Implementation of Machine Learning Approach to Detect Clickbaits in Online News

Prajwol Lamichhane
Student, Microdegree in Artificial Intelligence
Fuse Machines Inc.
Kathmandu, Nepal
prajwollamichhane11@gmail.com

Kamal Shrestha
Student, Microdegree in Artificial Intelligence
Fuse Machines Inc.
Kathmandu, Nepal
kamal.shrestha@student.ku.edu.np

***Abstract*—Citizen Journalism has boomed the opportunities for content marketing and public opinion. It is a golden rule of journalism, taught to any news reporter at the beginning of their career - your introduction should grab the reader straight away. Almost all of the online news media outlets completely depend upon the revenues generated from the clicks made by their readers,and due to the presence of numerous such outlets, they need to compete with each other for reader attention. So, In order to attract the readers to click on an article and subsequently visit the media site, the outlets often come up with catchy headlines accompanying the article links, which lure the readers to click on the link. Such headlines are known as Clickbaits. While these baits may trick the readers into clicking, in the long run, clickbaits usually don't live up to the expectation of the readers, and leave them disappointed.**

In this work, we attempt to detect clickbaits based on features extracted from the headline of the news and then classify it accordingly. We were able to run a number of extensive cross checking to different news headlines from multiple platforms and were able to obtain an accuracy of 79% in detecting clickbait headlines.

***Index Terms*—clickbait, natural language processing, machine learning, logistic regression**

I. Introduction

One of the most important considerations for online journalists is attracting a reader's attention through headlines alone. Without interest in a headline, there will be no interest in the story. By enticing a reader to click on a catchy headline, there is a high click rate for the website, which leads to advantages for the website owners.

Essentially, in the online world, every media outlet has to compete with many such outlets for reader attention and make their money from the clicks made by the readers. Therefore, to attract the readers to visit the media site and click on an article, they employ various techniques, such as coming up with catchy headlines accompanying the article links, which lure the readers to click on the links. Such headlines are known as Clickbaits.According to the Oxford English Dictionary, clickbait is defined1 as "(On the Internet) content whose main purpose is to attract attention and encourage visitors to click on a link to a particular web page." Examples of such clickbaits include *"Mycha started drinking two glasses of bitter-guard juice everyday for seven days and the results are amazing, A school girl gave her lunch to a homeless man. What he did next will leave you in tears!, 21 stars who ruined their face due to plastic surgery. Talk about regrets!, Only the people with an IQ above 160 can solve these questions. Are you one of them? Click to find out. . . , Was He Raised by a Family of Wolves, or Something Crazy?! , Whoa! I Can't Imagine What It Might Be!"*.

The widespread adoption of this style of journalism is already under criticism, that is, the seemingly sensationalised tabloid-style approach, and online clickbait is perceived to be even more dangerous [1]. Clickbait headlines condense certain aspects of a story, maximising manipulation of fact and using word-of-mouth as gospel. Although this approach clearly entices readers, it perpetuates gossip and mistruth, and journalistic integrity becomes ever weakened. The internet is a dangerous medium for news stories are presented as an ensemble in "streams" so that reputable and non-reputable sources are mixed together and headlines are taken out of context [2].

However, getting more clicks and thereby more visits on websites ultimately amounts to one thing: these websites can raise their advertising rates and make profit. Publishers often incentivise their writers to get clicks on stories, which can lead to the simplification of articles and headlines for monetary gain [3]. Slant, an online magazine, pays writers $5 for every 500 clicks on a story on top of their monthly salary (according to the Columbia Journalism Review) and this is a pattern followed by more and more websites. This analysis will uncover the function of 'clickbait' in practice and public response to such devices, by adopting a qualitative research method to record participants' online behaviour. This study is based on a number of previous research or theories, and its key consideration is the impact that factors such as length average word lengths, stop word's ratio, presence of cardinality, presence of verbs, nouns, auxiliaries, coordinating conjunctions and also the polarity and subjectivity of the clickbaits.

