sri Lanka for more than 10 years.

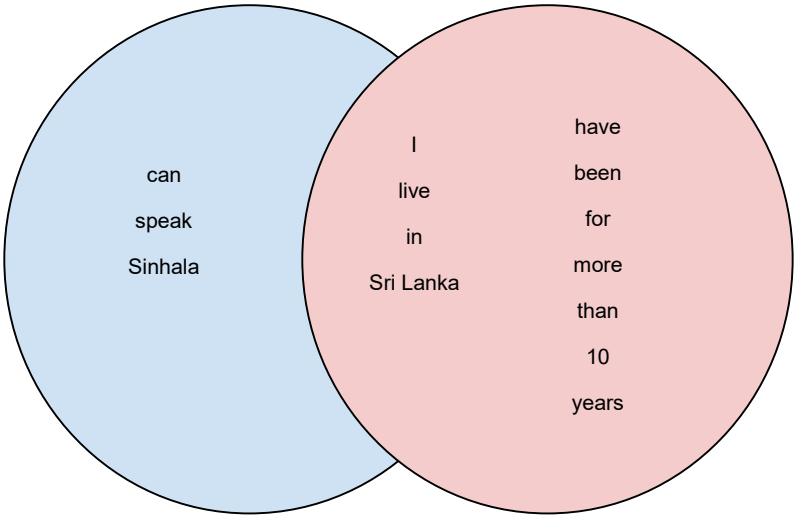


Figure 6: Intersection representation of two sentences.

First, perform **lemmatization** to reduce words to the same root word. In this case, "lived" and "living" will both become "live".

$$jaccard\ similarity\ score = \frac{4}{4+3+7} = 0.2857$$

Cosine Similarity

Cosine similarity calculates similarity by measuring the cosine of the angle between two vectors.

$$cosine\ similarity\ score = cos(\theta) = \frac{A.B}{||A||||B||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\ \sqrt{\sum_{i=1}^{n} B_i^2}}$$

In Jaccard similarity, the similarity score is proportional to the number of words in the intersection of two sentences. When the word count increases in the sentences, then the intersection word count also increases even though two sentences mean something different entirely, the score may be higher. Cosine Similarity is a metric used to determine how similar the documents are irrespective of their size (P. Xia, L. Zhang 2015).

When plotted on a multi-dimensional space, where each dimension corresponds to a word in the document, the cosine similarity captures the orientation (the angle) of the documents.

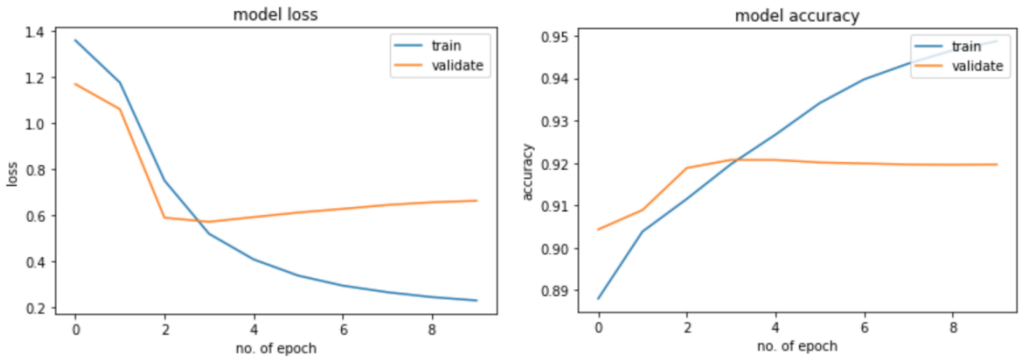
Universal sentence encoder

The Universal Sentence Encoder (Cer, Daniielle et al. 2018) is a sentence embedding model invented by Google that converts text into semantically-meaningful fixed-length vector representations. There are two Universal sentence encoder models one is based on transformer architecture and the other one based on Deep averaging network.

This work utilizes the transformer architecture that is based on the model. The sentence embedding makes context-aware representations for each word to deliver sentence embeddings. It is outlined for higher precision, but the encoding requires more memory and the computational time. This can be valuable for sentiment classification where words like 'not' can alter the meaning and are able to handle two-fold negation like "not bad".

