# Deep Hierarchical Encoder For Detecting Incongruity Between News Headline and Body Text

A Dissertation submitted in partial fulfillment of the
Requirements for the Degree of

**MASTER OF TECHNOLOGY**
**IN**
**COMPUTER ENGINEERING**


Submitted by:
**MOHD AMZAD-(19MCS010)**

Under the supervision of
**Prof. Bashir Alam**
(Prof. Department of Computer Engineering)




Department of Computer Engineering
Faculty of Engineering and Technology
JAMIA MILLIA ISLAMIA
New Delhi-110025

November-2022

# CERTIFICATE

This is to certify that the Dissertation work entitled **"Deep Hierarchical Encoder For Detecting Incongruity Between News Headline and Body Text"** submitted by **MOHD AMZAD(19MCS010)** for the partial fulfillment of the requirement for the award of the degree of **Masters Of Technology** in **Computer Engineering** is a bonafide record of project work carried out by him under my supervision at the Department of Computer Engineering, F/O Engineering And Technology, Jamia Millia Islamia, New Delhi.

The contents of this report have not been submitted to any other institution or university for the award of any degree, in whole or in part.

.....
Prof. Bashir Alam
*Supervisor*
Dept. of ComputerEngineering
F/O Engineering And Technology
Jamia Millia Islamia

New Delhi
November 2022

.....
Counter Signature Of HOD with seal
Prof. Bashir Alam
*H.O.D*
Dept. Computer Engineering
F/O Engineering And Technology
Jamia Millia Islamia

# DECLARATION

I hereby declare that the work presented in this report entitled **"Deep Hierarchical Encoder For Detecting Incongruity Between News Headline and Body Text"** submitted by me for partial fulfillment of the requirement for the degree of **M.Tech In (Computer Engineering)** is a record of original work carried out by me under the supervision of **Prof. Bashir Alam,** and has not formed the basis for the award of any other degree, in this or any other institution or university. Wherever the findings of others have been mentioned, due recognition has been made in accordance with the ethical practices of reporting scientific information.

MOHD AMZAD
(19MCS010)
Dept Of Computer Engineering
F/O Engineering And Technology
Jamia Millia Islamia

New Delhi
November 2022

# ACKNOWLEDGEMENTS

First of all, I want to express my gratitude to the Almighty for giving me the skills and tenacity required to do this work.
This project necessitated a great deal of time, effort, and investigation. Its implementation would not have been possible unless I had the help of a significant number of people. As a result, I'd like to express my heartfelt gratitude to them all. I want to express my sincere gratitude to Prof. Bashir Alam, who serves as my supervisor and HOD, for giving me the project direction and execution I needed. Without his guidance, encouragement, and support, this work would not have been possible. I successfully overcame my challenges with his instruction and gained a lot of knowledge. Despite having a busy schedule, he still took the time to examine my work progress, offer helpful recommendations, and offer improvements.

I am also thankful to Prof. Tanveer Ahmed, Dr. Mohd Amjad, Dr. Sarfaraz Masood, Dr. Danish Raza Rizvi, and various other professors and assistant professors of the department for offering the resources and infrastructure I needed to complete my research. I want to thank the entire faculty at the computer engineering department.
My Brothers: Thank you for supporting me during these few years and allowing me to pursue my goals.

Name: MOHD AMZAD
Roll No:19MCS010
Email: scientistamzad786@gmail.com

# ABSTRACT

In the past few years, the user's reliability of the information on the internet has increased drastically. Social media platforms are being used mostly for information gathering and news access. Social Network Sites (SNSs) allow users to freely share information over the internet and millions of users around the globe access that information. As a consequence, a lot of misinformation and hoaxes also spread over the SNS. And it's practically impossible to detect such misinformation or fake news over social sites timely. So, it highlights the need for an automatic fake news detection system.

As information sharing over social sites is low-cost and easy to access therefore millions of people around the globe access news over these social platforms. In order to assist users in consuming accurate news content, it has become vital to identifying news headlines that intentionally mislead readers with exaggerated or misleading information in advance. Previous research on headline incongruence has mostly concentrated on elements taken directly from the headlines, which has a negative impact on performance since headline and news body text consistency is not given enough consideration.

In order to identify news items with misleading titles, this study uses a large-scale pair of news headline and body text datasets with incongruity labels. We build two hierarchically organized neural networks on this dataset to generate a complicated textual representation of news items and analyze the mismatches between the title{Headline} and the message body{Body Text}.

Our tests and qualitative assessments reveal that the suggested techniques work better than current methods in identifying news stories on social media with incongruent headlines.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ACRONYM

| | |
|---|---|
| IT | Independent Training |
| NLP | Natural Language Processing |
| IDF | Inverse Document Frequency |
| SVM | Support Vector Machine |
| ANN | Artificial Neural Network |
| SMAN | Structure-aware Multi-head Attention Network |
| GHDE | Graph Hierarchical Dual Encoder |
| GNN | Graph Neural Network |
| AUROC | Area Under the Receiver Operating Characteristics |
| MuSeM | Mutual Attentive Semantic Matching |
| SeqGAN | Sequence Generation |
| RL | Reinforcement Learning |
| GAN | Generative Adversarial Networks |
| XGB | XGBoost |
| FNC | Fake News Challenge |
| RDE | Recurrent Dual Encoder |
| CDE | Convolution Dual Encoder |
| AHRDE | Attentive Hierarchical Recurrent Dual Encoder |
| ERE | Embedding Recurrent Encoder |
| GRU | Gated recurrent units |
| ReLU | Rectified Linear Unit |
| MLP | Multilayer Perceptron |
| DF | Document Frequencies |
| IDF | Inverse Document Frequencies |
| DA | Data Augmentation |

# Chapter 1

# Introduction

## 1.1 Motivation

With the fast expansion of the internet in recent years, several online news websites and social applications have developed appealing and catchy methods to offer news to their users. And it is very easy for users to access information over these social platforms as it is low-cost and available all the time. However, the growth of online journalism brings with it issues such as inaccuracy, unfairness, and subjectivity. These online news publishers make their headlines so catchy that it attracts users' attention and in most cases, they believe that it's real. But if readers tend to click on the links and want to read the full news article sometimes they find that the main information is missing. The facts mentioned in news articles are constantly exaggerated or distorted in misleading headlines. And they are left disappointed after reading the whole passage and discover that the passage speaks differently than its headline.

Misleading headlines and sharing false information in journalism have created many critical problems in the past and can create many serious issues in the future. As much fake news and false information are shared online to archive social, economic, or political agendas. Our society and institutions are at risk because a lot of the information shared on social media platforms is unreliable. Since the majority of information shared online is headline-based, it stands to reason that headlines are crucial in establishing readers' first impressions. Therefore, the viral potential of news stories on social media is determined by the headlines. People are less likely to read or click on the entire content in today's digital world due to information overload and instead prefer to read news headlines. Additionally, many pieces of information are shared on social networking sites by users who haven't necessarily read the entire news article. The first impression created by the headline, on the other hand, is so strong that it lasts long after reading the entire news article. Therefore, if a news

headline is incongruent with its body text it could mislead readers to access false information, which then becomes hard to revoke.

Previous research has looked at the linguistic structures of headlines (Blom and Hansen 2015; Chen, Conroy, and Rubin 2015) or the syntactic matches among article titles and paragraphs in an effort to find inconsistencies in news items (Blom and Hansen 2015; Chen, Conroy, and Rubin 2015).

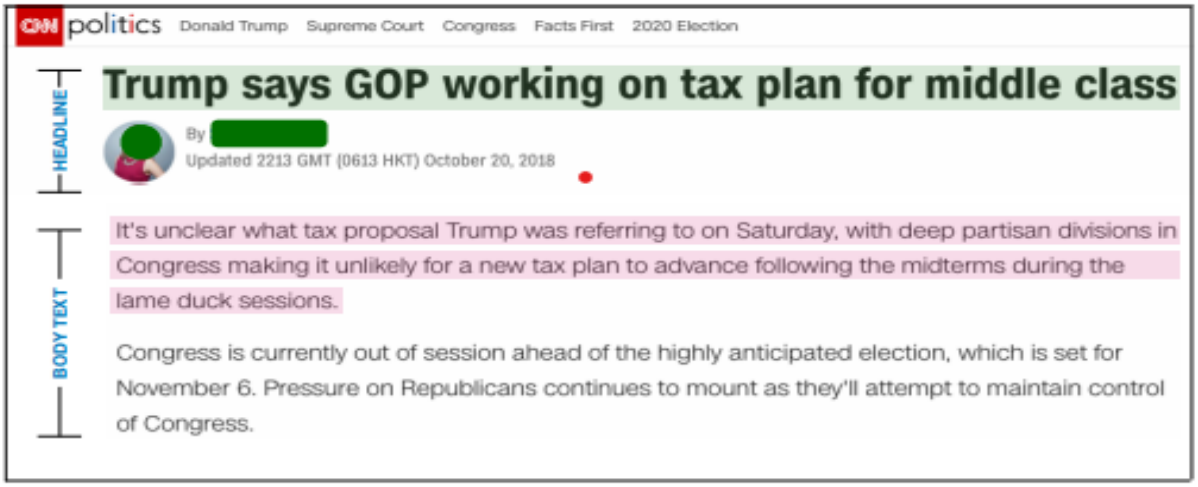




Figure 1.1 [1,2] : A news article example of incongruent headlines related to healthcare and politics [35 & 36].

Consider the headline in Figure 1.1,[1] from CNN as an example of an incongruent headline (cnn.com). It says in the headline, "Trump says GOP working on tax plan for the middle class," but the body actual content reads, "It's unclear what tax proposal Trump was referring to on Saturday," which is in opposition to the heading. In Figure 1.1,[2], the headline reads "The Scary New Science That Shows Milk Is Bad For You," however, the body text makes clear, "The study was small, to be sure, and it included no women." The study's claim is vastly generalized and exaggerated in the headline.

Therefore, it is important to recognize incongruent headlines in advance to help readers select which news stories to read and thereby minimize the possibility of misinformation.

## 1.2 Challenges

It is complicated for human beings to differentiate between real and fake news. Furthermore, catchy headlines are more likely to be believed by readers. Because the actual text is only seen after clicking, many individual users will just ignore the inconsistencies by going through the news titles. Content inconsistency is a growing issue that is changing how people read the news. And there are many more challenges faced to detect fake news. Concretely,

• On three different types of fake news, including satire, hoax, and propaganda, Rashkin compares the language used in false reports to that of accurate reports. Her research suggests that a variety of linguistic structures, such as subjectivity, intensification, and clarifying words used to provide ambiguous, confusing, emotional, or exciting language, have an impact on the production of fake news. Therefore, it will take a lot of time and effort to apply feature-based solutions. And from her research work, we get to know that complex language is used in fake news.

• The existing models concentrate on figuring out how a short title{Headline} relates to a long actual text{Body Text}, which can be tens of

thousands of long. This poses challenges for efficient neural network-based learning because of the excessively large lengths of news items.

• The data in fake news is limited, and the words chosen are determined by the agendas of the publishers. At the moment, only a dataset of political fake news has been released. Future research is still needed in fields other than politics. Additionally, it is challenging to train deep learning models to find headline inconsistencies, which involve a variety of parameters, in the absence of a large-scale dataset.

• Most online news users read only the headlines and they don't check the body of the news or don't go into detail about that news. So it's easy to spread fake news by making catchy headlines. That's why it becomes necessary to detect incongruity in advance between the headline and its body text.

## 1.3 Problem definition

When (Chesney et al. 2017) attempted to solve the Headline Incongruence Problem, they noticed that a headline contains statements that are irrelevant to one another or that are different from one another, with the stories dispersed across the actual text. They also identified inconsistency in news items as a major aspect of false news.
News headlines are grouped into three types from the perspective of journalism and communication [Marquez, 1980]. They are:

**Accurate Headlines:** Those headlines whose meaning is congruent with the content of the complete news story are called accurate headlines. Such headlines come under the category of non-clickbait.

**Ambiguous Headlines:** Headlines whose meaning is uncertain about the story's content are referred to as ambiguous headlines. And in ambiguous headlines, some key information is always missing. The reader's interest is raised by the lack of understanding, and they are encouraged to click.

**Misleading Headlines:** A headline that misleads readers is one whose meaning deviates from the story's actual content; such headlines frequently represent different aspects of the same story. The distinctions can be minor or significant. Exaggeration, distortion, and other common techniques are used to create an impression in the user that leads them to believe false information.

It takes time and effort for annotators to read through news headlines and body copy to figure out the exact nature of the issue.

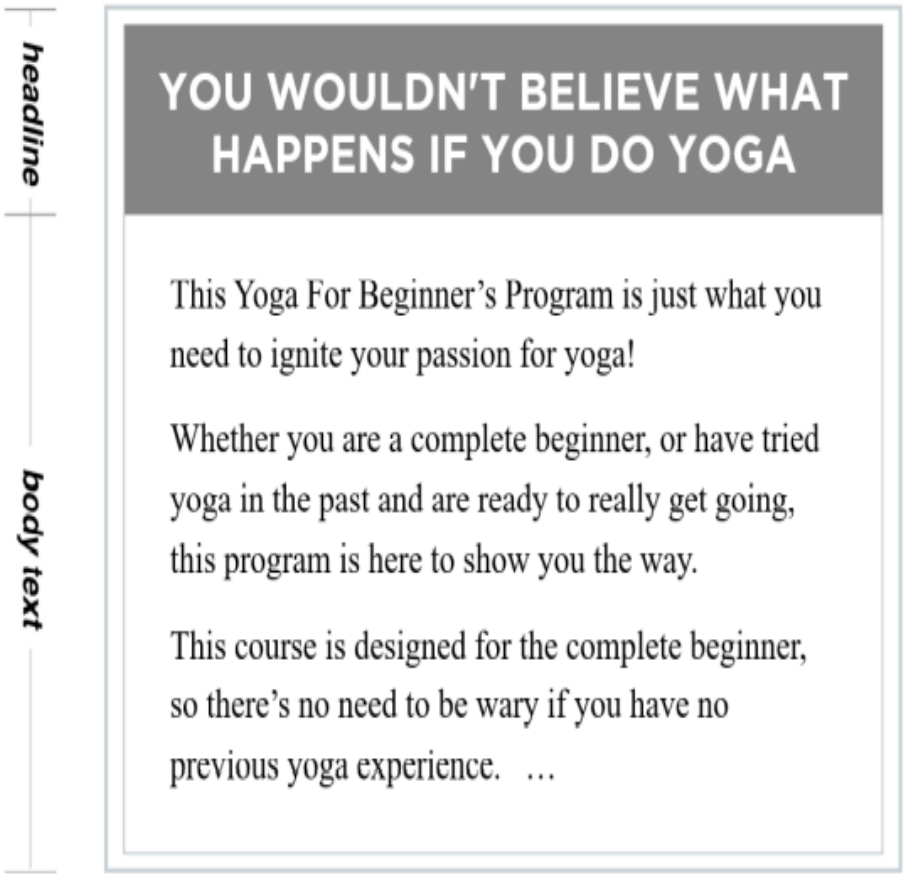



Figure 1.2: An example of an unrelated headline from a news story [6]

Figure 1.2 depicts a typical example of such misinformation, including both the headline and the body. This news article's headline promises to inform readers about the advantages of yoga, but the body text primarily serves as an advertisement for a new yoga program meant to convince readers to

sign up for the new yoga program. These misleading headlines not only give readers the wrong impression, and yet make things worse when they are circulated via social media, where most people share without examining the content. As a result, algorithmic methods for determining whether the headline and its body text are compatible must be developed.