II. State of Art

The origin of clickbaits can be traced back to the advent of tabloid journalism, which started focusing on 'soft news' compared to 'hard news', and sensationalization rather than

reporting in depth and truthful account of the events. Even with all these hue and cry around the ill effects of clickbaits, there has been little attempt to devise a systematic approach for a comprehensive solution. In 2014, Facebook declared that they are going to remove clickbait stories from users' news feeds, depending on the click-to-share ratio and the amount of time spent on these stories. Yet, Facebook users still complain that they continue to receive clickbaits and there is a renewed effort to clamp down on clickbaits. There has been recent works to understand the psy- chological appeal of clickbaits. Blom et. al. [4] examined how clickbaits employ two forms of forward referencing – discourse deixis and cataphora – to lure the readers to click on the article links whereas Chen et. al. [1] argued for labeling clickbaits as misleading content or false news.

In a recent work, Potthast et al. [5] attempted to detect clickbaity tweets in Twitter. The problem with such standalone approaches is that clickbaits are prevalent not only on particular social media sites, but also on many other reputed websites across the web. For example, the 'Promoted Stories' section at the end of the articles in the websites of 'The Guardian', or 'Washington Post' contain many clickbaits. Therefore, we need to have a comprehensive solution which can work across the web.

Rowe [6]examined how the common tabloid properties like sim- plification and spectacularization of news, are making its way into the more conventional newspapers and how it is changing the course of professional journalism.

There also have been some ad-hoc approaches like 'Downworthy' [7] which detects clickbait headlines using a fixed set of common clickbait phrases and then converts the headlines into something more garbage-ish, or 'Clickbait Remover for Facebook' [8] which prevents the links to a fixed set of domains from appearing in the users' news feeds. The problem with having a fixed rule set is they are not scalable and may need constant tuning with the emergence of new clickbait phrases. Similarly, preventing links to a fixed set of domains will also block article links which are not clickbaits.

[9] claims "Click-bait is rarely newsworthy, but it does attract eyeballs. The assumption seems to be that audiences might stay for the "serious" content after gorging on the fluff.One of the best qualities in the journalistic culture is skepticism. But when it comes to digital, skepticism has been replaced with unquestioning enthusiasm."

[4] maps the use of forward-referring headlines in online news journalism by conducting an analysis of 100,000 headlines from 10 different Danish news websites. The results show that commercialization and tabloidization seem to lead to a recurrent use of forward-reference in Danish online news headlines.

III. Datasets

The dataset contains two files each consisting the headlines of 16,000 articles. Two files, one for titles marked as click bait and another as non-click bait.

For Non-click bait, the dataset contains headlines extracted from the corpus of 18,513 Wikinews articles collected by NewsReader [10].In Wikinews, articles are produced by a community of contributors and each news article needs to be verified by the community before publication. There are fixed style guides which specify the way some events need to reported and presented to the readers. For example, to write the headline of a story, there are a set of guidelines 4 the author needs to follow. Due to these rigorous checks employed by Wikinews, the headlines of these articles can be considered as gold standard for non-clickbaits.

For Click bait, the dataset contains titles manually extracted from the following domains: 'BuzzFeed', 'Upworthy', 'ViralNova', 'Thatscoop', 'Scoopwhoop' and 'ViralStories'. A total of 8,069 articles were extracted during the month of September, 2015. Both the datasets were extracted, labelled and pre-processed for further experimentations.

IV. Proposed Approach

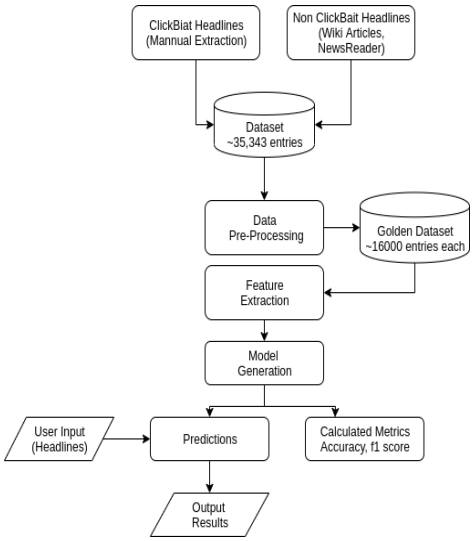

Fig. 1. System Flow Diagram

A. Data Pre-Processing

An extensive amount of both clickbait and non clickbait data sets were collected.The dataset at the begining was counted as 35,343 including both the clickbaits and non clickbaits entries. Now, Data Pre Processing Steps include:

1) The examples without presence of output label were removed.
2) The nan rows and the rows with nan values were removed.