Result and Discussion

The considering dataset has more than 60000 news articles. Training set contains 70% of selected articles and the remaining 30% is chosen as a validation set. The model was trained with a various learning rate of 0.001, 0.01, 0.1 and 10 epochs, where the learning rate is a hyperparameter that controls how much the model changes in response to the estimated error at each time for the updated model weights. In this research study, choosing the learning rate would be a challenging task. When the value of learning rate is too small, it may result in a long training process that could get stuck, whereas a value of learning rate too large, it may result in learning a sub-optimal set of weights too fast or an unstable training process. The major aim is to vary the learning rate to find a better rate to the proposed model.



(a) (b)

Figure 7: Model loss (a) and model (b) accuracy in learning rate of 0.01

From figure 7(a), it can be seen the model took considerable decrease in the first two epochs and remained fairly unchanged the next epoch and obtained a steady increase from 3rd epoch onwards. This is due to overfitting of the model, therefore to avoid this overfitting, added a callback to the model to stop training when loss achieved the minimum. And the model trained much faster than the other learning rates and achieved the maximum accuracy within a few epochs. Therefore, the learning rate, 0.01 is suitable for our model.

News	**Original Headline**	**Predicted Headline**
"the court of appeal today issued an interim injunction order restraining prime minister mahinda rajapaksa and his cabinet from functioning until the hearing of the writ of quo warranto filed against them holding office is concluded"	*"interim order issued preventing pm cabinet functioning"*	*"court consider petition challenging challenging pm request"*
"skywatchers are preparing for the latest supermoon as earth s satellite makes its closest approach since"	*"supermoon viewers get the closest glimpse since"*	*"supermoon viewers viewers closest glimpse"*
"the main suspect accused of instigating the incidents in kandy has been arrested this morning along with other suspects"	*"main suspect kandy incident arrested"*	*"main suspect arrested shooting incident"*

Figure 8: Result output from validation dataset

Figure 8 shows the sample results obtained by the model. After the generation headlines, analysed semantic similarity between the original headlines and the predicted headlines from the data where the model did not see previously at the training phase or validation phase.

After finding the similarity score from each method mentioned at similarity measures, it was essential to make an assumption. The similarity score more than 60% means that the content and the original headline had the same. If not, it was labelled as not-similar. The generated confusion matrix for each similarity measure was considered. Finally, find the accuracy of the model. Accuracy is evaluated as follows:

$$accuracy = \frac{Number\ of\ Correct\ prediction}{Total\ Number\ of\ prediction}$$

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

$$accuracy = \frac{True\ positive\ + True\ negative}{True\ negative\ +\ False\ positive + True\ positive + False\ negative}$$

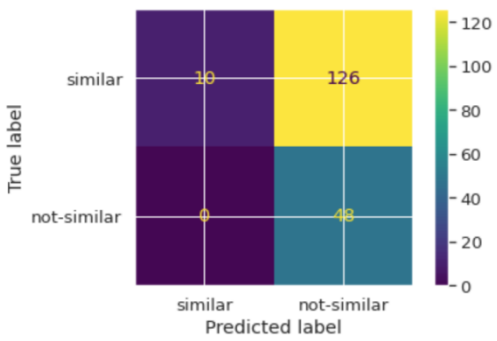
Since our data is not balanced, this metric is not suitable to measure the accuracy of the model. Therefore, it was tried to use another metric, called F_1score.

$$F_1\ score = 2 \times \frac{precision \times recall}{precision + recall}$$

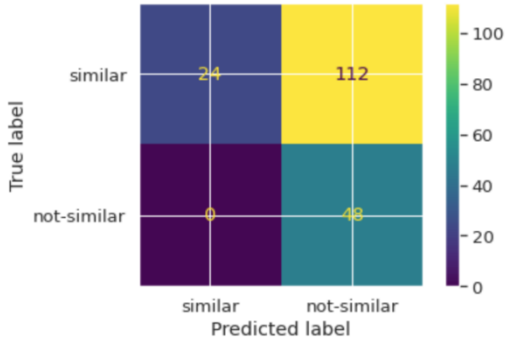
$$precision = \frac{True\ positive}{True\ positive\ +\ False\ \ positive}$$

$$recall = \frac{True\ positive}{True\ positive\ +\ False\ negative}$$

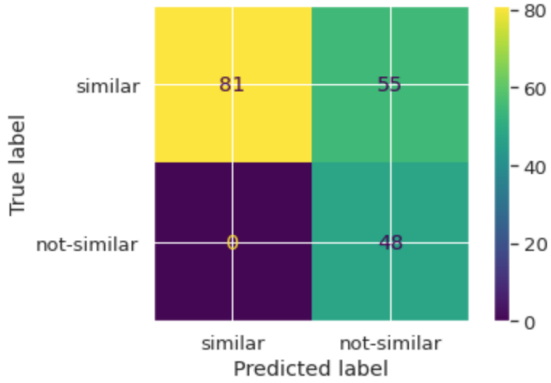
Jaccard similarity gave an F_1score of 0.1369 and the confusion matrix is shown in Figure 9a. Cosine similarity provided the F_1score of 0.3 is shown in the Figure 9b. Universal sentence encoders obtained an F_1score of 0.7465 is shown in Figure 9c.