Our main objective is to determine whether the news headline is consistent or inconsistent with the body text, or, quite challenging, to classify it into a particular level of integrity.

## 1.4 The method's introduction

In order to address the issue of conflicting news headlines, this dissertation used a large dataset made up of genuine news articles that were published over at least two years. The complex textual relationship between the news headline and the complete news content is learned using deep learning techniques, which are proposed and have proven to be essential for classifying headlines with inconsistencies. We build two hierarchical neural networks to simulate a complicated textual relationship of news stories and to measure the inconsistencies between the title{Headline} and the text{Body Text}. We also take into account some previously developed baseline techniques.

In order for the model to determine the significance of each paragraph in the {body text} based on the {headline} of the article, the attention mechanism has been added to the paragraph-level RNN. In order to use data from the past as well as the future, we also use composite RNNs at the paragraph level. Utilizing two hierarchical RNNs, the model encodes text from the word level to the paragraph level. Each of the news article's paragraphs, which make up the main body of the text, is embedded by averaging the word-embedding vectors of the words around it. Then, using the paragraph-encoded sequence input, a paragraph-level RNN is used to retrieve the final encoding vector for the entire {body text}. We initially calculate the mean of word vectors for each sentence to get the

incongruence score for each paragraph. RNN then uses the averaged word vectors as input to encode a sequence of sentences.

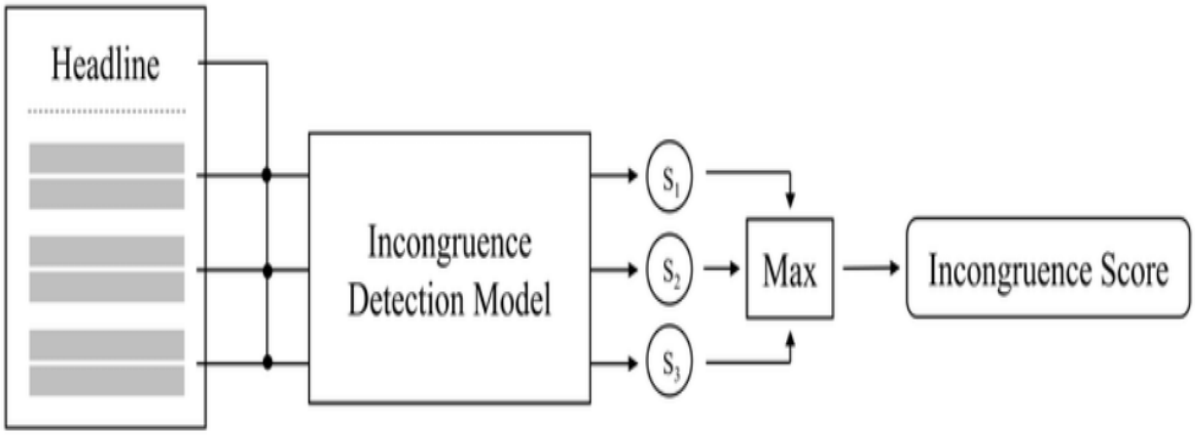



Figure 1.3: Basic model for headline incongruence Detection model.

Figure 1.3 shows the basic diagram of the headline incongruence detection model. The news article's body text is divided into several paragraphs, and then the headlines for each independent paragraph are compared. Between the headline and the text, it corresponds to, the incongruence detection model determines the score of incongruity. The headline and body text pair's incongruence score, which is known as the highest incongruence score, has the highest value of all values. Additionally, we present a data augmentation technique that significantly reduces the amount of text that a model must process by individually examining each paragraph of news articles. This method enhances performance and speeds up model training.

Because there are so many different baiting methods and we only have a tiny collection of labeled data, identifying false headlines is a difficult problem. We strive to make full use of the labeled data to develop a strong classifier that can handle various situations. The news in our dataset is crawled from numerous news websites that cover a wide range of topics such as sports, society, and world news.

## 1.5 Contributions

The following is a list of our contributions:

1. For the incongruent headline problem, we used a large dataset that includes nearly all of the news articles published in a country over a period of at least two years. The corpus consists of incongruity labels as well as pairings of news headlines and body content. The dataset consists of new articles from different fields such as politics, sports, economy, entertainment, etc.

2. We propose deep hierarchical architectures that encrypt the whole news story at the lexical level before proceeding up to the phrase level in order to compare each independent paragraph with the story's title. And we also show how to break news paragraphs and annotate each one separately using a data augmentation strategy. This strategy not only minimizes the amount of data a model has to deal with, but also improves the number of training instances, boosting performance even more. The data augmentation method that we developed here is the Independent Training(IT) method.

3. We have designed the model with fewer parameters. Neural networks with fewer parameters are fast and they don't face the problem of overfitting.

4. We extensively evaluate our models with real data. Our dataset's efficacy in training incongruent headlines was effectively demonstrated by manual verification. Furthermore, according to a crowdsourcing experiment, the perceived amount of incongruence for specific news topics (e.g., politics) may differ depending on individual attitudes and media providers.

5. To study and compare the results obtained with many other baseline models.

# Chapter 2

# Literature Survey

## 2.1 Deep learning methods

### 2.1.1 Networks of neurons

Artificial neural networks also referred to as neural networks or feedforward networks are algorithms that aim to imitate the structure and operations of the human brain. They are modeled after biological neuronal networks. A neuron will take input from its nerve cells or dendritic tree, and if the input is strong enough, it will move through a cell body and connect to a node from the other neuron. Synaptic gaps separate nerve cells and they only couple when the synaptic connection between the axons of one neuron and the dendrites of the other neuron is activated.

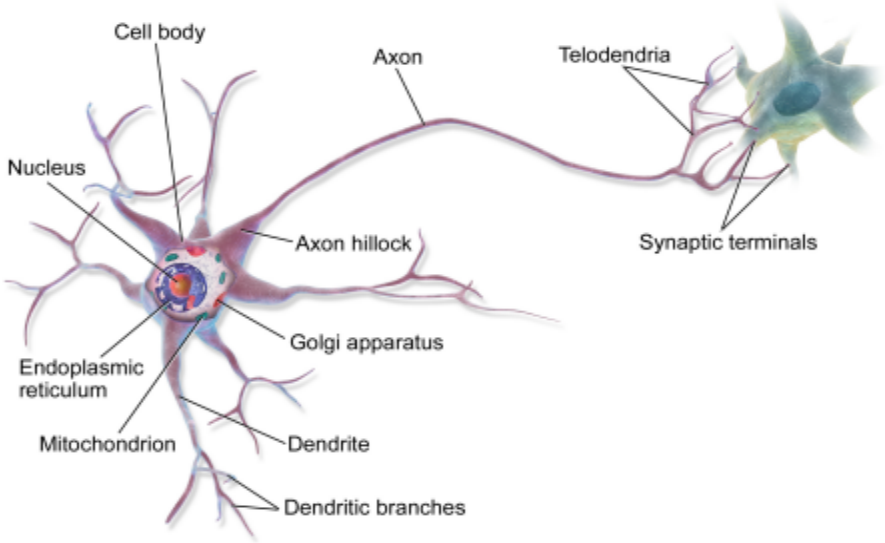



Figure 2.1: Network of biological neurons [33].

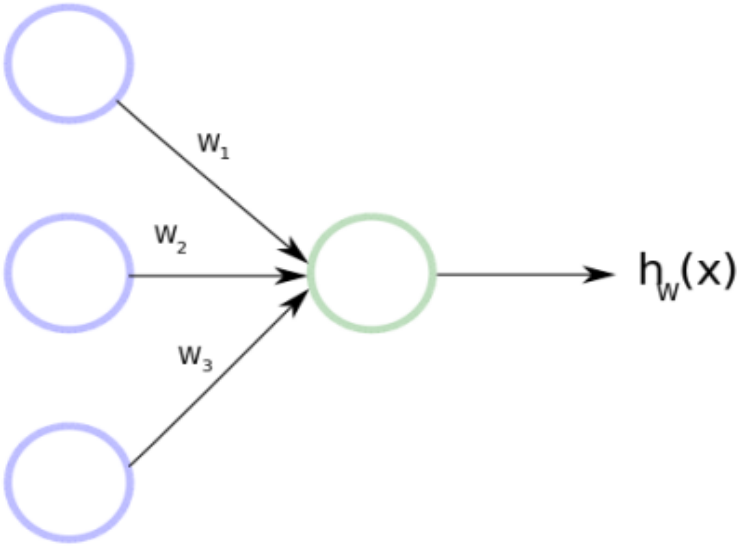



Figure 2.2: Shows a simple anns network with a single output unit [33].

**Illustrations in a Model:**

Weight $W_1$ is multiplied by a binary input $x_1 \in (0, 1)$ representing whether a neuron is triggered or not. $W_1$ represents the strength of the connection; it is larger if the connection is strong and smaller if the connection is weak. This section is used to model a synaptic connection between two neurons. In other words, it depicts how synaptic connections affect the choice of whether to stimulate or not to activate the axon. The same goes for $x_2$, $x_3$,..., $x_n$, which are multiplied by $W_2$, $W_3$,..., $W_n$, respectively. To show the combined influence of those inputs, all of the products are then added together to form a single unit. The axon is stimulated if the activation function's output is larger than 0. A neural network with a logistic activation function is shown in Figure 2.2. The activation $h_W(x)$ in this example is calculated as follows:

$$h_\Theta(x) = g(W^T x) \quad \text{.... (2.1)}$$

where the logarithmic activation function g(z) is calculated as follows:

$$g(z) = 1/( 1 + e^{-z} ) \quad \text{....(2.2)}$$

**Multiple-label classification using a neural network:**

An effective solution for a 3-label classification task could be a neural network like the one in Figure 2.3. A three-dimensional one-hot vector is the output of the $h_W$ vector.

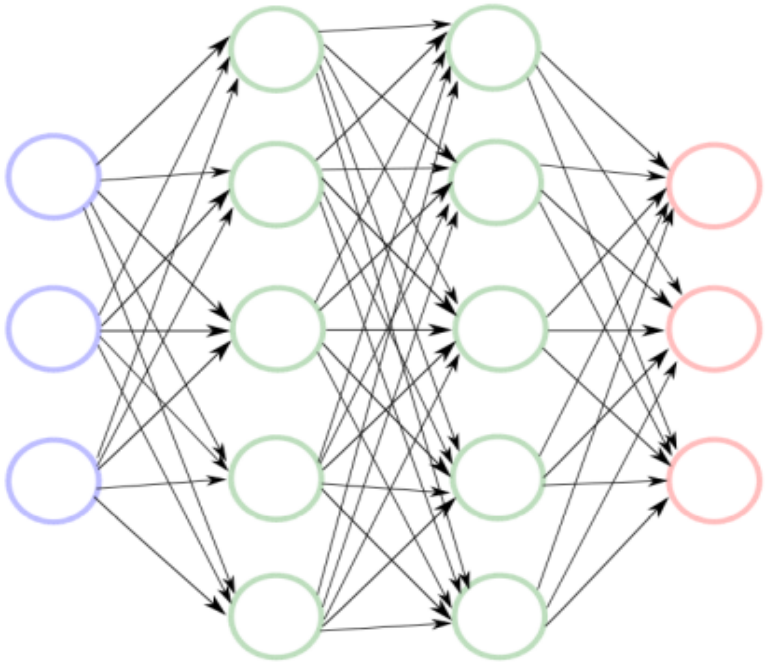


Figure 2.3: Diagram of a two-hidden-layer neural network for three labels[33].

**Squared Error:**

The error function is the deviation between the model's predicted output $\hat{y}$ and the ground truth y. Using the difference between them, or norm 1, a naive technique can be used:

$$L = |y - \hat{y}| \quad \text{....(2.3)}$$

Square error, or norm 2, is used for mathematical simplicity when taking derivatives as:

$$L = (y - \hat{y})^2 \quad \text{....(2.4)}$$

**Cross-Entropy**:

The following is the definition of cross-entropy among two separate distributions:

$$H = -\sum_{i=1}^{C} y_i \log(\hat{y_i}) \quad \text{....(2.5)}$$

### 2.1.2 Recurrent Neural Networks (RNN)

RNNs are a particular kind of neural network in which a cell, or subnetwork, is used to read a variety of inputs. Figure 2.4 illustrates the repetitive structure.

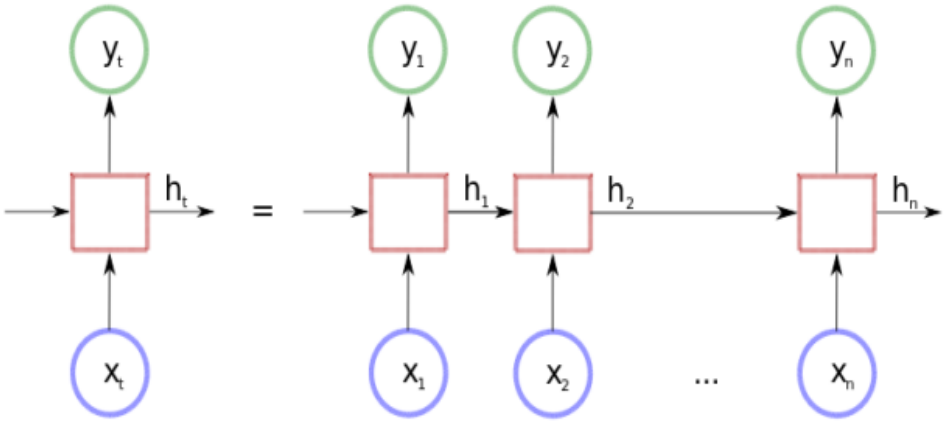



Figure 2.4: A recurrent neural network that has been unrolled [33].