Then, we randomly selected 16,000 of each clickbaits and non clickbaits entries totaling into 32,000 to prepare the golden dataset for further feature extractions.

B. Data Visualization

Following plots were used to visualize the distribution and correlation of dataset.

1) Bar and Pie Chart Showing number of Clickbait and Non Clickbait Entries
2) POS Tags Visualization
3) Correlation of Extracted Features
4) TSNE Plot and PCA Plot
5) Confusion Matrix Plot

C. Features Extraction and Selection

Following features were extracted from textual form of labelled data entries.

1) Normal Analysis on the Clickbaits
   a) **Length:** The length was calculated by summing up all the characters from the clickbait column.
   b) **Words Count:** The number of words present on the clickbait column were calculated.
   c) **Average Word Length:** It was calculated by dividing the length obtained by the number of words.
   d) **Stop Words to Content Words Ratio:** The number of stop words was counted and the ratio was calculated by dividing it with the number of words obtained previously.
   e) **Cardinality:** The presence of number or words which in any sense depicted the cardinal numbers were counted and stored.
2) POS Tag Analysis
   After the POS tags plot analysis, we came to know

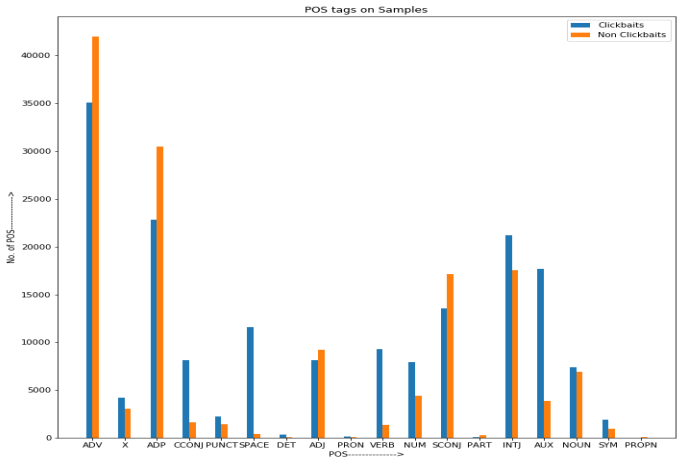

Fig. 2. Parts-of-Speech (POS) tags for words in both clickbait and non-clickbait headlines

that Coordinating Conjunctions, Auxiliary Verbs, Verbs and the number of words was present on a quite greater amount on the clickbaits rather than non clickbaits. Thus, the following features were extracted.

   a) **Coordinating Conjunction:** For, Nor, or, and, So, etc.
   b) **Verbs:** Eat, Sleep, Rest, Play, etc.