(a)



(b)



(c)

Figure 9: Confusion matrix of Jaccard,Cosine and USE Similarity

It can be concluded that the universal sentence encoder provides better results for the assumed similarity probability. It must tune the assumption value to a point where it gives a much higher f1 score value.

From the confusion matrix at figure 9(c), true similar and predicted not-similar cells have higher error rate, this means it has labelled similar data to not-similar, therefore it was tuned the assumed value to 50% and analysed the confusion matrix shown in figure 10(a). And again, gave an f1 score value of 0.8809. But still find some values in that cell again. Therefore, it can reduce tuning value much more. Hence, it was reduced to 35% and re-calculateF_1score and confusion matrix is shown in figure 10(b). And, it gave a f1 score of 0.906 at that time true not-similar - predicted similar cell has increased up to 22 that means tuning value over reduce. Therefore, it increased the value up to 40% and redo theF_1 score and confusion matrix is shown in figure 10(c). And given anF_1score of 0.91 and it seems to be a fair amount.

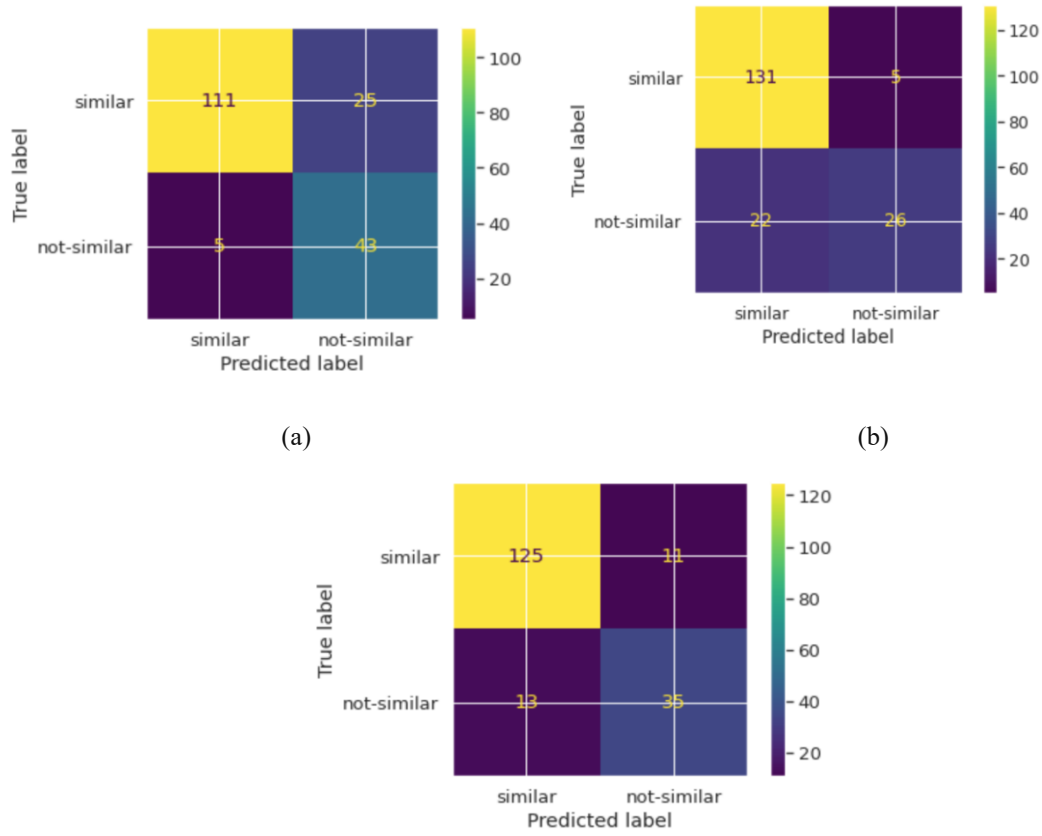


Figure 10: Confusion matrices for tuning value

Conclusion

This research study, mainly focused on the abstractive summarization methods to summarise the original articles and generate new headlines. Generated headlines were compared with the original headlines. Similarity measures, Jaccard similarity, Cosine Similarity and Universal sentence encoder are used to examine the distance between the original headlines and the content of the articles. In this work, calculate the score for similar titles with the content using the recurrent neural networks. The proposed model achieved an accuracy of 74% for selected dataset which has 60000 news articles collected from a benchmark dataset, BBC news. Estimating the leaning rates was the challenging task.