The new hidden state and output at time step t are calculated using input $x_t$ and the hidden state from the previous step, $h_{t-1}$:

$$h_t = \sigma_h(W_h x_t + U_h h_{t-1} + b_h) \quad \ldots.(2.6)$$

$$y_t = \sigma_y(W_y h_t + b_y) \quad \ldots.(2.7)$$

where:

- The input vector would be $x_t$ at time step t, the convolution layers vector is $h_t$ for the hidden layer, and the final output vector is $y_t$.
- Parameter vectors and matrices are W, U, and b.
- The activation functions $\sigma_h$ and $\sigma_y$.

This network is particularly well-suited to handle sequential data, where inputs are broken up into tiny chunks and then sent one by one into network cells. Since RNNs are made to replicate and focus on the sequence characteristics of specific types of data, it has been demonstrated that they struggle to capture long dependencies. In order to

get around these restrictions, the Long Short-Term Memory Network (LSTM), a modified RNN with gating mechanisms, was created.

### 2.1.3 Long Short-Term Memory Networks (LSTMs)

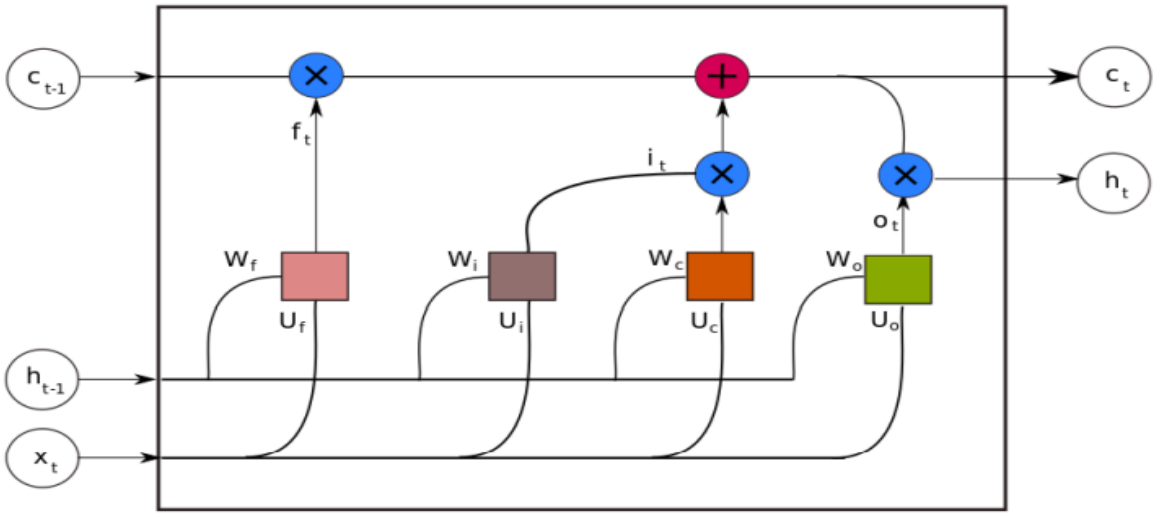



Figure 2.5: Long-Short Memory Networks' internal structure [33].

To overcome the memory limitation of regular RNNs, three gates are added to each network cell. When the cell reads inputs at each time step, a memory is preserved and updated. Figure 2.5 depicts four LSTM gates: input (I), output gate (o), forget (f) and memory (c).

From an old memory $C_{t-1}$, the new cell memory $C_t$ is calculated as follows:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C_t} \quad \ldots.(2.8)$$

**Forget gate:** This gate determines how much previous data will be forgotten and how much will be used in subsequent steps. When $x_t$ step size t is entered, it is calculated as

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad \ldots.(2.9)$$

After that, this $f_t$ is multiplied by $C_{t-1}$ to change it and delete some information.

**Memory gate:** establishes a fresh candidate memory. It is calculated as follows given an input of $x_t$:

$$\tilde{C_t} = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad \ldots.(2.10)$$

**Input gate:** The volume of data injected into the updated memory by this gate is managed from the candidate memory. It is calculated in the manner shown below given an input $x_t$:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad \ldots.(2.11)$$

The newly added memory in the new memory cell is then obtained by multiplying $\tilde{C_t}$ by $i_t$.

**Output gate:** identifies the volume of extracted cell memory. As follows is the calculation:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad \ldots.(2.12)$$

Then, the following information is updated for the new hidden state:

$$h_t = o_t * \sigma_c(C_t) \quad \ldots.(2.13)$$

The existence of internal memory and its capacity for sequential updates are said to solve the issue of long dependencies.

## 2.2 Related Work on Headline-Based Fake News Detection

I**n 2017 Chesney et al**, published a paper titled "Incongruent Headlines: Yet Another Way to Mislead Your Readers" that addressed the issue of incongruent headlines, which are headlines that do not adequately summarise the information in the article in which they are found. They suggest that this issue should be dealt with differently from other known harmful headline styles, like bait and exaggeration. According to them, the identification of headline inconsistencies cannot be achieved with existing natural language processing (NLP) approaches applied to these connected ideas, necessitating a more thorough examination than only visual considerations. As a foundation for future study in this field, they propose a variety of different techniques that may be appropriate for the task at hand.

Based on previous work in other fields, they suggest unique incongruence detection methods that will investigate many facets of the issue. The best way to detect headline inconsistencies is to go about it in stages. For example, it can be very difficult to analyze complex relationships between a title and a full news story because of how different their lengths and degrees of linguistic complexity are.

They emphasize that methodology applied to related concepts like clickbait and sensationalism cannot be used to approach headline incongruence because they use headline-specific stylometric features and ignore any deeper semantic relation between headline and text that would be critical to the task at hand. Since they have experience with summary and headline generation, stance identification, claim, and quotation extraction, and argument analysis, they offer a number of potential approaches for completing this task.

**In 2017 Peter, Julian, et al**, presented a paper "From Clickbait to Fake News Detection: An Approach based on Detecting the Stance of Headlines to Articles". They outline a method for figuring out where headlines fall in relation to the main body of the articles they are related to. The method can be used in fake news identification, particularly in clickbait detection circumstances. They use a process to determine if a certain headline/article combination is connected or unrelated, as per the challenge's scoring system.

The CoreNLP Lemmatizer is used to do n-gram matching of the lemmatized input (headline or article). The pair is assumed to be connected if the resulting score is greater than a certain threshold (they chose 0.0096 as the ideal value). Consider two arrays representing a title and a text (H and A) when $n \in [1,6]$, among all available lemmatized n-grams, with h(i) and a(i) containing each array's I'th member, len() being a function that calculates the length in tokens of a string (n-gram), $TF_T^k$ showing how frequently term k appears in the range T, and $IDF^k$ indicating that term k appears in texts with a negative frequency across all paragraphs.

$$SC = (\sum_{i=1}^{len(H)} TF^{h(i)} * IDF^{h(i)})/ (len(H) + len(A)) \quad ....(2.14)$$

where

$$TF^{h(i)} = \{(TF_H^{h(i)} + TF_A^{H(i)}) * len(h(i))\} \quad ....(2.15)$$

Table 2.1 demonstrates that the vast majority of "related" instances fall under the class "discuss," and assigning this class to all instances of "relatedness" yields an accuracy of 61.51 (with this portion of the data set), which is displayed in the "Majority vote" column. They additionally employ Mallet's Logical Regression classifier solution (McCallum, 2002), and received training just on titles, in order to build on this baseline and further classify the linked pairs into "agree," "disagree," or "discuss". And the weighted score was 79.82 (column "3-class classifier").

Table 2.1: Key figures of the FNC-1 data set [34].

| | | |
|---|---|---|
| Unique headlines | 1.648 | |
| Unique articles | 1.668 | |
| Annotated pairs | 49.972 | 100% |
| Class: unrelated | 36.545 | 73% |
| Class: discuss | 8.909 | 18% |
| Class: agree | 3.678 | 7% |
| Class: disagree | 840 | 2% |

They used three binary classifiers to choose between the top and second-best scoring classes if the difference was less than the threshold (one binary classifier for "agree"-"disagree," one for "agree"-"discuss," and one for "discuss"-"disagree"). Both the headline and the article are used to train these classifiers. Table 2.2 shows the results in the column "Combined classifiers."

Table 2.2: Results of 50-fold cross-validation [34].

| | Majority vote | 3-class classifier | Combined classifiers |
|---|---|---|---|
| Relatedness score | 93.27 | 93.26 | 93.29 |
| Three-class score | 61.51 | 75.34 | 88.36 |
| Weighted score | 69.45 | 79.82 | **89.59** |

For the binary classification of "related" vs. "unrelated" headline/article pairs, their system is based on simple, lemmatization-based n-gram matching. They receive a (valued) overall accuracy of 89.59 on a publicly available data set that has been labeled for the positioning of titles in relation to their corresponding article bodies.

**In 2021 Normala, Iskandar, Fatimah, et al,** published the paper " Fakeheader: A Tool to Detect Deceptive Online News Based on Misleading News Headlines and Contents ". Their suggested work develops a method to identify fake news based on deceptive headlines or content. A Support Vector Machine and a suggested feature combination are used in the software's data veracity framework for online news. The suggested technique is a novel set of feature combinations. To create high-accuracy prediction tools, it is proposed to combine Bigram and Lemmatization characteristics with Base (TFIDF), Syntactic, Bigrams (N-Grams), and Punctuation features (precision, recall, and F-score). In Figure 2.6, the suggested method for identifying false news based on deceptive headlines is depicted.

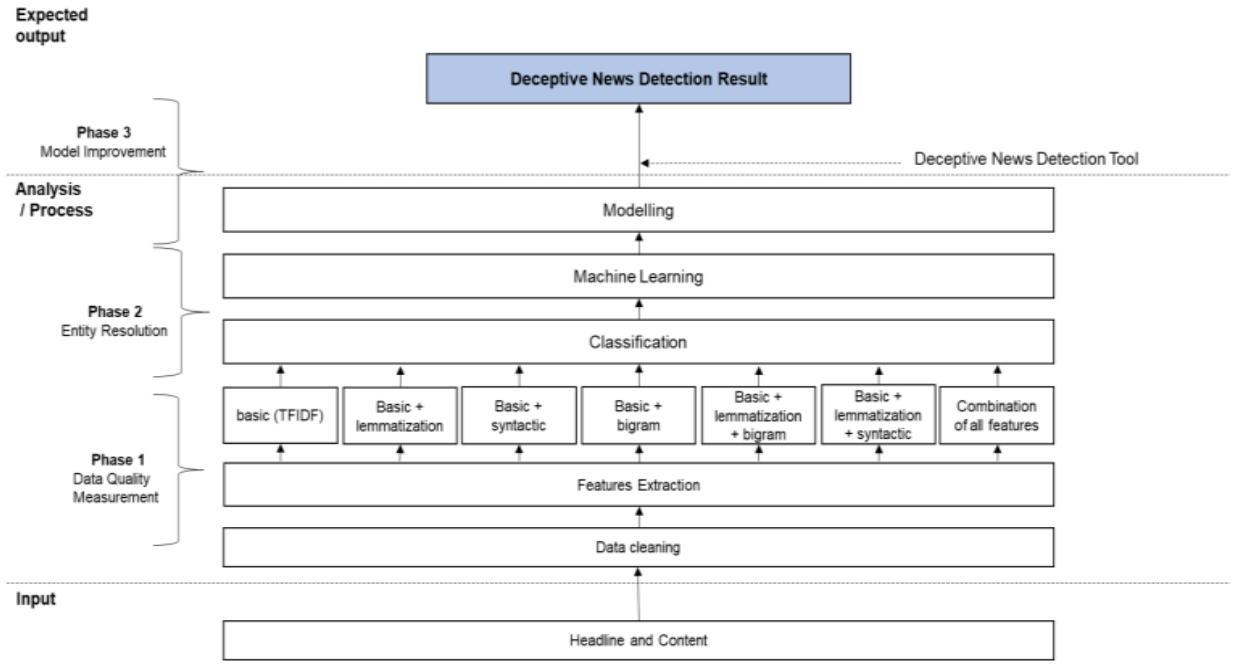



Figure 2.6. Data Veracity Architecture for Detecting False News Based on False Headlines [5].

The SVM classifier was determined to be the preferred model based on the training results, with the best accuracy in the majority of the features used, and more than 90% efficiency recorded for all kinds of data sets examined. Over the headline database, SVM was the model that performed better than the rest in terms of efficiency in identifying fake news for all parameters taken into account. With the exception of Base + Syntactic, SVM had the best accuracy over the entire dataset. For Base + Syntactic, ANN had the highest accuracy.

Their proposed methods and dataset demonstrates that the suggested algorithms when combined with aforesaid attributes, achieved good prediction precision on the headline dataset, which is composed primarily of short texts. Similar features and Recalls and F-Scores were among the greatest precision features in the Content dataset (without title). A prototype tool called FakeHeader can be used to identify fake news based on its headlines, contents, or a combination of the two. Users can enter the news story's headline, content, or entire body into the text area to obtain a prediction and accuracy of the false news detection result.

The suggested method uses a Support Vector Machines classifier and incorporates a unique data veracity framework to determine the accuracy of internet news based on news headlines. In the testing, the suggested method with the suggested combination of attributes had good success in identifying false news based on headlines, contents, and combined news data. According to the testing outcomes, the suggested tool was able to produce high-performance results with precisions and recalls of over 90%.

**In 2020 Chunyuan Yuan, Qianwen Ma et al**, proposed a paper "Early Detection of Fake News by Utilizing the Credibility of News, Publishers, and Users Based on Weakly Supervised Learning". By combining the news content, publishing, and reposting relations of publishers and readers, they suggest a special Structure-aware Multi-head Attention Network (SMAN) in their research to jointly optimize the fake news detection and credibility prediction tasks. They propose a special multi-head attention network that is structure-aware for the early detection of fake news. They start by

developing a structure-aware multi-head attention module that builds publisher representations for predicting publisher credibility and learns the structure of the publishing graph. In order to encode the news diffusion graph among users and build user representations for user credibility prediction, the structure-aware multi-head attention module is next used. In order to detect fake news early, they use a convolutional neural network to convert the news text from word embedding to semantic space. After that, they use the fusion attention module to combine the representations of the publisher, user, and news.