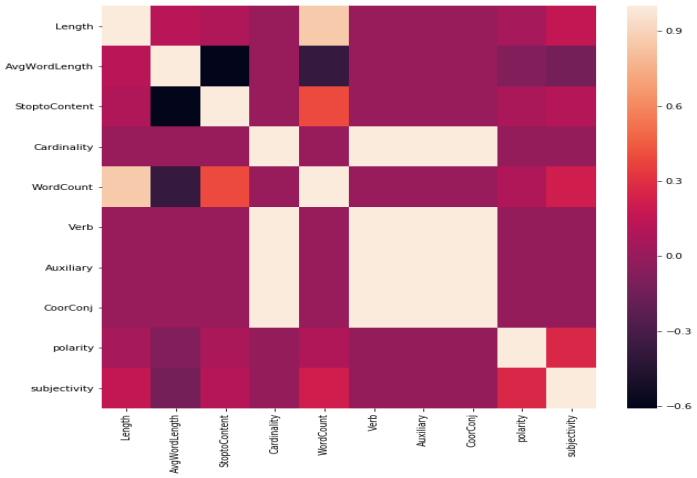

Fig. 3. Heatmap and Correlation Plot of Extracted Features

   c) **Auxiliaries:** Be, Have, Had, Will, etc.
3) Sentinment Analysis
   On the basis of the clickbait and non clickbait headlines the sentimental analysis was also performed using the library TextBlob [11]. The following features were extracted from it:
   a) **Polarity:** Its value ranged from -1 to +1 and it determined the intention of the text. i.e. positivity, neutral and negativity of the text where +1 denoted the most positive, -1 the most negative and the 0 the neutral one.
   b) **Subjectivity**: Its value ranged from 0 to 1. And this feature gave the measure of the data being a fact.

As a whole, a total of 10 features were extracted from the click bait and non clickbait headlines. After about 10 features were extracted, heatmap and correlation plot were visualized as follows :

From this heatmap, it seemed like that Verbs, Auxiliaries, Coordinating Conjunctions and cardinality were corelated to some extent but not completely. So, all the features that were extracted were forwarded for classification.

D. Algorithm Description (Logistic Regression)

1) Definition: Logistic regression [12] is a classification algorithm used to assign observations to a discrete set of classes. Some of the examples of classification problems are Email spam or not spam, Online transactions Fraud or not Fraud, Tumor Malignant or Benign. Logistic regression transforms its output using the logistic sigmoid function to return a probability value.So, like naive Bayes, logistic regression is a probabilistic classifier that makes use of supervised machine learning.

The goal of binary logistic regression is to train a classifier that can make a binary decision about the class of a new

input observation. Here we introduce the sigmoid classifier, as shown in figure 2, that will help us make this decision. Mathematically Sigmoid function is given as:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

where

$$h_\theta(x)$$

denotes the hypothesis or value of y for z

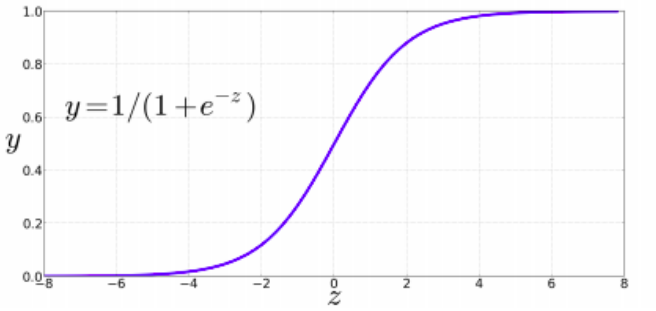

Fig. 4. Sigmoid Function

Logistic regression has two phases:

1) **training:** we train the system (specifically the weights w and b) using stochastic gradient descent and the cross-entropy loss.
2) **test:** Given a test example x we compute p(y|x) and return the higher probability label y = 1 or y = 0.

Similarly, We need a loss function that expresses, for an observation x, how close the classifier output $\hat{y} = \sigma(w.x + b)$ is to the correct output (y, which is 0 or 1). We'll call this:

$L(\hat{y}, y)$ = How much $\hat{y}$ differs from the true y.

2) Loss Function: We do this via a loss function that prefers the correct class labels of the training examples to be more likely. This is called conditional maximum likelihood estimation: we choose the parameters w,b that maximize the log probability of the true y labels in the training data given the observations x. The resulting loss function is the negative log likelihood loss, generally called the cross-entropy loss.

$$L_{CE}(\hat{y}, y) = -logp(y|x) = -[ylog\hat{y} + (1-y)log(1-\hat{y})]$$

Finally, we can plug in the definition of $\hat{y} = \sigma(wx + b)$:

$$L_{CE}(w,b) = -[ylog\sigma(w.x+b) + (1-y)log(1-\sigma(w.x+b))]$$

This loss function also ensures that as the probability of the correct answer is maximized, the probability of the incorrect answer is minimized; since the two sum to one, any increase in the probability of the correct answer is coming at the expense of the incorrect answer. It's called the cross-entropy loss, because above equation is also the formula for the cross-entropy between the true probability distribution y and our estimated distribution $\hat{y}$.