References

Rubin, V.L., Conroy, N.J., and Chen, Y. (2015). "Towards News Verification: Deception Detection Methods for News Discourse". *In Proceedings of the Hawaii International Conference on System Sciences (HICSS48) Symposium on Rapid Screening Technologies, Deception Detection and Credibility Assessment Symposium*, January 5-8, 2015 Grand Hyatt, Kauai [Accessed 6 February 2020].

Younus L.L,(2019)"Analysis of the Structure of Scientific News Headlines in Online Newspapers", *Journal of College of Education for Women*, 2019. Available: 10.36231/coedw/vol30no4.16 [Accessed 6 February 2020].

Ecker, S. Lewandowsky, E. Chang and R. Pillai,(2014) "The effects of subtle misinformation in news headlines.", *Journal of Experimental Psychology: Applied*, vol. 20, no. 4, pp. 323-335. Available: 10.1037/xap0000028 [Accessed 9 February 2020].

Knight, K., and Marcu, D. 2000. Statistics-based summarization-step one: Sentence compression. AAAI/IAAI 2000:703–710.

Rush, A. M.; Chopra, S.; and Weston, J. 2015. A neural attention model for abstractive sentence summarization. In *Empirical Methods in Natural Language Processing* Available: arXiv:1509.00685v2 [cs.CL] 3 Sep 2015

Filippova, K.; Alfonseca, E.; Colmenares, C. A.; Kaiser, L.; and Vinyals, O. 2015. Sentence compression by deletion with lstms. In *Empirical Methods in Natural Language Processing* (EMNLP), 360–368.

Nallapati, R.; Zhou, B.; Gulcehre, C.; Xiang, B.; et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. SIGNLL *Conference on Computational Natural Language Learning* (CoNLL).

Yun-Zhu Song, Hong-Han Shuai, Sung-Lin Yeh, Yi-Lun Wu, Lun-Wei Ku, Wen-Chih Peng, "Attractive or Faithful? Popularity-Reinforced Learning for Inspired Headline Generation" Available: arXiv:2002.02095v1 [cs.CL]

Narayan, S.; Cohen, S. B.; and Lapata, M. 2018b. Ranking sentences for extractive summarization with reinforcement learning. In the *North American Chapter of the Association for Computational Linguistics*.

Narayan, S.; Cohen, S. B.; and Lapata, M. 2018a. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Empirical Methods in Natural Language Processing* (EMNLP).

Xiaotao Gu, Yuning Mao, Jiawei Han, Jialu Liu, Hongkun Yu, You Wu, Cong Yu, Daniel Finnie, Jiaqi Zhai, Nicholas Zukoski, 2001 "Generating Representative Headlines for News Stories "Available: arXiv:2001.09386v1 [cs.CL] 26 Jan 2020.

Sieg A. 2020, "Text Similarities: Estimate the degree of similarity between two texts", Medium. [Online]. Available: https://medium.com/@adriensieg/text-similarities-da019229c894. [Accessed: 09- Feb- 2020].

D. Greene and P. Cunningham. "Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering", Proc. ICML 2006. [PDF][BibTeX]

"Sri Lanka News | Sri Lankan Breaking news,Hot News - Adaderana -Truth First", *Adaderana.lk*, 2020. [Online]. Available: http://www.adaderana.lk/news_archive.php. [Accessed: 19- May- 2020]

P. Xia, L. Zhang and F. Li, "Learning similarity with cosine similarity ensemble", *Information Sciences*, vol. 307, pp. 39-52, 2015. Available: 10.1016/j.ins.2015.02.024.

Cer, Daniel & Yang, Yinfei & Kong, Sheng-yi & Hua, Nan & Limtiaco, Nicole & John, Rhomni & Constant, Noah & Guajardo-Cespedes, Mario & Yuan, Steve & Tar, Chris & Sung, Yun-Hsuan & Strope, Brian & Kurzweil, Ray.. Universal Sentence Encoder .