They begin by assessing the influence of the credibility prediction subtasks provided by the publishers. They've noticed that without the PC, performance suffers greatly. By transferring the influence of publishers' credibility to the credibility of the news, the credibility prediction subtask for publishers can use publishing relationships between publishers and matching news to make it simpler to identify fake news. A publisher's reputation must be expressly encoded, as shown by the ablation results. They then look at how the user credibility prediction subtask affects the results. Additionally, keep in mind that the removal of UC causes significant performance drops across all datasets. It makes sense that a news story's credibility would be significantly damaged if it were widely shared by people with a poor reputation. A diffusion graph, similar to how it can be done with PC jobs, can be used to translate user credibility to news credibility, improving detection performance. Finally, they discover that after deleting both the publisher and user credibility prediction subtasks, the performance is much lower than the whole model SMAN, demonstrating that both jobs contribute complementary information. In order to effectively combat fake news, it is essential to combine the tasks of identifying it and determining its veracity.

SMAN was tested on three different data sets, and the research shows that it can identify misinformation with 91% correctness in 4 hours, which is a lot quicker than the techniques currently in use.

**In Jan 2021 Seunghyun Yoon, Kunwoo Park et al** released a paper titled 'Learning to Detect Inconsistency in News Headline and Body Text through a Graph Neural Network'. This study focuses on the issue of identifying article title content discrepancies, in which the headline story is totally pointless or perhaps even contradictory to a language within the body. To effectively learn content similarity between news headlines and lengthy body paragraphs, the graph-based hierarchical dual encoder (GHDE) model uses a graph neural network. Between the headline and paragraph nodes, the GHDE learns to compute weight values and assigns a higher edge weight to the edges that are more pertinent. Then, by aggregating information from surrounding nodes, GHDE updates each node representation. The iterative update mechanism propagates relevant data from paragraph nodes to the headline node, which is critical for detecting content inconsistencies.

In a graph neural network (GNN), data is represented explicitly or implicitly using information about the structure of a graph. Network information can be embedded using a variety of approaches. They create two distinct datasets: one with "sample" news items picked at random (i.e., Random set) and another containing "sample" articles picked to include similar news stories (i.e., Similar set). The "target" news article is then updated with passages again from "sample" media articles. Depending on how complex the incongruity is, the number of swapped paragraphs can vary from 1 to the number of selected sample paragraphs. After that, the remaining news pool is sampled for an equal number of articles to include relevant information (i.e., those with a "negative" label). Headline similarity is calculated using the Euclidean distance of the fastText word embedding. And, to control the incongruity difficulty of the generated dataset, set a maximum threshold for the similarity measure.

Their objective is to ascertain whether a news headline conflicts with any section of the body text. They conceptualize the detection task as a tuple (H, P), where H denotes the headline and P denotes the body text's collection of paragraphs. In each paragraph $p_i \in P$ is a word series that may or may not include a sentence. They want to locate a bitwise

inconsistency label called y. They start by going over the proposed learning approaches for identifying headline discrepancies. Finally, they provide a label neural network algorithm that takes into consideration the related data in between title{Headline} and the text{Body Text} that it corresponds to.

The newest model, AHDE, makes use of a hierarchical framework to handle large news articles and leads to inconsistencies at the end of each paragraph. They use a hierarchy system, considering both headings as well as the text's body as separate units for analysis. Additionally, they employ graph-based learning to identify inconsistencies by studying the significance of each paragraph from beginning to end. The suggested model, a graph-based hierarchical dual encoder (GHDE), calculates the likelihood that a news article's headline will be inconsistent in four steps. Every news item as well as text is represented as a node using a multilayer RNN framework. By calculating a matching score for each pair of paragraph endpoints, the edge weight for every sentence endpoint is established. The graph is finished using the steps above, and then the graph neural network propagates data between nodes to look at any inconsistencies in the article. In the final step, the recent data from each node is combined, and inconsistencies are predicted.

The research findings demonstrate that in the area under the receiver operating characteristic (AUROC) curve, the present graph-based neural network model greatly improves the prior form models (5.3 percent). The trained model can recognize inconsistent headlines, according to tests on recent news articles.

**In 2020 Rahul Mishra, Piyush Yadav, et al,** published a paper 'MuSeM: Detecting Incongruent News Headlines using Mutual Attentive Semantic Matching'. This study proposes a strategy for a multi-attention-based system that includes the original as well as synthetic headings that benefit from the variations in all pairs of word embeddings of the relevant terms. Additionally, the paper examines two additional applications of the new

methodology that make use of word embeddings of authentic and artificial headline words through the use of concatenation and dot products.
They determine whether a news item with a headline and body content of $n_i \in N$ is incongruent by classifying it as "Congruent(C)" or "Incongruent(I)."

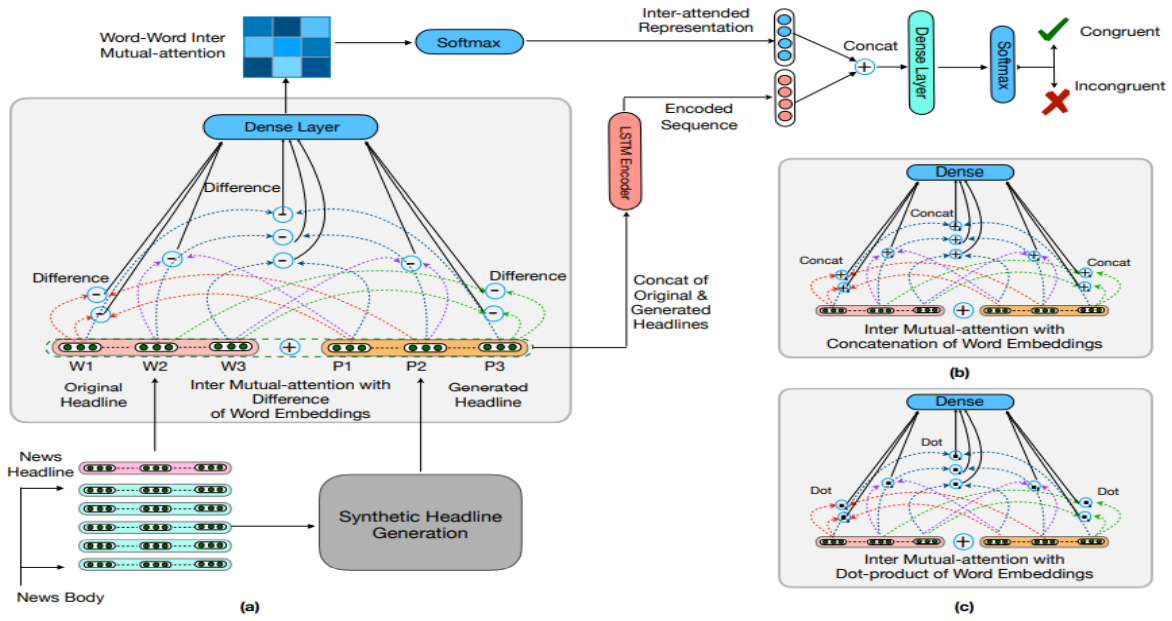



Figure 2.7: Overall Architecture of the MuSeM model [27].

Two neural networks, a discriminator, and a generator make up a standard generative adversarial network. Each of these semi-networks is in opposition to the other. In contrast to the generator network, which seeks to maximize the performance of the classifier, the discriminator network strives to reduce it. The classifier and the generator both converge and attain an equilibrium condition.
SeqGAN is a generative adversarial network-based sequence generation approach. Regular GANs and SeqGANs differ primarily in that SeqGAN directly implements a gradient policy update using a reinforcement learning(RL) based approach, which allows SeqGAN to avoid the issue of generator differentiation that afflicts original GANs. They investigate and test two alternative MuSeM variations that employ the concatenation of word embedding pairs and the dot product of applicant word embedding

pairs. They discover through their experiments that the concatenation and dot-product variants fall short of the difference-oriented variant in terms of performance. Additionally, they combine all three operations, namely distinction, scalar product, and concatenation, which outperform the model that only focuses on differences gently.

All MuSeM model versions outperform all baseline approaches, with the MuSeM dpc SHG design outperforming all others with scores of 0.735 and 0.747 in Macro F1 and AUC, respectively.

**In 2017 Xiaojun Wan, et al,** published a paper 'Learning to Identify Ambiguous and Misleading News Headlines'. In their study, they clearly define the issue and list each unclear or deceptive headline. When detecting confusing headlines, they use class sequential rules to exploit structure information. Additionally, they identify false headlines by features extracted based on the consistency between headlines and bodies. They also employ a co-training strategy to make use of the big unlabeled data set and improve performance. They classify material crawled from four major Chinese news websites using the final classifiers, which span a variety of topics including sports, sociology, and foreign news. They begin by extracting features from headlines using the prior method. As a result, they mine class sequential rules (CSR) and then generate CSR features as a second step. Finally, to train an SVM classifier, both the basic and CSR features are used. They first identify linguistic patterns in both ambiguous and clear headlines using a frequent pattern mining method, after which they create features based on sequential ideas.

Their task precisely satisfies the requirement of co-training because they extract both body-independent and body-dependent features from every news item. To maintain the class distribution in the labeled data in their experiments, the component values p and n are matched at each iteration. Two sub-classifiers gain useful information from each other by adding confidently predicted instances. Both sub-classifiers will provide prediction scores for each instance during co-training. Normalized prediction scores

are [0, 1]. Finally, each instance's overall prediction score is calculated using the average of the normalized values.

The sub-classifiers employ the same SVM-based machine learning algorithm as the main classifiers.

Table 2.3: Comparison of results of supervised method and co-training [26].

| | Precision | Recall | F-score |
|---|---|---|---|
| Body-dependent | 0.602 | 0.660 | 0.630 |
| Body-independent | 0.637 | 0.716 | 0.674 |
| All Features | 0.646 | 0.768 | 0.702 |
| Co-training | 0.670 | 0.788 | 0.724 |

The co-training strategy is used in the later job to make use of the bigger unlabeled data set. The identifier's robustness is demonstrated by the experimental results. They find confusing and misleading headlines in the entire data set by using the final classifiers.

# Chapter 3

# Methodology

Our goal is to determine whether a news story must have an incongruent title for a set of a title{Headline} and a text{Body Text}. In our dissertation, we, therefore, calculate the output probabilities as an incongruence level.

## 3.1 Baseline approaches

We outline some established strategies that have been applied to address the headline incongruence issue. Due to their efficiency and simplicity, feature-based ensemble techniques are frequently used. The XGBoost algorithm has proven to perform better than other techniques in a variety of prediction tasks (Chen and Guestrin 2016).

First, we applied the XGBoost (XGB) classifier using the attributes provided in the FNC-1 competition's winning model, such as the cosine similarity between the {headline} and {body text}.

### 3.1.1 XGBoost

Gradient-boosted decision trees are applied by the well-known and quick algorithm XGBoost, which is used for classification techniques (Chen and Guestrin 2016). Because it was incorporated into the winning model for the stance detection challenge in news headlines, we chose XGBoost as a representative baseline.
Extreme gradient boosting, sometimes referred to as XGBoost, is a popular gradient-boosting methodology that boosts the speed and accuracy of tree-based (complex decision tree) deep learning strategies. We obtained a feature set consisting of TF-IDF vectors based on word occurrences and then implemented the winning model from this challenge by substituting an

incongruity label. Word-vector similarities between a {headline} and its corresponding {body text} are indicated by singular values decomposed from these vectors. We designate this model as XGB.

### 3.1.2 Recurrent Dual Encoder (RDE)

To compute the similarity between two text inputs, a recurrent dual encoder composed of dual RNNs was used (Lowe et al. 2015). A word-embedding layer, which transforms a word index into a vector, is applied to each word as it is passed through an RNN during word encoding. After the encoding step, the likelihood of having an inconsistent headline is determined using the last hidden state of each headline and body text RNN. The incongruence score in the training approach is as follows:

$$P(lb) = \sigma((hd^{H}_{th})^{T} M\, hd^{B}_{tb} + b),$$

$$Z = -\log \boldsymbol{\Pi}^{N}_{n=1} p(lb_n | hd^{H}_{n,th}, hd^{B}_{n,tb}) \quad \ldots.(3.1)$$

where the final hidden states of each {headline} and {body text} RNN with the dimensionality $hd \in R^{d}$ are $hd^{H}_{th}$ and $hd^{B}_{tb}$ respectively. The learned hyperparameters are bias b and $M \in R^{d*d}$. N represents the total number of training samples and σ is the sigmoid function.

### 3.1.3 Convolution Dual Encoder (CDE)

A limited set of words is used as context in the CDE method, which is based on CNN. The model calculates the maximum value of the local features using filters of different sizes. To check whether the article's headlines are incongruent, CDE extracts one of this method's most helpful characteristics. To solve the headline incongruence issue, we use Convolutional Dual Encoder in accordance with CNN's text understanding architecture (Kim 2014). We computed convolution with k filters using the headline and body text word sequences as input to the convolutional layer, as shown below, and then for each section of the text, derived a vector set $V = \{V_i | i = 1, \cdot \cdot \cdot, k\}$:

$$V_i = G(f_i(W)) \quad \ldots.(3.2)$$

in which G denotes the max pooling method, $f_i$ the i-th convolution filter for the CNN algorithm, and $W \in R^{t*d}$ for the word sequence matrix. We vectorize the text for the {headline} and {body} using dual CNNs.

## 3.2 Proposed methods

Previous research has shown that long word sequences in news articles reduce performance. The success of RNN would decrease as the length of word sequences in the text increased due to its inherent tendency to forget knowledge from long-range data. For instance, recalling details from the distant past is difficult for recurrent neural networks, which are used in RDE. The normal convolutional filter length of the model prevents it from detecting any relationship between words in different places even though CDE learns local word dependencies. Since news stories can be very long, one key issue of adopting typical deep approaches to solve the headlines incongruence challenge is their inability to handle extended segments. Our news sample has an average character count of 518.97.

As a result, we propose neural architectures that effectively learn hierarchical structures of large text sequences to fill this gap. Additionally, we present a data augmentation technique that efficiently lengthens the training set while reducing the length of the target content. Now we'll discuss the model we've proposed. The AHRDE architecture was created to solve this issue.

### 3.2.1 Attentive Hierarchical Recurrent Dual Encoder (AHRDE)

Using two-level hierarchies of RNN architecture, this model divides the texts into a list of segments and encodes the complete input text from the level of the word to the end of each paragraph. The model learns the relevance of each paragraph in the body text based on the article's headline by including an attention mechanism in the paragraph-level RNN.

In paragraph-level RNNs, we also use bi-directional RNNs to utilize information from the past and future. This model makes use of paragraph structure to address the news article's arbitrary length.