3) Optimization Function(Gradient Descent): Gradient descent is a method that finds a minimum of a function by figuring out in which direction (in the space of the parameters θ) the function's slope is rising the most steeply, and moving in the opposite direction. The intuition is that if you are hiking in a canyon and trying to descend most quickly down to the river at the bottom, you might look around yourself 360 degrees, find the direction where the ground is sloping the steepest, and walk downhill in that direction.
For logistic regression, this loss function is conveniently convex. A convex function has just one minimum; there are no local minima to get stuck in, so gradient descent starting from any point is guaranteed to find the minimum. (By contrast, the loss for multi-layer neural networks is non-convex, and gradient descent may get stuck in local minima for neural network training and never find the global optimum.)

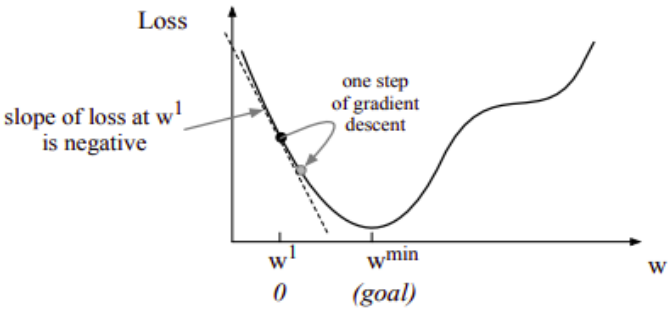

Fig. 5. Gradient Descent

The magnitude of the amount to move in gradient descent is the value of the slope

$$\frac{\partial(f(x,w))}{\partial(w)}$$

weighted by a learning rate η. A higher (faster) learning rate means that we should move w more on each step. The change we make in our parameter is the learning rate times the gradient (or the slope, in our single-variable example):

$$w^{t+1} = w^t - \eta \frac{\partial}{\partial(w)} f(x,w)$$

In order to update θ we need the definition of gradient $(f(x;\theta), y)$. Recall for logistic regression, the cross entropy, the derivative of this function for one observation vector x is:

$$\frac{\partial(L_{CE}(w,b))}{\partial(w_j)} = [\sigma(w.x+b) - y]x_j$$

Note that the gradient with respect to a single weight w_j represents a very intuitive value: the difference between the true y and our estimated $yhat = \sigma(w.x + b)$ for that observation, multiplied by the corresponding input value x_j.

4) Regularization [13]: There is a problem with learning weights that make the model perfectly match the training data. If a feature is perfectly predictive of the outcome because it happens to only occur in one class, it will be assigned a

very high weight. The weights for features will attempt to perfectly fit details of the training set, in fact too perfectly, modeling noisy factors that just accidentally correlate with the class. This problem is called overfitting. A good model should be able to generalize well from the training generalize data to the unseen test set, but a model that overfits will have poor generalization. regularization To avoid overfitting, a new regularization term $R(\theta)$ is added to the objective function maximizing log probability rather than minimizing loss, and removing the $\frac{1}{m}$ term which doesn't affect the argmax):

$$\hat{\theta} = argmax_\theta \sum_{i=1}^{m} logP(y^{(i)}|x^{(i)} - \alpha R(\theta))$$

The new regularization term $R(\theta)$ is used to penalize large weights. Thus a setting of the weights that matches the training data perfectly but uses many weights with high values to do so will be penalized more than a setting that matches the data a little less well, but does so using smaller weights.

V. Results

Using the Logistic Regression as the classification model with varied regularisation following metrics of calculations were obtained:

Reg.	Pre.	Rec.	f_1 Score	Acc.	ROC AUC
L1	0.793468	0.786617	0.790028	0.788125	0.788146
L2	0.794512	0.785692	0.790078	**0.788438**	0.788475

Use of varied regularisation did not made a much difference but the overall score using L2 was slight greater than the score given by L1. So, L2 regularisation was used.

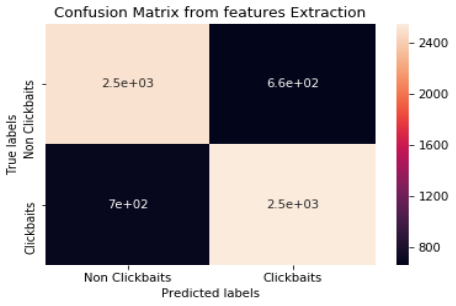

Fig. 6. Confusion Matrix Based on Features Extracted

VI. Some Other Approaches

1) **Cosine Similarity [14]:** Following Steps were performed:
   a) A clickbait or non clickbait headline was taken as input.
   b) The headline was compared with 4000 random samples using spacy.

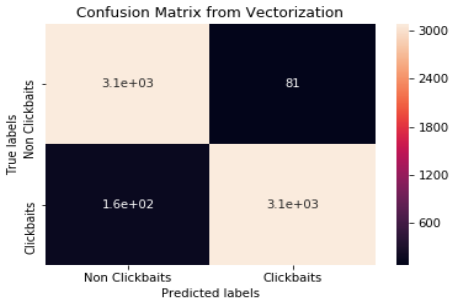

Fig. 7. Confusion matrix Based on Word Vectorization

   c) The similarity was stored in an array which consisted 4000 values.
   d) An average of those values was taken.
   e) If the average was greater than 70% the input was classified as "Clickbait". If the average was between 40 and 70 , it was classified as "Not able to classify" and if the average was even less than 40%, the input was classified as a non clickbait.

   This approach was unable to give a proper classification of random input samples. It just laid an accuracy of 61.2% which was worser to the other approaches we performed.
2) **Word Vectorization:** Following Steps were performed:
   a) Count and TF-IDF vectorizer was used to covert the given clickbaits and non clickbaits into vector representations.
   b) The vectorizer was then given as an input to a Logistic Regression classifier.
   c) Classificaition Model was trained.

   Similarly, following scores were obtained when Logistic Regression model was trained and tested using features from word vectorization: Accuracy: 0.962656, Precision: 0.974416, Recall: 0.951280, f_1 score: 0.962709, ROC AUC: 0.962811.

The approach was simple and result was also good. But, in some cases, the classifier also was not able to identify some simple inputs which were easily identified by the feature extracted approach. Thus, this can also imply that the extracted features was able to classify the inputs as clickbaits and non clickbaits. As a whole implying that the numbers were able to represent the clickbaits and non clickbaits data.

VII. Discussion on Achievements

The results so obtained based on the number and types of features extracted from the textual data were pretty inspiring. An f_1 score of 0.790 solely based on lexical features paves a promising pathway for the introduction and inclusion of semantic features like word embedding and n-grams in future works. On varied regularisation penalty, there was no any significant changes to the result indicating the model is not

over fitted and the accuracy is not misleading. With an addition of source of extraction of the news headlines itself as a feature, the results is sure to be improved.

Cosine Similarity has its own problems. There is a dilemma for making similarity comparison, either with manually labelled headlines or with randomly selected headlines from golden dataset. Here, with randomly selected entries 61.2% accuracy shows poor performance of the model. The word vectorization approach yielded maximum score in testing dataset, but performed poorly in further experimentation with foreign entries clearly demonstrating overfiting. Hence it was not preferred.

VIII. Conclusion and Future Enhancements

In this project, we compared clickbait and non clickbait headlines based on several highlighting differences between these categories. These differences were then utilized to extract features which were used to train our model and make predictions. A web platform based input form submission is presented to the user where they enter the suspicious title. On the basis of our trained model it would then be classified as clickbait or not. Although, only lexical or syntactic types of features were used highlighting grammatical and form of writing the news heading, such inspiring score were obtained. Hence, a clickbait detector based on ml approaches is developed.