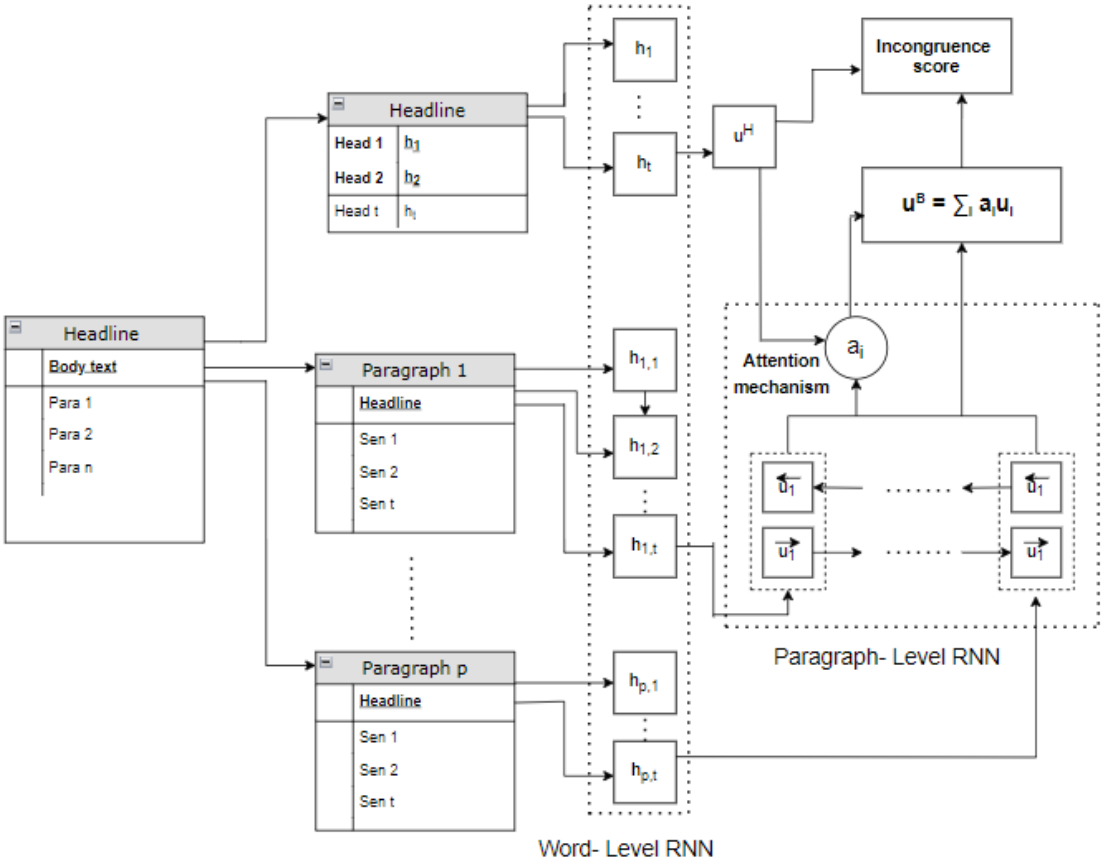



Figure 3.1: Diagram of AHRDE model.

A diagram of the AHRDE model is shown in Figure 3.1. All text input, from words to paragraphs, is encoded using a two-level hierarchy. Based on the article's headline, an attention mechanism can teach the model the significance of each paragraph in the body of the text. Each paragraph's word sequences, starting with $wd_p = \{wd_{p,1:t}\}$ and ending with $hd_p = \{hd_{p,1:t}\}$ are encrypted by the word-level RNN. The next-level RNN then features a series of paragraphs while maintaining order using the hidden states from the word-level RNN. With fewer sequential steps than RNNs, the

hierarchical design, as opposed to RDE, can learn the textual structures of news items. These hidden states are present in hierarchical RNNs:

$$hd_{p,t} = F_{\Theta}(hd_{p,t-1}, wd_{p,t}),$$
$$u_p = G_{\Theta}(u_{p-1}, hd_p) \quad \ldots.(3.3)$$

where $u_p$ is a hidden state of the paragraph-level RNN at the pth paragraph in the sequence and $hd_p$ is a final fully connected state of each sentence's term RNN. Due to the fact that each paragraph of the body text is matched to the title, each paragraph $u_p$ of the {body text} is grouped and classified in accordance with how closely it matches the {headline}.

$$S_p = V^T \tanh(WB^B_u \; u^B_p + WH^H_u \; u^H),$$
$$a_i = \exp(S_i)/\sum_p \exp(S_p),$$
$$u^B = \sum_i a_i u^B_i \quad \ldots.(3.4)$$

where $u^B_p$ stands for the p-th hidden value of the RNN at the phrase level used to learn the display of text in the bodies. The paragraph-level RNN with the most recent hidden state for {headline} is identified by the $u^H$. As with the RDE model, the same training objective is used to calculate the incongruence score as is shown below:

$$P(lb) = \sigma((u^H)^T M \, u^B + b) \quad \ldots.(3.5)$$

### 3.2.2 Embedding Recurrent Encoder (ERE)

The AHRDE system employs two layered RNNs to encode text at various levels, from individual sentences and words to entire passages. The model requires higher computational resources for training and inferencing as compared to conventional alternatives like RDE and CDE.

The primary {main text} of the news is broken into sections, and each one is integrated by extracting the typical vector representation from the words it contains. To put it another way, ERE figures out $hd_p$ in equation (3.3) such as taking the average of embeddings of words in subsection p,

$$hd_p = \sum_i \text{embedding}(wd_i),\ wd_i \subset \text{p-th subsection.}$$

Given an input sequence of length x = $\{x_{1:t}\}$ and these K vectors, the embedding procedure is built as follows:

$$p_k = \text{softmax}((x)^T m_k),$$
$$x_K = \sum_{k=1}^{K} p_k m_k,$$
$$e = \text{concat}\{x, x_k\} \quad \ldots.(3.6)$$

Dot-product is used to calculate the similarity of the x and each latent topic vector. The softmax function $\text{softmax}(z_k) = e^{zk}/\sum_i e^{zi}$ is then used to normalize the resulting K values, generating the similarity possibility $p_k$. After calculating the latent topic probability $p_k$, $x_k$ is recovered by adding up $m_k$ weighted by $p_k$. The final encoding vector e is generated by concatenating this result with the initial encoding vector. Then, using the paragraph-encoded sequence input, a paragraph-level RNN is used to retrieve the final encoding vector for the entire body text.

## 3.3 Data Augmentation

Data augmentation in data analysis refers to techniques for increasing the amount of data available by including extra copies that are only slightly altered from the existing data or by generating fresh synthetic data from the existing data. It serves as a regularizer and lowers overfitting when a deep-learning model is being trained. AHRDE performs better than ERE when there are only a few paragraphs, but as the number of paragraphs increased, the performance gap closed, and ERE was given the highest mark for input that was unusually long (i.e., a news article containing 19-20 paragraphs). As compared to AHRDE, ERE has fewer trainable parameters, which may help to explain this pattern.

After using the IT data augmentation strategy, the newly proposed ERE and AHRDE models consistently demonstrated competitive performance regardless of the paragraph size in the {body text}. When the IT approach was used, prediction performance also improved significantly. RDE and

CDE benefited the most from the IT technique, achieving results that were comparable to the hierarchical model.

### 3.3.1 Independent Training (IT) Method

We use an Independent Training (IT) Method, a data preprocessing method which divides the body text's paragraphs and explains the relationships between each one and the headlines separately. To determine the incongruence score, each paragraph of a news report is compared to the headline. The maximum value is used to determine the final incongruence score.
The IT technique, as shown in Figure 3.2, examines each paragraph's relationship to the news headline to determine the degree of incongruence.

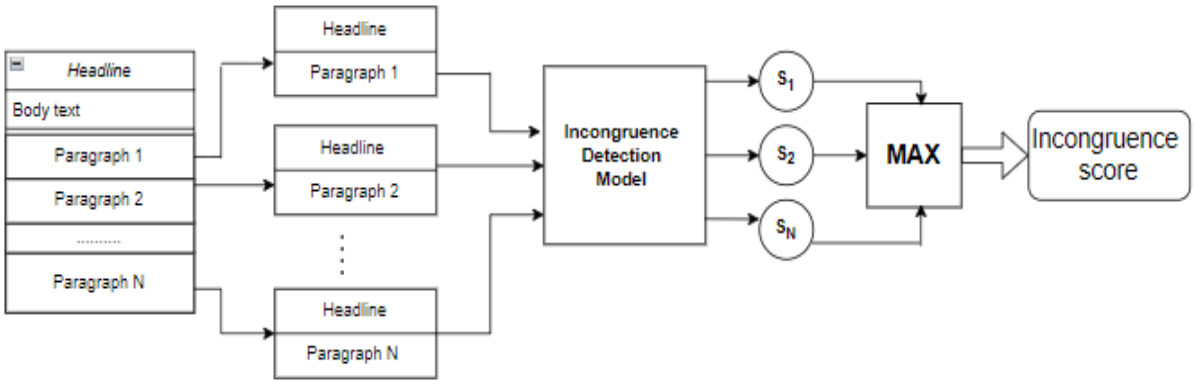



Figure 3.2: Diagram of the independent training method.

To determine the incongruence score, each paragraph of a news report is compared to the headline. The completed incongruence score is computed using the maximum value. The sum of the greatest number of incongruence levels for the pair of title{Headline} and message body{Body Text} yields the final incongruence score as follows:

$$P(lb) = \max(S_{1:p}), \quad \ldots.(3.7)$$

here $S_p$ is the incongruence value derived from the body text's p-th phrase and the title. The method that received the best rating was able to more

accurately recognize news items that contained paragraphs that had no relation to the headline.

The IT approach is used to train models using the paragraph dataset, and the following incongruence scores are obtained:

• **XGB with IT:** The cosine similarities between the {headline} and the {body text} are calculated using the XGBoost (XGB) classifier. These variables' singular values, when decomposed, show how similar certain words are in a title and its matching body content. XGB measures the incongruence score for each paragraph to extract features from the headline and each paragraph of the body text.

• **RDE/CDE with IT:** RDE calculates the similarity of two text inputs and turns the result into a 300-dimensional vector. CDE, on the other hand, takes the most helpful information to determine whether the article's headlines are incongruent. The number of incongruent words is used by CDE to investigate the relationship between the headline and the body text rather than the placement or order of the implanted paragraphs. Both models compare the encoded {headline} to word sequences in each paragraph of {body text}.

• **AHRDE with IT:** These RNN-based techniques outperformed the others in all kinds, which can be related to their capacity to exploit sequential data. For the word-level RNN part, for each title and text, we used two single-layer GRUs units to encrypt the word order. For each phrase RNN of the {title} and {body text} for the paragraph-level RNN component, we employed two individual reversible GRUs units to encode the entire hidden layers sequences. Each paragraph in the body text is encoded using word sequences by the first-level RNN, and the second-level RNN takes the input of the first-level RNN—a sequence of sentences—and models the sequence of paragraphs while maintaining the order.

• **ERE with IT:** When the amount of news content implanted increases, HRE performance decreases. ERE is an RNN-based model, like

AHRDE/RDE, which means it uses sequential data as well. ERE initially calculates the mean of word vectors for each sentence to get the incongruence score for each paragraph. RNN then takes the mean word vectors as input and encodes a sequence of sentences.
When the amount of implanted content is little, the IT method and trend become much more obvious. This is because the model can study each portion of the news item more thoroughly when it is separated into many paragraphs. The model's hierarchical architecture allows it to achieve similar effects over the entire text.

## 3.4 Flowchart of the Model

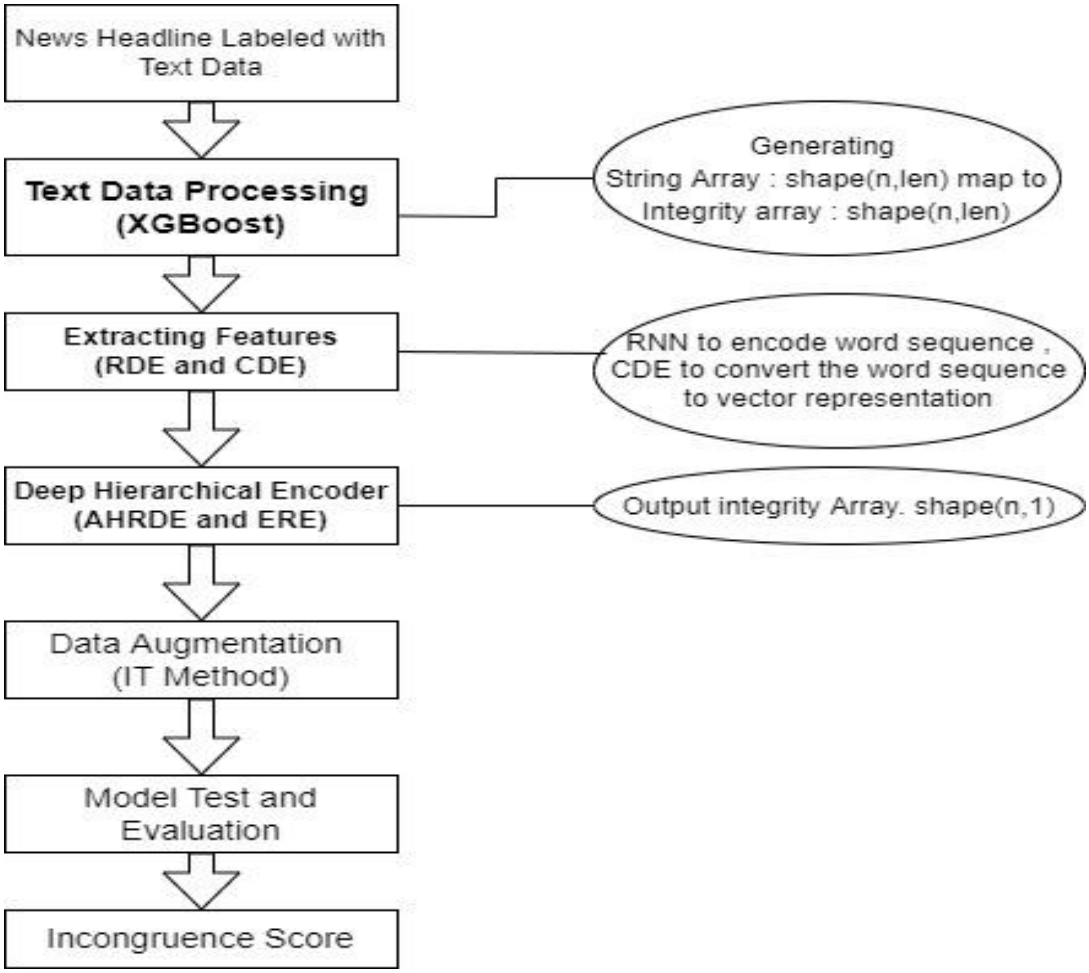



Figure 3.3: Flowchart showing the various modules of the Deep Hierarchical Encoder.

# Chapter 4

# Evaluation

## 4.1 Dataset

- We have downloaded the ‘Fake News Detection’ dataset from kaggle. This is a high-quality dataset with labeled attributes.
- There are no spelling mistakes or colloquial use in news headlines because they are written by experts formally. This minimizes the sparsity of the data while also increasing the likelihood of identifying pre-trained embeddings.
- This news headlines dataset is self-contained, as opposed to tweets that react to other tweets. This would enable us to distinguish the actual sarcastic portions for incongruence detection.
- This fake news detection dataset for headline incongruence issues is compiled from a variety of news websites in an effort to overcome the drawbacks associated with noise in datasets.
- To cover a wide range of news, they collected number of 40,000 items across six different fields (local, global, social, entertainment, games, and science) through leading news websites.

### 4.1.1 About the dataset file:

Each record has four characteristics:

**Headline:** the news article's headline.
**Body:** contains the body of the news article.
**Article_link(URLs):** Here's a link to the original news article. This can be used to acquire further information.
**Label:** If the record is sarcastic/incongruent its value is 1, otherwise it is 0.

Table 4.1:The table below shows examples of assertions as well as side information.

| |
|---|
| **Headline:** Four ways Bob Corker skewered Donald Trump<br><br>**Body:** Image copyright Getty Images On Sunday morning, Donald Trump went off on a Twitter tirade against a ...<br><br>**URLs:** http://www.bbc.com/news/world-us-canada-41419190<br><br>**Label:** 1 |
| **Headline:** Linklater's war veteran comedy speaks to modern America, says star<br><br>**Body:** LONDON (Reuters) - "Last Flag Flying", a comedy-drama about Vietnam war veterans, will resonate with…<br><br>**URLs:** https://www.reuters.com/article/us-filmfestival-london-lastflagflying/linklaters-war-veteran-comedy-...<br><br>**Label:** 1 |
| **Headline:** War on Cash Backfires on India's Economy<br><br>**Body:** By Clint Siegner Indian Prime Minister Narendra Modi launched a surprise attack on cash in late 2016...<br><br>**URLs:** https://www.activistpost.com/2017/09/war-cash-backfires-indias-economy.html<br><br>**Label:** 0 |

## 4.1.2 Dataset Description:

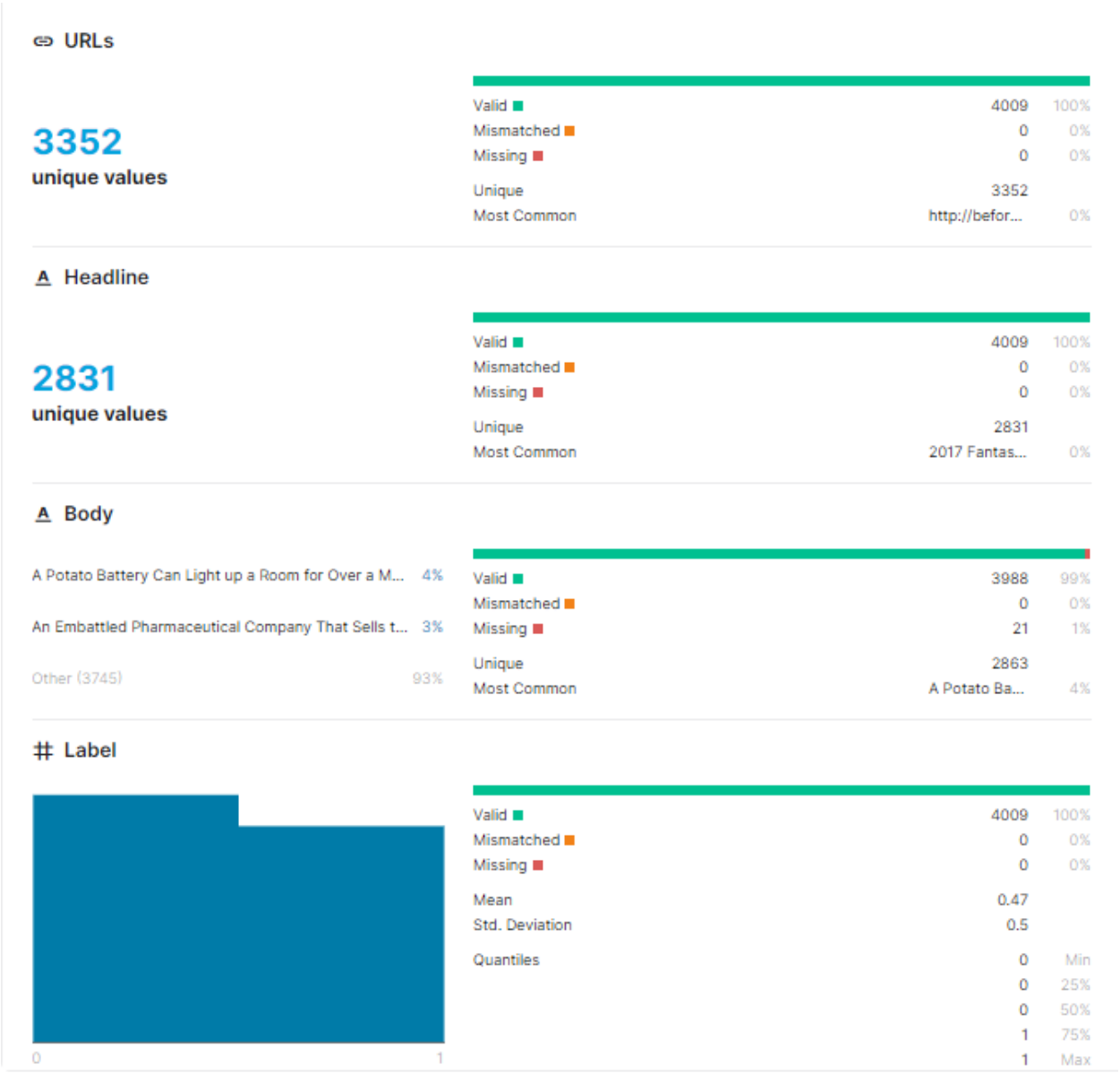



Figure 4.1: Screenshot of the dataset description [32].

The dataset contains 28,619 samples separated into the separate train, validation, and test sets with an 8:1:1 ratio. Each sample includes a statement as well as a variety of related side information, such as URLs, headlines, body parts, and labels. The label indicates whether or not the news is contradictory. Although the remarks in the dataset are strictly political in nature, they come not just from Democrats and Republicans, but

also from non-political persons or groups, as well as from entertainment and sports.

### 4.1.3 Statistics of the Dataset:

The following table summarizes the general statistics of this dataset:

Table 4.2 Dataset statics.

| Statistic/Dataset | Headline | SemEval |
|---|---|---|
| # Records | 28,619 | 3,000 |
| # Sarcastic records | 13,635 | 2,396 |
| # Non-sarcastic records | 14,984 | 604 |
| % of pre-trained word embeddings not available | 23.35 | 35.53 |

We can observe that the percentage of terms not included in word2vec vocabulary is far lower in the Headlines dataset than in the SemEval dataset, despite the fact that the text is much more formal in language. The goal of the SemEval series of worldwide natural language processing (NLP) research training seems to be to advance the state-of-the-art in text summarization and to support the creation of high-quality annotated datasets in a range of progressively more difficult natural language semantics challenges.

## 4.2 Preprocessing

We undertake the following pre-processing steps prior to training models:

- For statements having tokens, stopwords and punctuation are removed. All money characters, as well as percentages and numbers, are converted into a single token.
- Filtering headlines:- Headlines are filtered by removing any non-alphanumeric characters but stop words are preserved since a

text's word order is directly learned by an algorithm.

- Every user comment and tag is substituted for a space, while photographs and flags from posts are skipped.
- The headline and body are each handled as a word sequence. They are combined into one unit vector after being turned into an embedded vector.
- We replace the URLs with the string ‘http‘. And define max features.
- Vectorize & Convert text for input.
- Splitting data to train & test.

## 4.3 Experimental settings

- Python was used for all of the experiments, and Google Colab was used to build and train some of the modules.
- Text is vectorized and converted into Sequences using the tokenizer so that the Network can handle it as input.
- Optimization was done using Adam and activation functions such as sigmoid and softmax.
- We used a batch size of 32 to train each classifier model. We only utilized the first token output from each model produced by each strategy as input to the classifier layer.
- We apply 300 pretrained GloVe embeddings, 100 hidden LSTM states, a learning rate of 0.001, and a batch size of 100. with the loss function being logits.
- With a maximum sentence length of 50, a dropout rate of 0.2, and a number of epochs of 10, we use softmax cross-entropy.
- We utilized the weighted f1 score as a measure for evaluating performance in both the classification and detection tasks during the evaluation.
- To measure the aggregate performance of four different baseline models, we used a weighted fine-grained F1 score, where the weights for the scores of each class are the percentage of their positive cases.
- We apply an IT data augmentation method to all four models and

that gives an increase in performance.

- Whether or not the IT technique was applied, the currently proposed ERE and AHRDE model successfully showed superior results regardless of the number of the paragraph in {actual text}.

Word embeddings were first learned and updated randomly for our model while it was being trained. After all, 50-dimensional embedding matrices were employed. On the development set, we strictly turned on all hyperparameters, and we looked for the optimal outcome based on the accuracy score. Table 4.3 describes our arrangement in more detail.

Table 4.3: The experiment setup

| Hyperparameters | Value |
|---|---|
| Batch size | 32 |
| Word embedding size | 50 |
| Side information size | 32 |
| Character embedding size | 20 |
| Attention size | 32 |
| Test size | 0.33 |
| Random state | 42 |
| Learning rate | 0.001 |
| Max statement length | 30 |
| Dropout | 0.2 |
| Validation_size | 1500 |
| Optimizer | Adam |

### 4.3.1 Metrics for evaluation:

The following are the important categorization metrics:

• ***Accuracy:*** The number of correct predictions divided by the total number

of examples yields accuracy.

$$\text{Accuracy} = (tp+tn)/(tp+fp+fn+tn) \quad \ldots.(4.1)$$

• ***Precision:*** The fraction of accurate positive predictions divided by the total number of correct cases yields the precision:

$$\text{Precision} = tp/(tp + fp) \quad \ldots.(4.2)$$

• ***Recall:*** Recall is a metric used to assess how many accurate predictions the classifier was able to identify:

$$\text{Recall} = tp/(tp + fn) \quad \ldots.(4.3)$$

• ***F1 Score:*** F1 is used to discover the optimal balance of recall and accuracy and is calculated as follows:

$$F1 = 2 * (\text{Precision} * \text{Recall})/(\text{Precision} + \text{Recall}) \quad \ldots.(4.4)$$

Fp, tp, and fn stands for False Positive, True Positive, and False Negative. We examine the intricacies of those notions because they are well-known in natural language processing.

**Implementation tools:** All of the source code is written in Python 2.73 and uses the deep learning library Tensorflow..

### 4.3.2 Case Study:

We tested our pre-trained models on more recent news stories to determine how effective our dataset and proposed models are at spotting incongruent headlines in the real world. The generalizability of our technique in the actual world may be demonstrated by testing our model with the dataset. We began by manually analyzing a random sampling of news items to see whether there were any headlines that were inconsistent. We were, however, unable to locate a significant number of newspapers with incongruent headlines for analysis. This might be because there aren't many headlines that contradict each other in the reality. Therefore, we chose to manually examine the top N news stories

using the incongruence scores generated by the given model rather than scanning through a random selection set and marking them for assessment.

A case study of fake news is given in table 4.4 below. The first fake news report says that Donald Trump received support from the American Amish Brotherhood[1]. But it's a fake CNN page on the website. On the internet, misleading news headlines are simple to find and rarely correspond to the news stories themselves. The second false claim is that Hillary Clinton had less than a 24-hour window to withdraw from the election, which is untrue [2].

Table 4.4 Case Study of the dataset.

| Headline | Body | URL | Label |
|---|---|---|---|
| The Amish Brotherhood has endorsed Donald Trump for president | The Amish, who are direct descendants of the protestant reformation sect known as the Anabaptists, have typically stayed out of politics in the past. As a general rule, they don't vote, serve in the military, or engage in any other displays of patriotism. This year, however, the AAB has said that it is imperative that they get involved in the democratic process. | http://cnn.com.de/news/amish-commit-vote-donald-trump-now-lockpresidency/ | 1 |
| Wikileaks Gives Hillary An Ultimatum: QUIT, Or We Dump Something Life-Destroying | On Sunday, Wikileaks gave Hillary Clinton less than a 24-hour window to drop out of the race or they will dump something that will destroy her completely. Recently, Julian Assange confirmed that WikiLeaks was not working with the Russian government, but in their pursuit of justice, they are obligated to release anything that they can to bring light to a corrupt system and who could possibly be more corrupt than Crooked Hillary? | http://thelastlineofdefense.org/wikileaks-gives-hillary-an-ultimatum-quitor-we-dump-something-life-destroyi | 1 |

We employ a semi-supervised approach in an effort to fully exploit the

larger unlabeled data set in order to build a strong classifiers effective of dealing with a variety of cases. A number of tests are performed to compare baseline procedures to the newly proposed models.

- We started by manually inspecting random samples of news stories to check whether they had any headlines that were incongruent.
- We train using 80% of the data, validate with 10% of the data, and test with 10% of the data. Each experiment is carried out a minimum of ten times.

## 4.4 Results

The effectiveness of LSTM and RNN neural networks for detecting inconsistencies between the news headline and the body content is shown in Table 4.5. Single attention outperforms the hybrid CNN even if it is far from ideal. In fact, in the development and training sets, our deep encoder outperformed the hybrid CNN by 4% and 0.25%. Our approach has several advantages over composite CNN, including a network layout that is much easier than composite CNN networks. Our model has a lot fewer parameters to train because the hybrid CNN uses two different CNNs and an LSTM instead of any deep model as an intermediary layer.

Table 4.5: Deep hierarchical encoder accuracy

| | Accuracy | Precision | Recall | F-Score |
|---|---|---|---|---|
| | | | | |
| Without IT-Method | 0.84 | 0.82 | 0.81 | 0.82 |
| With IT-Method | 0.97 | 0.96 | 0.97 | 0.96 |

The three deep learning models perform better than feature-based machine learning models, as our first finding shows. In terms of accuracy, the

recently proposed AHRDE outperformed all other deep learning models (0.97). Second, when the IT method was used, prediction performance improved significantly. When the IT data augmentation method was used, the newly proposed models (AHRDE and ERE) consistently outperformed baseline approaches (e.g., XGB, CDE).