With the limitation of this project to only machine learning approaches, semantic feature set couldn't be included. However the job is far from over. Our future works lies in improving the classification by extracting more features based on further extensive study and with the use of Deep Learning Classification Techniques like neural networks and more. We intend to extract semantic based features using NLP techniques and readily available Stanford tools and implement several other machine learning models to hit and trail the best one. Implementation of Image Recognition for classification is also one of the project milestones to be acheived in near future. Implementation of Multi-Lingual Clickbait Detection will also have huge prospects. Finally, it is our belief that combating the prevalence of clickbaits should be a community initiative and towards that end, so we have made our source code publicly available at [15], so that the researcher and the developer communities can come forward, and collectively make the effort a grand success.

Acknowledgment

We would like to express our heartfelt gratitude to Sanjeeb Prasad Pandey sir, PhD and Basanta Joshi sir, PhD and all our teaching assistants for their extensive support during the project and co-operating with us to help us carry our project smoothly. Their experiences and teachings in the field have been a great asset for our project. Their encouraging words and working techniques have made this project a successful one.

Also, Our deepest appreciation to entire Fuse Machines AI Center Team and to all those who provided us the possibility to complete this project. We extend our gratitude to all who have deliberately or unknowingly added a brick in the completion of this project. We enjoyed the duration of the work studying different modules and creating this project.

References

[1] Y. Chen, N. J. Conroy, and V. L. Rubin, "Misleading online content: Recognizing clickbait as false news," in *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection*. ACM, 2015, pp. 15–19.

[2] S. E. Bird, "Storytelling on the far side: Journalism and the weekly tabloid," *Critical Studies in Media Communication*, vol. 7, no. 4, pp. 377–389, 1990.

[3] B. Frampton, "Clickbait: The changing face of online journalism," *BBC News*, vol. 14, no. 09, p. 2015, 2015.

[4] J. N. Blom and K. R. Hansen, "Click bait: Forward-reference as lure in online news headlines," *Journal of Pragmatics*, vol. 76, pp. 87–100, 2015.

[5] M. Potthast, S. Köpsel, B. Stein, and M. Hagen, "Clickbait detection," in *European Conference on Information Retrieval*. Springer, 2016, pp. 810–817.

[6] D. Rowe, "Obituary for the newspaper? tracking the tabloid," *Journalism*, vol. 12, no. 4, pp. 449–466, 2011.

[7] A. Gianotto, "Downworthy: A browser plugin to turn hyperbolic viral headlines into what they really mean," *downworthy. snipe. net*, 2014.

[8] W. Markus, ""clickbait remover for facebook," *chrome. google. com/webstore/detail/clickbait-remover-forfac/hkbhmlgcpmneffdambmemapgiiiniagj*.

[9] J. Dvorkin, "Column: Why click-bait will be the death of journalism," *pbs. org/newshour/making-sense/what-you-dont-know-aboutclick-bait-journalism-could-kill-you*, 2015.

[10] P. Vossen, R. Agerri, I. Aldabe, A. Cybulska, M. van Erp, A. Fokkens, E. Laparra, A. Minard, A. Aprosio, G. Rigau *et al.*, "Newsreader: how semantic web helps natural language processing helps semantic web. special issue knowledge based systems."

[11] S. Loria, "textblob documentation," Technical report, Tech. Rep., 2018.

[12] R. E. Wright, "Logistic regression." 1995.

[13] M. Nikolova, "Regularisation functions and estimators," in *Proceedings of 3rd IEEE International Conference on Image Processing*, vol. 2. IEEE, 1996, pp. 457–460.

[14] A. Huang, "Similarity measures for text document clustering," in *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*, vol. 4, 2008, pp. 9–56.

[15] K. S. Prajwol Lamichhane, "Implementation of machine learning approach to detect clickbaits in online news." Fuse Machines Inc., 2020, pp. 15–19.