Three observations are made here. First, the AHRDE model with IT augmentation regularly outperforms the AHRDE model without the IT technique in terms of precision. This study backs up the IT method's higher performance in a variety of tests. Second, for the top 25 articles, the AHRDE model with IT had a precision of 1.0. Despite being trained on a different dataset, the model was still able to distinguish accurately between real-world situations where the headline conveyed one story and the associated body content told another. Third, the AHRDE model with IT had a precision of 0.82 when we examined the top 250 articles, which is high enough to be utilized for detection in legitimate news sites. The outcomes show that the headline incongruence issue might be resolved in the actual world using our strategy and this dataset.

### 4.4.1 Representation And Classification of the result

#### 4.3.1.1 Representing Articles

We investigated the relationship between the headline and the entire article when representing an article as a sequence of words. We used the conditional encoder to test encoding a headline conditioned on the representation of the associated article, $A \rightarrow H$, as well as encoding an article conditioned on the representation of the corresponding headline, $H \rightarrow A$. Furthermore, we changed the conditional encoder so that the first RNN processes the article bidirectionally into two separate conditional states, encoding the headline with two separate RNNs, $A^{\leftrightarrow}\rightarrow H$. The idea behind this relationship representation is that the beginning and end of an article are more important than the content in between for detecting the relationship.

We also studied the relationship between each sentence in an article and the headline, representing the article as a set of sentences, due to the frequent asymmetry of an article being a large text document and the associated headline being a short text. The relationship between the headline and the entire article is represented by the average of the relationship representations of all sentence-headline pairs. All sentences were padded to equal length, allowing them to be processed as a batch of sentences, significantly reducing the time required to train the model.

#### 4.4.1.2 Employing Attention

First, we looked at the RNN attention model, which was applied to a conditional encoder that represented articles as word sequences, and extracted an attention-weighted representation of an article for each word in a headline. The attentive description of an article, which corresponded to the last word in a headline, was combined with the relationship representation retrieved using the conditional encoder and supplied into the classifier as input. The idea was to include an attention mechanism in the model that learns to focus on specific words or phrases in an article that are important for detecting the relationship based on the content in the headline. We used the attention model for each sentence-headline pair later in the experiment. In this setup, the attentive representation of the sentence corresponding to the last word in a headline was concatenated with the conditional encoder-extracted sentence-headline relationship representation.

The set of attentive sentence representations and sentence-headline relationship representations that resulted was averaged to produce a representation of the entire article and its relationship to the associated headline. Another experiment focused on an attention mechanism that learned to find relevant phrases in an article based on the headline's substance. Using an additional RNN encoder, we derived a representation of the corresponding headline in this scenario. The attention model generated attention weights for the set of sentence representations of an article using the final state of this RNN, i.e. the set of final states of the first

RNN in the conditional encoder.

Finally, we ran an experiment that combined the two attention techniques outlined above. We used one attention model to generate attention weights for sentences and another attention model to generate attention-weights for words in sentences based on the content of a headline. The findings of our experiment are depicted in figure 4.2 as the combined attention of the deep hierarchical encoder is represented as a confusion matrix.

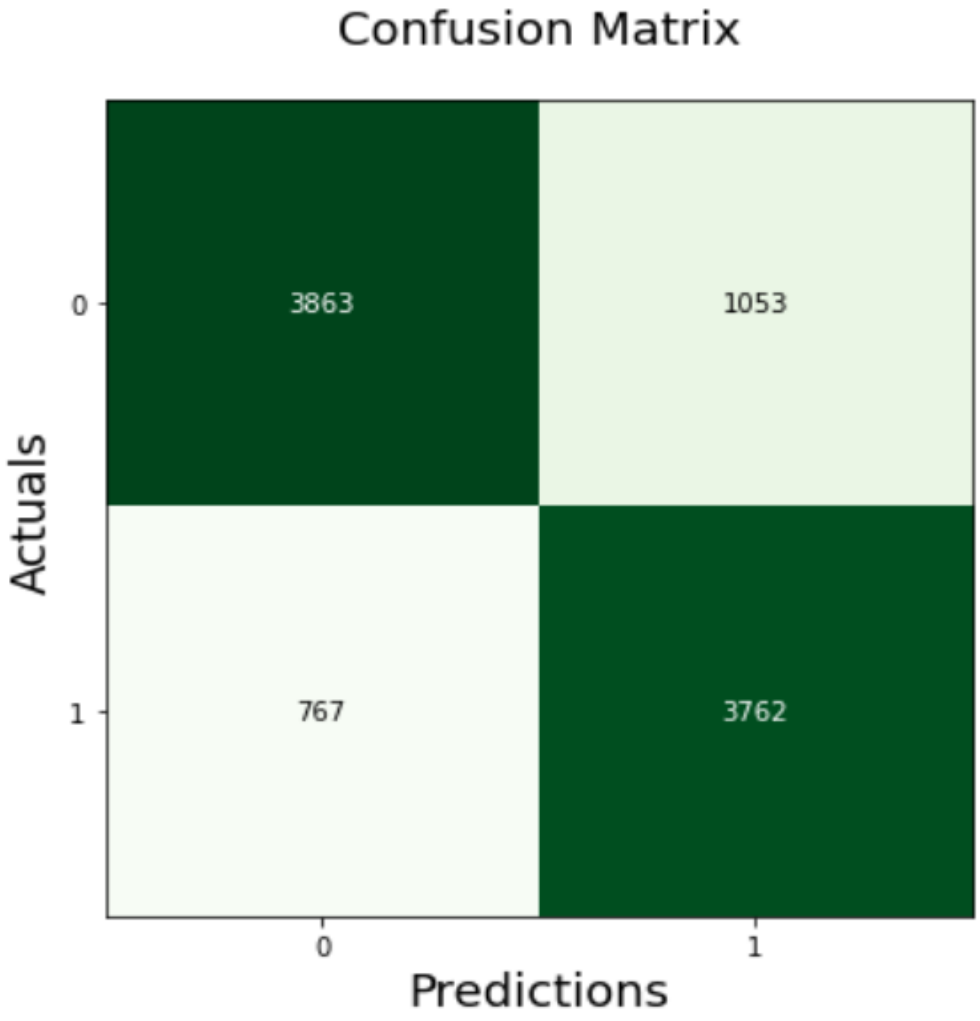



Figure 4.2: Confusion matrix of our best model (AHRDE and ERE)

### 4.4.2 Learning With Limited Supervision

A recurrent neural network often requires a large number of training examples to adequately learn the underlying relationship in the data and function well in cases it has never seen before. During our research into various models, we discovered that they dramatically overfit the training data. We studied commonly used strategies for reducing generalization errors when training RNNs, such as adopting dropout and regularization of parameters, to address the issue of overfitting the training data. During training, we also looked into a data augmentation method based on resampling samples and substituting key phrases. Another experiment looked into the effects of transfer learning by pre-training an RNN encoder for the purpose of language modeling on another dataset without supervision.

Since this design performed well in earlier experiments, we employed GRU architecture in the RNNs and the ReLu activation function in the MLP's hidden layer. Because a model has seen all headlines in the official hold-out set during training and because its performance on the hold-out set is much better than on the test set, we will now place a greater emphasis on a model's performance on the test set in order to reduce generalization error.

#### 4.4.2.1 Dropout Words

During training, we undertook a series of tests to see if dropout might be used as a regularization strategy by eliminating words from headlines or articles at random. The goal was to train a model that can generalize effectively on terms found in headlines or articles and learn to identify which words in a given context are important for classification. The dropout approach we looked into is as follows: keep each word in a headline or article with a probability of p = 0.9.

#### 4.4.2.2 Dropout Sentences

We conducted a series of studies to investigate dropout as a regularization strategy by randomly removing sentences from an article during training, similar to the notion behind applying dropout to words. We looked into the following dropout strategy: Keep the sentence with probability $p \in \{0.5, 0.6, 0.7\}$. for each sentence in the article.

#### 4.4.2.3 Synonym Replacement

We studied a data augmentation strategy based on replacing specific words during training in a series of trials. The purpose was to artificially increase the number of training examples in order to improve generalization on the terms in the training set. We tagged each word in articles and headlines with its part-of-speech tag, one from the set of nouns, adjectives, verbs, and adverbs, during the pre-processing step. Using the WordNet database, we also extracted synonyms for each term at this stage. Using a multinomial distribution, the technique we tested for replacing words during training randomly resampled each word into itself or one of its counterparts. For a set of synonyms, we investigated three multinomial distributions: (i) normalizing their document frequencies (DF), (ii) normalizing their inverse document frequencies (IDF), and (iii) normalizing their document frequencies multiplied by the cosine similarity between the original word and each synonym (DF × COS).

#### 4.4.2.4 Transfer Learning

Transfer learning seeks to transfer the information of a model that has been trained for a specific task to better learn in a new task. We examined transfer learning in a series of experiments by pre-training the headline RNN in the conditional encoder for a language modeling challenge. We were especially interested in the unsupervised task of predicting the next word for each word in a headline. The RNN must capture a meaningful semantic representation of the headline for this assignment. The goal was to improve the conditional encoder's parameter initialization before

fine-tuning them on the real supervised job of recognizing the relationship between a headline and the related article.

The parameters generated from pre-training an RNN on the combined dataset for a single epoch were used as the starting point for the headline RNN in the conditional encoder. Figure 4.3 shows the outcomes of the experiments after the data augmentation method as a confusion matrix.

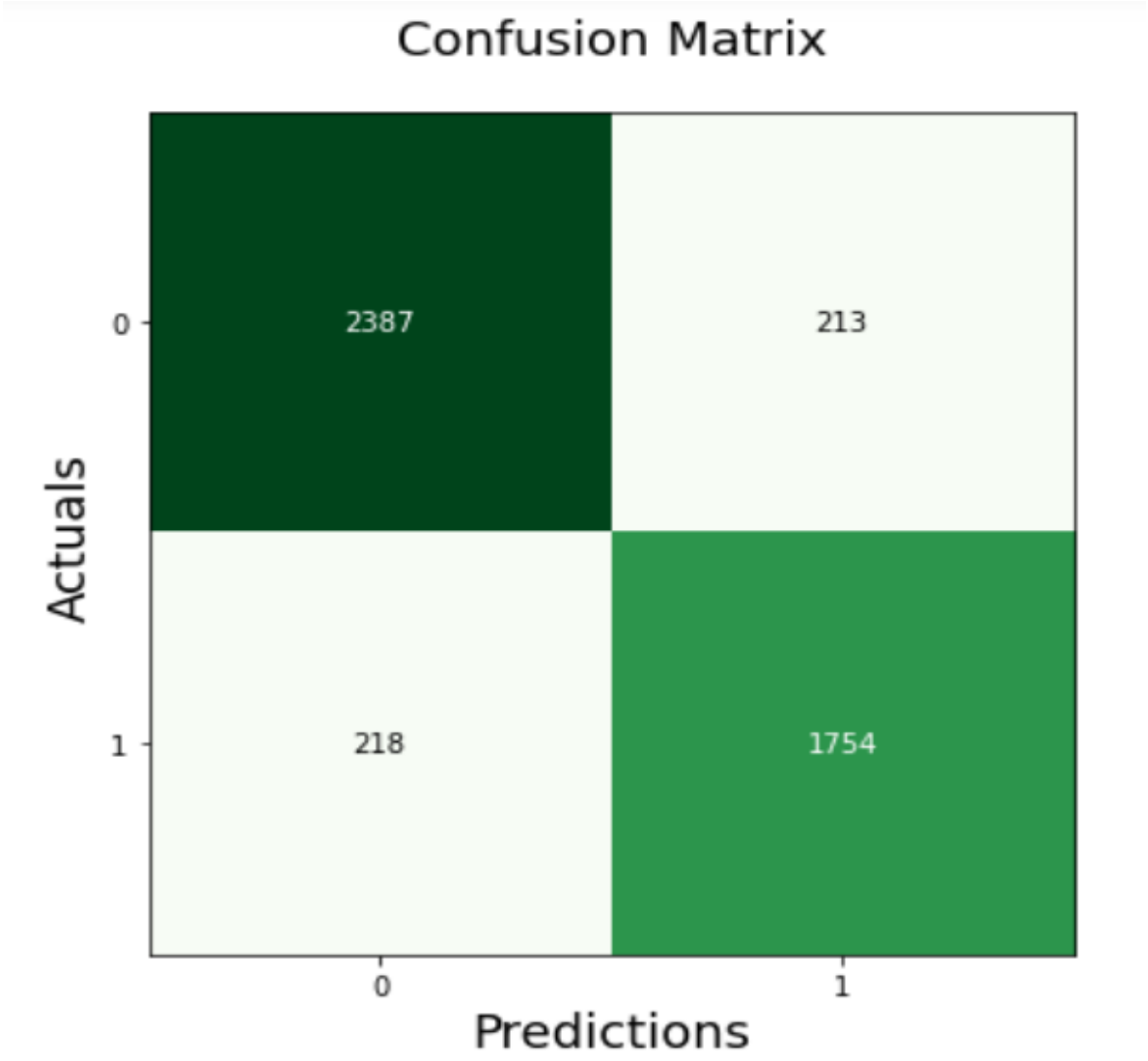
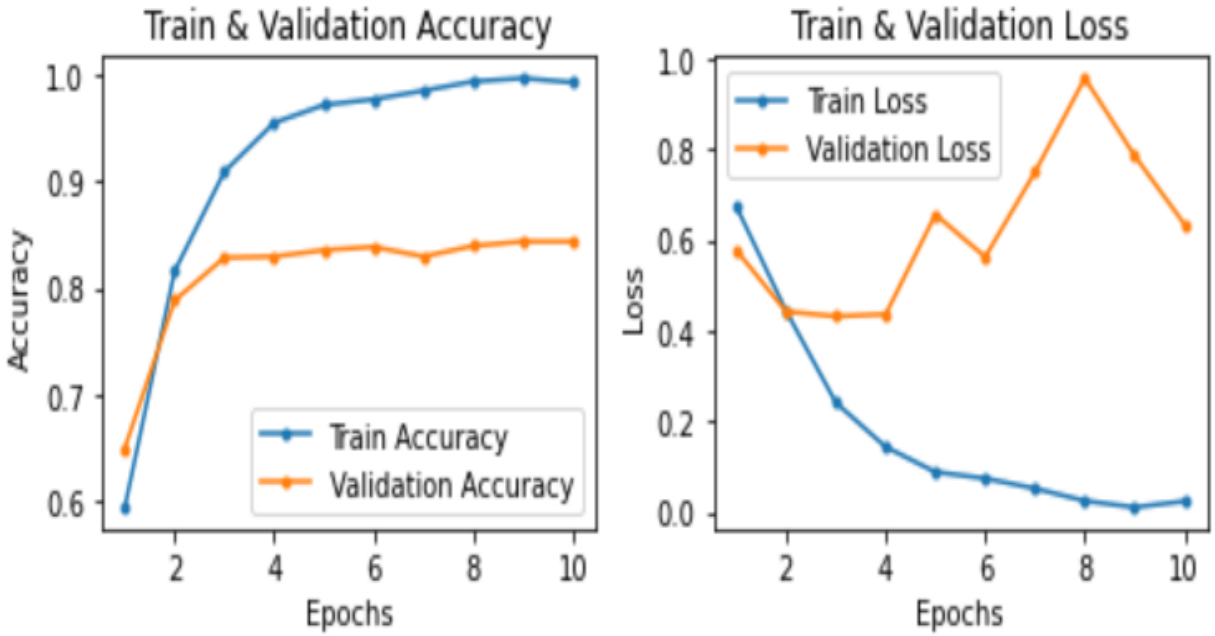



Figure 4.3: Confusion matrix of our best model (AHRDE and ERE) after applying the data augmentation methods.

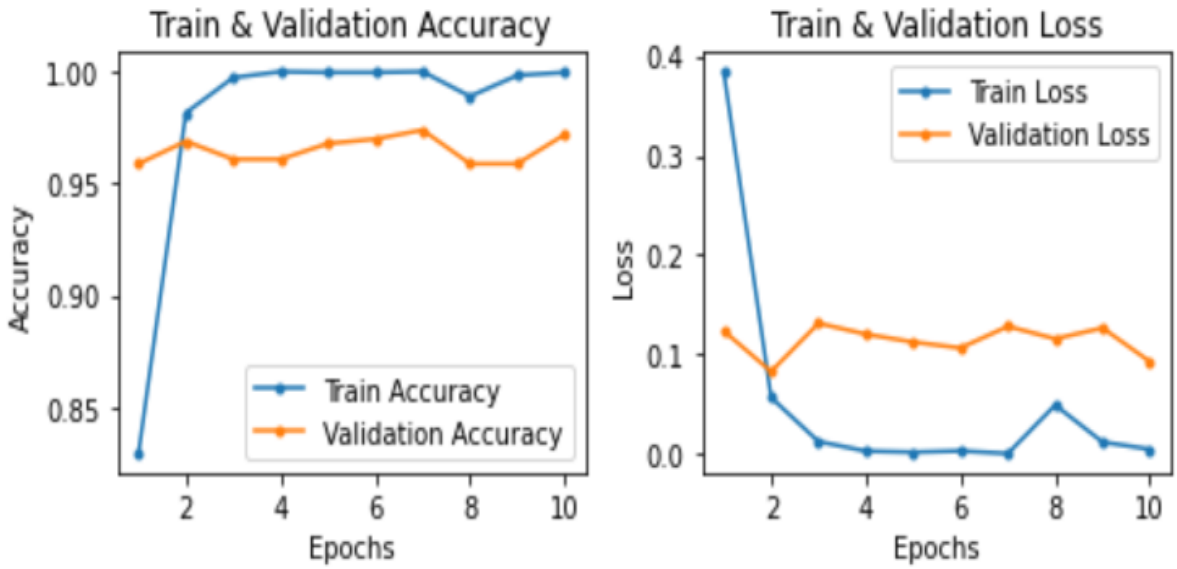
### 4.4.3 Plotting the training procedure of an RNN based model

As a result, we found that on the standby sets, the RNN-based models we trained beat the test set. More crucially, because the effectiveness of a model on the standby set does not necessarily represent the performance on the test set, we were unable to utilize the performance of a model on the hold-out set as a foundation for selecting which model to be examined on the test set. Figure 4.4 illustrates this.




(a) Model Accuracy and Loss without IT

We have carried out trials for each type utilizing both with-IT and without-IT approaches. Figures 4.4(a) and 4.4(b) show that the IT technique outperforms the method without IT in terms of performance. When the amount of implanted content is only a small percentage, an IT method and such a tendency are very noticeable. This is so that the model can more thoroughly assess each paragraph when news content is broken up into multiple paragraphs.




(b) Model Accuracy and Loss without IT

Figure 4.4: For each epoch, plotting the training procedure of an RNN-based model for 10 epochs, where (a) and (b) represent the relative accuracy and loss on the test set, non-overlapping hold-out, and training sets.

## 4.5 Performance Comparison

We compared feature-based approaches and traditional deep learning models to our hierarchical deep learning approaches (AHRDE, ERE). Due to the fact that an ensemble of XGB and CDE was successful in the FNC-1 competition (Talos 2017), we offered ensemble models that incorporate XGB's predictions with each deep learning model.

Performance data for each strategy is shown in Table 4.6. Performances of the entire dataset are also compared with results obtained using the paragraph dataset's data augmentation method. We provide the accuracy value, a balanced metric in relation to label distribution. Our hybrid RNN has a lot fewer parameters to train because it combines two different RNNs, an LSTM, and a deep hierarchical model as intermediate layers. Basically, our dual attention approach performs better than the other baseline model.

Table 4.6: Comparison of model performance (top-2 scores marked as bold).

| Model | Accuracy Without DA | Accuracy With DA |
|---|---|---|
| SVM | 0.660 | 0.770 |
| XGB | 0.718 | 0.773 |
| CDE | 0.793 | **0.907** |
| RDE | 0.741 | 0.823 |
| AHRDE And ERE | 0.843 | **0.975** |

With an accuracy of 0.907, CDE outscored the other single models. The AHRDE model was the most accurate among deep learning models, with an accuracy of 0.975. Insufficient variability of training instances in the FNC-1 dataset may be the source of XGB's superior performance over deep techniques. Our findings imply that, in comparison to traditional approaches, the proposed hierarchical neural networks are more effective at learning textual relationships between two texts i.e headline and the body of the news article. We think that the limitation of the FNC-1 dataset is connected to the maximum accuracy of CDE among single models and that the ensemble technique might not be necessary if the dataset is big enough to train neural networks. We discovered that the AHRDE model alone outperformed any combination of other approaches for the ensemble in additional experiments on the dataset given in this study.

## 4.6 Discussion

Until now, we've thought of incongruence scores as a set value for each news story. We used a dataset composed primarily of news scraped from news websites. Domestic, global, social, entertainment, sports, and technology are among the six domains covered by the data. We detect stories with incongruent headlines and congruent ones after training the

two classifiers. The statistics are then analyzed, and some interesting and valuable results are obtained.

Domestic and international news, which typically cover political and military events, are both highly accurate. The lack of seriousness in entertainment journalism may be the source of the most erroneous headlines. The headlines of society news stories that reveal strange stories about social life are often confusing.

We compared feature-based techniques and traditional deep learning models to our hierarchical deep learning approaches (AHRDE, ERE). With an accuracy of 0.975, the AHRDE model was the most accurate among deep learning models. The insufficient variability of training instances in the FNC-1 dataset may explain why XGB outperforms deep techniques. We think that the limitation of the FNC-1 dataset is connected to the XGB's maximum accuracy among single models and that the ensemble technique may not be necessary if the dataset is big enough to train neural networks. We observed that the AHRDE model alone outperformed any combination of other techniques for the ensemble in additional experiments on the dataset reported in this research.

# Chapter 5

# Conclusion and Future Work

## Conclusion

We investigate the problem of incongruent headline detection in this dissertation. To do so, we employed a fake news dataset designed to recognize news stories whose body text contradicts the headline. We also present two neural networks that use a hierarchical recurrent design to efficiently learn the textual link between the headline and body text. The experiments show that models trained on our available corpus perform well on both synthetic and real-world datasets. Additionally, we provide Independent Training, a data augmentation technique that enhances the performance of the models by generating headline-paragraph pairings by dividing the body text into separate paragraphs.

Using the final classifiers, we identify congruent and inconsistent headlines across the entire dataset. The findings of the data study point to the necessity for more severe journalistic rules, particularly in the areas of entertainment and society reporting. Our dataset and method can aid in the detection of news articles with deceptive headlines, which can attract readers' attention and contribute to the growing issue of disinformation in our society.

## Future Work

As a logical continuation of this research, prediction models for identifying news stories with incongruent titles using syntactic characteristics is being developed and improved. Using NLP pipelines like part-of-speech tagging or named entity identification can be a straightforward way to simplify the complexity of raw text input. To more precisely discern the structural relationship between the title and main content, tree-shaped deep neural networks may be developed.

In the future, this study might be expanded to assess the uniformity of titles and content across various online content categories. The rapid identification of such contradictory titles for diverse commodities can make people happier, much like the incongruent news problem. Future studies might integrate different datasets and enhance the AI algorithms that evaluate title and content consistency to achieve this.

# References


[1] Chen, Yimin, Niall J. Conroy, and Victoria L. Rubin. "Misleading online content: recognizing clickbait as" false news"." *Proceedings of the 2015 ACM on workshop on multimodal deception detection*. 2015.

[2] Chesney, Sophie, et al. "Incongruent headlines: Yet another way to mislead your readers." *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*. 2017.

[3] Ecker, Ullrich KH, et al. "The effects of subtle misinformation in news headlines." *Journal of experimental psychology: applied* 20.4 (2014): 323.

[4] Gabielkov, Maksym, et al. "Social clicks: What and who gets read on Twitter?." *Proceedings of the 2016 ACM SIGMETRICS international conference on measurement and modeling of computer science*. 2016.

[5] Normala, Che Eembi Jamil, et al. "Fakeheader: A tool to detect deceptive online news based on misleading news headlines and contents." *Turkish Journal of Computer and Mathematics Education* 12.3 (2021): 2217-2223.

[6] Park, Kunwoo, et al. "BaitWatcher: A lightweight web interface for the detection of incongruent news headlines." *Disinformation, Misinformation, and Fake News in Social Media*. Springer, Cham, 2020. 229-252.

[7] Tacchini, Eugenio, et al. "Some like it hoax: Automated fake news detection in social networks." arXiv preprint arXiv:1704.07506 (2017).

[8] Yoon, S., Park, K., Lee, M., Kim, T., Cha, M., & Jung, K. (2021). Learning to Detect Incongruence in News Headline and Body Text via a Graph Neural Network. *IEEE Access*, *9*, 36195-36206.

[9] Monti, Federico, et al. "Fake news detection on social media using geometric deep learning." arXiv preprint arXiv:1902.06673 (2019).

[10] Zellers, Rowan, et al. "Defending against neural fake news." arXiv preprint arXiv:1905.12616 (2019).

[11] Lu, Yi-Ju, and Cheng-Te Li. "GCAN: Graph-aware co-attention networks for explainable fake news detection on social media." arXiv preprint arXiv:2004.11648 (2020).

[12] Saikh, Tanik, et al. "A deep learning approach for automatic detection of fake news." arXiv preprint arXiv:2005.04938 (2020).

[13] Yuan, Chunyuan, et al. "Early Detection of Fake News by Utilizing the Credibility of News, Publishers, and Users Based on Weakly Supervised Learning." arXiv preprint arXiv:2012.04233 (2020).

[14] Kamal, Ojasv, Adarsh Kumar, and Tejas Vaidhya. "Hostility Detection in Hindi leveraging Pre-Trained Language Models." arXiv preprint arXiv:2101.05494 (2021).

[15] Bhatt, Gaurav, et al. "On the benefit of combining neural, statistical and external features for fake news identification." arXiv preprint arXiv:1712.03935 (2017).

[16] Yang, Yang, et al. "TI-CNN: Convolutional neural networks for fake news detection." arXiv preprint arXiv:1806.00749 (2018).

[17] Nguyen, Duc Minh, et al. "Fake news detection using deep markov random fields." Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). (2019).

[18] Saikh, Tanik, et al. "A Deep Learning Approach for Automatic Detection of Fake News." arXiv preprint arXiv:2005.04938 (2020).

[19] Umer, Muhammad, et al. "Fake news stance detection using deep learning architecture (cnn-lstm)." IEEE Access 8 (2020): 156695-156706.

[20] Cui, Limeng, et al. "defend: A system for explainable fake news detection." Proceedings of the 28th ACM International Conference on Information and Knowledge Management (2019).

[21] Girgis, Sherry, Eslam Amer, and Mahmoud Gadallah. "Deep learning algorithms for detecting fake news in online text." 2018 13th International Conference on Computer Engineering and Systems (ICCES). IEEE (2018).

[22] Thota, Aswini, et al. "Fake news detection: a deep learning approach." SMU Data Science Review 1.3 (2018):

[23] Sahoo, Somya Ranjan, and B. B. Gupta. "Multiple features based approach for automatic fake news detection on social networks using deep learning." Applied Soft Computing 100 (2021): 106983.

[24] Girgis, Sherry, Eslam Amer, and Mahmoud Gadallah. "Deep learning algorithms for detecting fake news in online text." 2018 13th International Conference on Computer Engineering and Systems (ICCES). IEEE (2018).

[25] Zhang, Jiawei, et al. "Fake news detection with deep diffusive network model." arXiv preprint arXiv:1805.08751 (2018).

[26] Wei, Wei, and Xiaojun Wan. "Learning to identify ambiguous and misleading news headlines." *arXiv preprint arXiv:1705.06031* (2017).

[27] Mishra, Rahul, et al. "MuSeM: Detecting incongruent news headlines using mutual attentive semantic matching." *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE (2020).

[28] Chen, Y.; Conroy, N. J.; and Rubin, V. L. 2015. Misleading online content: Recognizing clickbait as false news. In Proceedings of the ACM Workshop on Multimodal Deception Detection (2015)

[29] Chesney, S.; Liakata, M.; Poesio, M.; and Purver, M. 2017. Incongruent Headlines: Yet Another Way to Mislead Your Readers. In Proceedings of the EMNLP Workshop: Natural Language Processing meets Journalism, 56–61 (2017)

[30] Chopra, S.; Jain, S.; and Sholar, J. M. 2017. Towards Automatic Identification of Fake News: Headline-Article Stance Detection with LSTM Attention Models (2017).

[31] Ecker, U. K.; Lewandowsky, S.; Chang, E. P.; and Pillai, R. 2014. The effects of subtle misinformation in news headlines. Journal of experimental psychology: applied 20(4):323 (2014).

[32] "https://rishabhmisra.github.io/publications/"

[33] https://en.wikipedia.org/wiki/Biological_neural_network

[34] Bourgonje, Peter, Julian Moreno Schneider, and Georg Rehm. "From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles." *Proceedings of the 2017 EMNLP workshop: natural language processing meets journalism*. 2017.

[35] https://edition.cnn.com/2018/10/20/politics/

[36] https://www.motherjones.com/environment/2015/11/